GAM-Agent: Game-Theoretic and Uncertainty-Aware Collaboration for Complex Visual Reasoning

Jusheng Zhang[1,*], Yijia Fan[1,*], Wenjun Lin[1], Ruiqi Chen[1],

Haoyi Jiang[1], Wenhao Chai[2], Jian Wang[3], Keze Wang[1,†]

[1]Sun Yat-sen University
[2]Princeton University
[3]Snap Inc.

[†]Corresponding author: kezewang@gmail.com

Abstract

We propose **GAM-Agent**, a game-theoretic multi-agent framework for enhancing vision-language reasoning. Unlike prior single-agent or monolithic models, GAM-Agent formulates the reasoning process as a non-zero-sum game between base agents—each specializing in visual perception subtasks—and a critical agent that verifies logic consistency and factual correctness. Agents communicate via structured claims, evidence, and uncertainty estimates. The framework introduces an uncertainty-aware controller to dynamically adjust agent collaboration, triggering multi-round debates when disagreement or ambiguity is detected. This process yields more robust and interpretable predictions. Experiments on four challenging benchmarks—MMMU, MMBench, MVBench, and V*Bench—demonstrate that GAM-Agent significantly improves performance across various VLM backbones. Notably, GAM-Agent boosts the accuracy of small-to-mid scale models (e.g., Qwen2.5-VL-7B, InternVL3-14B) by 5–6%, and still enhances strong models like GPT-4o by up to 2–3%. Our approach is modular, scalable, and generalizable, offering a path toward reliable and explainable multi-agent multimodal reasoning.

1 Introduction

Recently, large language models (LLMs) [43, 31, 2, 19, 50, 53, 67] has made significant progress in tasks such as complex reasoning, code generation, and knowledge integration, overcoming the limitations of single models through multi-agent collaboration [74, 11, 28, 33, 4, 71]. In contrast, although there have been advances in integrating visual perception with language understanding in the field of vision language models (VLMs) [42, 27, 6, 10, 9, 68, 70, 69, 73], the potential of multi-agent collaboration for VLMs remains largely untapped. Most existing VLM approaches rely on single models or simple ensemble strategies, which struggle to address challenges like multi-step reasoning and visual ambiguity in complex visual reasoning tasks [17, 10]. Benefiting from the development of multi-agent systems [8, 55, 7], some recent studies have proposed multi-agent debate frameworks to compensate for the shortcomings of VLMs and have obtained promising results[56, 11, 57, 25]. However, these methods typically adopt simple mechanisms such as averaging or voting mechanisms and lack visual reasoning-based strategic interactions between agents [55, 18]. Hence, they perform poorly in high-complexity visual reasoning scenarios (e.g., MMMU [64]). Just as human experts reach consensus through strategic game-like interactions when they disagree [47, 45, 58, 3], introducing a deep collaborative game-theoretic mechanism into complex visual reasoning is a valuable endeavor[13], which has already seen preliminary applications in some LLM-based works [33, 52].

However, current collaborative game-theoretic methods are often highly complex and heavily rely on reasoning paths, clues, and the fusion of multiple information, making them difficult to apply directly

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

to visual reasoning [54, 14, 12, 72, 48]. To address this issue, this paper explores the underlying architecture of existing VLMs, effectively utilizes important intermediate results in the reasoning process, and extracts representations of uncertainty in reasoning outcomes [21]. Specifically, we propose a collaborative game-theoretic framework [37, 26, 44, 30], named GAM-Agent, based on game-theoretic and uncertainty-aware inference. In this way, the complex visual reasoning process can be modeled as a non-zero-sum game involving multiple agents collaborating to reach a consensus[63, 26]. Specifically, we encourage agents to share their respective assessments of reasoning uncertainty and engage in a progressive interactive game to guide GAM-Agent to evolve step-by-step and ultimately reach a consensus.

To address these challenges, we introduce **GAM-Agent**, a novel agent collaboration framework centered around a strategic interplay between two specialized agent cohorts: *Base Agents* and *Critical Agents*. The Base Agents are tasked with initial visual interpretation and evidence generation from distinct perspectives, such as object recognition, scene description, and textual analysis from images. Concurrently, Critical Agents, acting as reasoning critique experts, scrutinize the outputs from Base Agents and evaluate factual accuracy, logical coherence, and overall completeness. The core of our GAM-Agent lies in modeling the interaction between these Base and Critical Agents as a non-zero-sum game, fundamentally arbitrated by quantified uncertainty. In this game, agents iteratively share and refine their uncertainty assessments regarding their claims and evidence, engaging in a structured debate process aimed at progressively reducing ambiguity and converging toward a consensus. This uncertainty-driven, game-theoretic collaboration allows for dynamic and strategic integration of diverse insights, leading to more robust and reliable visual reasoning outcomes. Specifically, *Base Agents* first generate diverse preliminary analyses and identify supporting evidence for their claims. These outputs are then processed by a *Claim Parser* module, which deconstructs the unstructured responses into structured information tuples, and an *Evidence Mapping* module, which links these textual claims to specific visual regions in the input image, thereby grounding the reasoning process. Besides, an *Uncertainty Quantification* mechanism continuously assesses the confidence of each agent's contribution. The entire process is orchestrated by a *Debate Controller & Integrator*. This component first evaluates the initial consensus and system uncertainty. If significant discrepancies or high uncertainty are detected, it initiates an iterative debate. During this debate, *Base Agents* refine their arguments, while *Critical Agents* provide evaluations, with the *Uncertainty Quantification* guiding the dynamic weighting and integration of information. This iterative loop continues, progressively refining the collective understanding and reducing uncertainty until a robust consensus is achieved or termination criteria are met. Extensive and comprehensive evaluations on large-scale benchmarks demonstrate the superiority of our GAM-Agent. Experimental results show that GAM-Agent achieves significant performance improvements on multiple complex visual reasoning benchmarks such as MMMU, MMBench, MVBench, and V*Bench. For example, it boosts the accuracy of small-to-mid scale VLMs by 5–6% and enhances the accuracy of top-tier models like GPT-4o by up to 2–3%.

2 Methodology

Our GAM-Agent is defined as a six-tuple $S = (E, A, \Phi, M, P, D)$: $E = 1, 2, \ldots, N$ represents the set of N expert agents, each possessing distinct cognitive perspectives. $A = A_1, A_2, \ldots, A_N$ defines the set of analysis capability mapping functions, where $A_i : \mathcal{X} \times \mathcal{P}r \rightarrow \mathcal{R}$ maps an input image X and a question P_r to agent i's response R_i. $\Phi = \Phi_1, \Phi_2, \ldots, \Phi_N$ is the set of uncertainty assessment functions, where $\Phi_i : \mathcal{R} \rightarrow [0, 1]$ quantifies the uncertainty U_i associated with response R_i. $M = M_1, M_2, \ldots, M_N$ represents the set of evidence localization mapping functions, where $M_i : \mathcal{C}i \times \mathcal{X} \rightarrow \mathcal{V} \times [0, 1]$ associates a claim c with a visual region r and a confidence score σ. $\mathcal{C}i$ denotes the set of claims made by agent i, and $\mathcal{V}$ is the space of visual regions. $P : \mathcal{R} \rightarrow (c_j, \sigma_j, e_j, r_j)j = 1^m$ is the claim and evidence parser, converting an unstructured response R into a set of structured information tuples, each containing a claim c_j, confidence σ_j, textual evidence e_j, and visual region reference r_j. $D : \mathcal{X} \times \mathcal{P}r \times R_i, U_i i = 1^N \rightarrow \mathcal{R}_{final}$ the debate module aligns conflict and consensus for R_{final}.

2.1 Base and Critical Agents

Our GAM-Agent employs two distinct categories of agents: Base Agents and Critical Agents, each with specialized roles in the visual reasoning process. **Base Agents[11]** are conceived as *specialized*

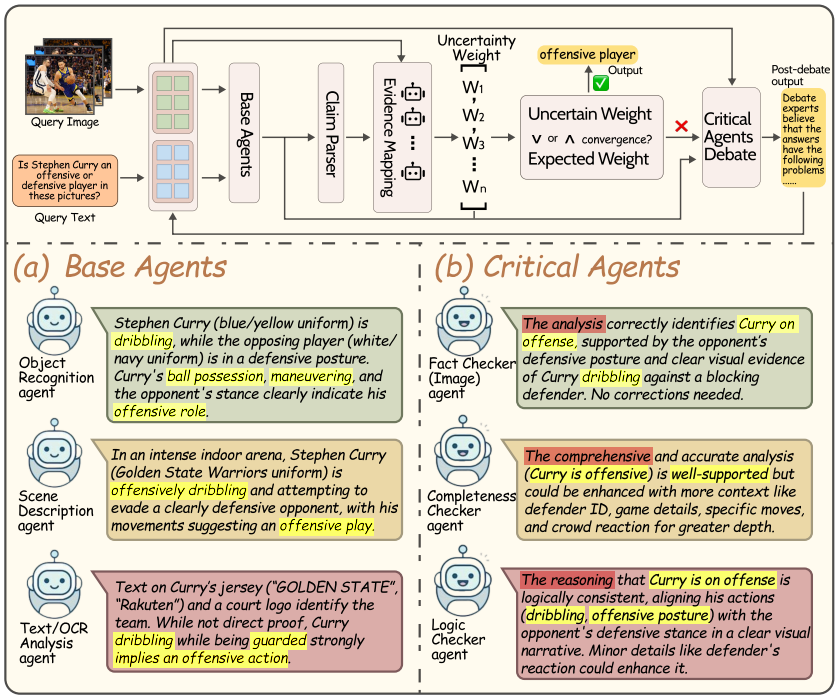


Figure 1: Our GAM-Agent pipeline processing a visual query: Initial responses from Base Agents(R_i) undergo Claim Parsing and Evidence Mapping before an Uncertainty Convergence Judgment assesses consensus. If required, the Debate Controller & Integrator orchestrate an iterative loop, engaging both Base Agents for argumentation and Critical Agents for verification, refining the response until convergence or termination to produce the final output.

visual reasoning experts. Their primary function is to analyze the visual input and generate initial interpretations or findings related to specific aspects of the scene or query. Examples of Base Agents include: an **Object Recognition Agent**, tasked with identifying and localizing objects within the visual input; a **Scene Description Agent**, focused on generating a textual description of the overall scene, its context, and the relationships between entities; and an **OCR Agent**, specialized in extracting textual information from images. Critical Agents function as *specialized reasoning critique experts*. Their role is to scrutinize the outputs generated by the Base Agents (and potentially other Critical Agents), identify inconsistencies, and assess factual accuracy, logical coherence, and overall completeness. Examples of Critical Agents include: a **Fact Checker Agent**, which verifies the factual claims made by other agents against known information or contextual understanding; a **Completeness Checker Agent**, which assesses whether the information provided is sufficient and comprehensive in addressing the query or task; and a **Logic Checker Agent**, which evaluates the logical consistency and validity of the reasoning steps and conclusions presented. With these two tiers of expert agents, our GAM-Agent introduces an iterative game-theoretic interplay and information refinement process via uncertainty quantification to perform robust and accurate visual reasoning.

2.2 Uncertainty Quantification Function (Φ)

We propose a dual-level Φ_i that models uncertainty from both response generation and overall semantics. **Generation-Process Based Uncertainty (Φ_{igen+})** When the probability distribution $P_{i,t}$ of the underlying VLM during the generation of the token sequence $Y_i = (y_{i,1}, \ldots, y_{i,T_i})$ is accessible, we propose an integrated multi-feature uncertainty metric $\Phi_{igen+}(R_i)$

$$\Phi_{igen+}(R_i) = \frac{1}{T_i} \sum_{t=1}^{T_i} \left[\alpha \cdot \underbrace{H(P_{i,t})}_{\substack{\text{Information Entropy} \\ \text{(Hesitation)}}} + \beta \cdot \underbrace{\max(0, 1 - \Delta_{top}(P_{i,t}))}_{\substack{\text{Probability Difference Inverse} \\ \text{(Lack of Conviction)}}} \right] \tag{1}$$

where $H(P_{i,t})$ is the information entropy quantifying the dispersion of the token distribution, reflecting the model's hesitation, and $\Delta_{top}(P_{i,t}) = p_{i,t}(1) - p_{i,t}(2)$ is the difference between the top two token probabilities, capturing the model's conviction in its top choice. Parameters α and β balance

these complementary signals. This method captures microscopic uncertainty patterns during token generation, shifting from output evaluation to generation process evaluation.

2.2.1 Semantic-Marker Based Uncertainty (Φ_{isem})

When generation probabilities are unavailable, our GAM-Agent adopts a semantic marker-based strategy, $\Phi_{isem}(R_i)$, to assess uncertainty. This strategy systematically identifies and quantifies uncertainty markers within the text: $\Phi_{isem}(R_i) = \sigma_{\text{sigmoid}}\left(k \cdot (\rho(R_i) - \text{offset})\right)$, where W represents a multi-level lexicon of uncertainty markers. For each marker $w \in W$, weight(w) reflects its uncertainty intensity, count(w, R_i) denotes its frequency in response R_i, and $|R_i|$ is the length of the response. The parameters k and offset control scaling and bias, respectively, while $\sigma_{\text{sigmoid}}(\cdot)$ normalizes the score to the $(0, 1)$ interval. By statistically analyzing semantic features in the text, this strategy provides a reliable estimate of uncertainty. Moreover, the uncertainty U_i for agent i is determined using a priority strategy: if generation probabilities are available, $U_i = \Phi_{igen+}(R_i)$, capturing uncertainty inherent to the generation process. Otherwise, $U_i = \Phi_{isem}(R_i)$ is used, reflecting uncertainty inferred from semantic markers. This enables reliable uncertainty quantification for U_i in any setting, supporting response weighting, conflict detection, and iterative debate.

2.3 Initial Integration and Conflict Detection

After obtaining initial responses from expert agents, GAM-Agent introduces an adaptive weight allocation and conflict assessment mechanism. This initial integration process efficiently aggregates diverse viewpoints and precisely identifies cognitive disagreements requiring debate.

Initial Weight Allocation[74] and Response Integration. GAM-Agent dynamically assigns initial weights $w_i^{(0)}$ based on each agent's uncertainty assessment U_i and generates an integrated response $R^{(0)}$: $w_i^{(0)} = \frac{e^{-\beta U_i}}{\sum_{j=1}^{N} e^{-\beta U_j}} \rightarrow R^{(0)} = \text{IntegrateJudge}(X, P_r, (R_i, w_i^{(0)})_{i=1}^{N})$. The parameter β controls the sensitivity of weights to uncertainty. The IntegrateJudge function performs quality assessment, mutual support analysis, and information merging, balancing high-confidence opinions with diverse perspectives.

Conflict Detection and Debate Triggering. GAM-Agent calculates the weighted average uncertainty $U_{sys}^{(0)}$ and an inter-expert conflict score ($ConflictScore$) to decide whether to initiate the debate process using a dual-criterion strategy:

$$U_{sys}^{(0)} = \underbrace{\sum_{i=1}^{N} w_i^{(0)} U_i}_{\text{Weighted average uncertainty}} \quad \rightarrow \quad \text{TriggerDebate} = \underbrace{(U_{sys}^{(0)} > \theta_U)}_{\substack{\text{System uncertainty} \\ \text{exceeds threshold}}} \vee \underbrace{(ConflictScore > \theta_C)}_{\substack{\text{Significant inter-expert} \\ \text{conflict detected}}} \tag{2}$$

where $ConflictScore$ is computed from the consistency of key claims, evidence interpretation, and agent logic. If $U_{sys}^{(0)} > \theta_U$ or $ConflictScore > \theta_C$, the system triggers an iterative debate. This ensures quick consensus on easy cases while invoking deeper collaboration when needed.

2.4 Evidence Mapping (M) and Claim Parsing (P)

GAM-Agent incorporates an Evidence Mapping module (M) and a Claim Parsing module (P) as core components bridging visual and language reasoning. These modules construct a traceable network linking textual claims to specific visual evidence, ensuring precise alignment between reasoning and visual information. The collaborative mechanism of these modules can be formalized as follows: the evidence mapping module M_i establishes explicit links between an agent's claim $c \in \mathcal{C}_i$ and a visual region r in image X (specified by bounding box, mask, or description), with $\sigma \in [0, 1]$ quantifying the confidence of this association, that is, $M_i : \mathcal{C}_i \times X \rightarrow (r, \sigma)$. This creates a direct "claim-visual evidence" link. The claim parsing module P transforms an unstructured response R_i into a set of structured tuples $P(R_i) \rightarrow \{(c_j, \sigma_j, e_j, r_j)\}_{j=1}^{m_i}$, where each tuple consists of c_j (the specific textual claim), σ_j (the associated confidence), e_j (a description of the supporting textual evidence), and r_j (a reference to the relevant visual region).

Working Mechanism of Evidence Mapping and Parsing. The evidence mapping process can be represented as a conditional mapping function from the claim space to the visual region space: $p(r|c, X) = M_i(c, X) \rightarrow (r, \sigma)$ where $p(r|c, X)$ represents the probability distribution over visual

regions r being relevant evidence given claim c and image X, and σ is the confidence of this mapping. The claim parsing process can be formalized as text decomposition and structural reorganization:

$$P(R_i) = \underbrace{M_i(\{c_j\}, X) \to \{r_j\}_{j=1}^{m_i}}_{\text{Visual Region Mapping}} \circ \underbrace{\text{Associate}(\{c_j\}, R_i) \to \{e_j\}_{j=1}^{m_i}}_{\text{Evidence Association}} \circ \underbrace{\text{Assess}(\{c_j\}) \to \{\sigma_j\}_{j=1}^{m_i}}_{\text{Confidence Assessment}} \circ \underbrace{\text{Extract}(R_i) \to \{c_j\}_{j=1}^{m_i}}_{\text{Claim Extraction}} \tag{3}$$

This decomposition highlights four steps: claim extraction, confidence assessment, evidence association, and visual region mapping, collectively transforming R_i into structured tuples. This can enhance transparency and evidence traceability in complex visual reasoning by grounding the visual evidence, enabling precise conflict localization, and performing fine-grained debate focusing.

2.5 Iterative Debate Process: Dynamics, Termination, and Consensus

When initial integration reveals high uncertainty or critical disagreements, GAM-Agent initiates an iterative debate process ("Debating Uncertainty"). This structured interaction drives the agent collective towards a high-quality consensus through a feedback loop focusing on key disputes, evidence sharing, and dynamic weight adjustments. The iterative debate process is formalized as a sequence of state transitions. Each round of debate k involves five key steps forming a complete state evolution cycle: (a-b) Disput focus and expert argumentation: Model identifies key points of contention and guides experts to generate targeted arguments: $C_{debate}^{(k)} = \text{IdentifyDisputes}((c_j, \sigma_j, e_j, r_j)j = 1^m, R^{(k-1)}, U_{sys}^{(k-1)})$ $Arg_i^{(k)} = A_i(X, P_r, R^{(k-1)}, C_{debate}^{(k)})$ The IdentifyDisputes function analyzes structured information from the previous round to select key claims $C_{debate}^{(k)}$ with low confidence σ_j or significant inter-expert conflict. Each expert then generates an argumentation package including argumentative text $w_i^{(k)} = \frac{\exp(-\beta U_i^{(k)}) \text{ or } \exp(\gamma C_i^{(k)})}{\sum_{j=1}^N \exp(-\beta U_j^{(k)}) \text{ or } \sum_{j=1}^N \exp(\gamma C_j^{(k)})}$ Parameters β and γ control sensitivity to uncertainty or confidence changes. This adaptive weighting incentivizes effective argumentation, aligning with game-theoretic principles. **(d-e) Iterative Response Generation and Uncertainty Evaluation:** we integrate updated expert views to generate the response of the current round and assess the consensus

$$R^{(k)} = \text{IntegrateJudge}\Big(\underbrace{X, P_r}_{\text{Original Input}} , \underbrace{R^{(k-1)}}_{\text{Prev. Round Base}} , \underbrace{\{R_j\}_{j=1}^N}_{\text{Initial Resp.}}, \underbrace{\{Arg_j^{(k)}, E_j^{(k)}, r_j^{(k)}, C_j^{(k)}\}_{j=1}^N}_{\text{Current Expert Updates}}, \underbrace{\{w_j^{(k)}\}_{j=1}^N}_{\text{Dynamic Weights}}, \underbrace{C_{\text{debate}}^{(k)}}_{\text{Dispute Focus}} \Big) \tag{4}$$

In this process, the IntegrateJudge function fuses the original input (X, P_r), the previous round's answer $R^{(k-1)}$, each expert's initial response $R_j * j = 1^N$, and their structured debate updates $Arg_j^{(k)}, E_j^{(k)}, r_j^{(k)}, C_j^{(k)} * j = 1^N$, along with dynamic weights $w_j^{(k)} * j = 1^N$ and the current set of disputed points $C * \text{debate}^{(k)}$, to produce the integrated decision $R^{(k)}$. This progressive integration focuses on resolving the present disputes $C_{\text{debate}}^{(k)}$ while preserving consensus from $R^{(k-1)}$. The overall system uncertainty $U_{sys}^{(k)} = \sum_{i=1}^N w_i^{(k)} U_i^{(k)}$quantifies the level of consensus. Through multiple iterations, GAM-Agent aggregates agreement and resolves conflict, steadily converging to a high-confidence, low-uncertainty final answer. **Debate Termination Decision Mechanism** monitors multiple criteria to decide whether to continue the debate. Termination occurs as follows:

$$\text{Terminate Debate} = \underbrace{\left(U_{\text{sys}}^{(k)} < \theta_U\right)}_{\text{Uncertainty threshold condition}} \vee \underbrace{\left(k \geq K_{\max}\right)}_{\text{Max iteration limit}} \vee \underbrace{\left(\Delta U_{\text{sys}}^{(k)} < \epsilon\right)}_{\text{Convergence speed condition}} \tag{5}$$

where $\Delta U_{sys}^{(k)} = |U_{sys}^{(k)} - U_{sys}^{(k-1)}|$ represents the change in system uncertainty between consecutive rounds, and ϵ is a convergence threshold. These conditions correspond to reaching satisfactory certainty, resource limits, or debate stagnation. Upon termination at round K, the final response is $R_{final} = R^{(K)}$. This multi-step, multi-round debate enables knowledge sharing and complementarity among agents and forms a self-improving cognitive loop for complex visual scenarios.

3 Experiments

3.1 Image and Video Understanding Benchmarks

Experiment Setup To comprehensively evaluate our GAM-Agent on challenging image and video benchmarks, we compare it with recent multi-agent methods using various state-of-the-art vision-language models. We select five prominent VLMs: Qwen2.5VL[41] (7B and 72B parameter

Table 1: The evaluation of image and video understanding. Values in parentheses denote the improvement in accuracy over the respective base model (Ori). The best performance for each base model on each benchmark is **bolded**. Best is in red, and Base (Ori) data is in blue.

Base Model	Framework	MMMU	MMBench_V11_Test	MVBench_Test
Qwen2.5VL-7B (Small)	Base (Ori)	53.82	82.61	69.62
	DMAD	55.53 (+1.71)	85.62 (+3.01)	72.78 (+3.16)
	DMPL	56.27 (+2.45)	86.24 (+3.63)	73.49 (+3.87)
	ChatEval	56.98 (+3.16)	86.93 (+4.32)	74.25 (+4.63)
	MAD	55.01 (+1.19)	85.11 (+2.50)	72.17 (+2.55)
	DebUnc	57.59 (+3.77)	87.65 (+5.04)	74.89 (+5.27)
	GAM-Agent (Ours)	**58.93 (+5.11)**	**89.02 (+6.41)**	**76.15 (+6.53)**
Qwen2.5VL-72B (Large)	Base (Ori)	68.24	88.39	70.38
	DMAD	68.98 (+0.74)	89.88 (+1.49)	71.85 (+1.47)
	DMPL	69.43 (+1.19)	90.31 (+1.92)	72.36 (+1.98)
	ChatEval	69.81 (+1.57)	90.75 (+2.36)	72.81 (+2.43)
	MAD	68.73 (+0.49)	89.47 (+1.08)	71.42 (+1.04)
	DebUnc	70.15 (+1.91)	91.12 (+2.73)	73.19 (+2.81)
	GAM-Agent (Ours)	**70.98 (+2.74)**	**91.82 (+3.43)**	**74.12 (+3.74)**
InternVL3-14B (Small)	Base (Ori)	67.09	83.54	76.59
	DMAD	68.83 (+1.74)	86.63 (+3.09)	79.78 (+3.19)
	DMPL	69.52 (+2.43)	87.35 (+3.81)	80.57 (+3.98)
	ChatEval	70.27 (+3.18)	88.14 (+4.60)	81.39 (+4.80)
	MAD	68.25 (+1.16)	86.01 (+2.47)	79.13 (+2.54)
	DebUnc	70.94 (+3.85)	88.86 (+5.32)	82.15 (+5.56)
	GAM-Agent (Ours)	**72.26 (+5.17)**	**90.15 (+6.61)**	**83.23 (+6.64)**
InternVL3-78B (Large)	Base (Ori)	72.18	87.65	78.81
	DMAD	72.99 (+0.81)	89.21 (+1.56)	80.43 (+1.62)
	DMPL	73.47 (+1.29)	89.68 (+2.03)	80.95 (+2.14)
	ChatEval	73.88 (+1.70)	90.15 (+2.50)	81.41 (+2.60)
	MAD	72.71 (+0.53)	88.83 (+1.18)	80.01 (+1.20)
	DebUnc	74.24 (+2.06)	90.57 (+2.92)	81.83 (+3.02)
	GAM-Agent (Ours)	**75.01 (+2.83)**	**91.53 (+3.88)**	**82.62 (+3.81)**
GPT-4o-0513 (Large)	Base (Ori)	68.97	83.13	75.48
	DMAD	69.78 (+0.81)	84.77 (+1.64)	77.15 (+1.67)
	DMPL	70.26 (+1.29)	85.29 (+2.16)	77.71 (+2.23)
	ChatEval	70.65 (+1.68)	85.73 (+2.60)	78.19 (+2.71)
	MAD	69.42 (+0.45)	84.31 (+1.18)	76.73 (+1.25)
	DebUnc	71.03 (+2.06)	86.15 (+3.02)	78.57 (+3.09)
	GAM-Agent (Ours)	**71.91 (+2.94)**	**87.04 (+3.91)**	**79.52 (+4.04)**

versions), InternVL3[76] (14B and 78B parameter versions), and GPT-4o-0513[2]. On top of these base models, we implemented our GAM-Agent framework and five other multi-agent frameworks: DMAD[32], DMPL[25], ChatEval[5], MAD (Multi-Agent Debate)[29], and DebUnc[62]. The evaluation is conducted on three challenging benchmarks: MMMU[64] (multi-discipline multimodal understanding), MMBench_V11_Test (visual reasoning and perception), and MVBench_Test[34] (video temporal reasoning). For all experimental runs involving debate frameworks, we set a maximum of 3 debate rounds (`max_debate_round=3`). Each framework utilized 3 expert agents ($N = 3$) for generating initial responses and argumentation, and 3 critic agents ($N_{\text{crit}} = 3$) for frameworks requiring critique/verification, such as GAM-Agent. We used greedy decoding for text generation. Open-source models (Qwen2.5VL, InternVL3) were deployed locally on NVIDIA A100 GPUs, while the closed-source GPT-4o-0513 was accessed via the OpenRouter API. Detailed hyperparameter settings and other experimental configurations are provided in the Supplementary Material. On each benchmark, our GAM-Agent achieves higher overall accuracy (Overall ACC, %) than its corresponding base model, with improvements summarized in Table 1. Unless otherwise stated, all experiments in this paper are performed on NVIDIA A100 GPU.

Experiment Results Table 1 summarizes the comparative performance of GAM-Agent and other multi-agent frameworks across all base models and benchmarks. Results show that GAM-Agent consistently achieves the highest overall accuracy. Improvements are especially notable for smaller models (e.g., Qwen2.5VL-7B and InternVL3-14B), with gains of +5.8% to +6.7% on MMBench_V11_Test and MVBench_Test (e.g., +6.61% on InternVL3-14B/MMBench). Larger models (Qwen2.5VL-72B, InternVL3-78B, GPT-4o-0513) also see consistent gains of +2.6% to +4.1% (e.g., +3.91% on GPT-4o-0513/MMBench). On MMMU, improvements are about 1–1.5 percentage points lower but GAM-Agent still leads, such as +5.11% for Qwen2.5VL-7B. The supplementary material provides more detailed experimental analyses, including cost-effectiveness studies (Appendix A), cost dynamics during debates (Appendix B), a comparative analysis of uncertainty handling capabilities (Appendix C), empirical analysis of debate termination conditions (Appendix C.4), and additional hyperparameter studies (Appendix H). GAM-Agent consistently outperforms other multi-agent frameworks (DMAD, DMPL, ChatEval, MAD, DebUnc), delivering larger accuracy gains

Table 2: Performance on V*Bench (%). AR: Attribute Recognition, SR: Spatial Reasoning. Values in parentheses denote the accuracy improvement over the respective base model (Ori). The best performance for each base model is **bolded**. Best overall with GAM-Agent framework is highlighted in red. Base (Ori) data is in blue.

Base Model	Framework	Attribute Rec. (%)	Spatial Reas. (%)	Overall (%)
Qwen2.5VL-7B (Small)	Base (Ori)	57.39	67.11	61.26
	DMAD	59.12 (+1.73)	69.03 (+1.92)	63.28 (+2.02)
	DMPL	59.98 (+2.59)	70.15 (+3.04)	64.21 (+2.95)
	ChatEval	60.73 (+3.34)	70.92 (+3.81)	65.03 (+3.77)
	MAD	58.65 (+1.26)	68.44 (+1.33)	62.72 (+1.46)
	DebUnc	61.54 (+4.15)	71.88 (+4.77)	65.87 (+4.61)
	GAM-Agent (Ours)	**62.38 (+4.99)**	**72.53 (+5.42)**	**66.51 (+5.25)**
Qwen2.5VL-72B (Large)	Base (Ori)	64.52	73.08	68.80
	DMAD	65.88 (+1.36)	74.51 (+1.43)	70.19 (+1.39)
	DMPL	66.57 (+2.05)	75.29 (+2.21)	70.93 (+2.13)
	ChatEval	67.12 (+2.60)	75.83 (+2.75)	71.48 (+2.68)
	MAD	65.31 (+0.79)	74.02 (+0.94)	69.67 (+0.87)
	DebUnc	67.93 (+3.41)	76.54 (+3.46)	72.24 (+3.44)
	GAM-Agent (Ours)	**68.77 (+4.25)**	**77.39 (+4.31)**	**72.78 (+3.98)**
InternVL3-14B (Small)	Base (Ori)	61.06	69.87	65.53
	DMAD	62.97 (+1.91)	71.92 (+2.05)	67.50 (+1.97)
	DMPL	63.78 (+2.72)	72.81 (+2.94)	68.35 (+2.82)
	ChatEval	64.45 (+3.39)	73.55 (+3.68)	69.05 (+3.52)
	MAD	62.24 (+1.18)	71.23 (+1.36)	66.80 (+1.27)
	DebUnc	65.21 (+4.15)	74.38 (+4.51)	69.80 (+4.27)
	GAM-Agent (Ours)	**66.15 (+5.09)**	**75.32 (+5.45)**	**70.33 (+4.80)**
InternVL3-78B (Large)	Base (Ori)	68.90	76.45	72.23
	DMAD	70.02 (+1.12)	77.68 (+1.23)	73.85 (+1.62)
	DMPL	70.63 (+1.73)	78.33 (+1.88)	74.48 (+2.25)
	ChatEval	71.15 (+2.25)	78.82 (+2.37)	74.99 (+2.76)
	MAD	69.58 (+0.68)	77.21 (+0.76)	73.40 (+1.17)
	DebUnc	71.89 (+2.99)	79.57 (+3.12)	75.73 (+3.50)
	GAM-Agent (Ours)	**72.63 (+3.73)**	**80.24 (+3.79)**	**76.16 (+3.93)**
GPT-4o-0513 (Large)	Base (Ori)	71.15	78.30	74.72
	DMAD	72.33 (+1.18)	79.58 (+1.28)	75.96 (+1.24)
	DMPL	72.98 (+1.83)	80.29 (+1.99)	76.64 (+1.92)
	ChatEval	73.51 (+2.36)	80.77 (+2.47)	77.14 (+2.42)
	MAD	71.92 (+0.77)	79.05 (+0.75)	75.49 (+0.77)
	DebUnc	74.24 (+3.09)	81.49 (+3.19)	77.87 (+3.15)
	GAM-Agent (Ours)	**74.98 (+3.83)**	**82.15 (+3.85)**	**78.32 (+3.60)**
SEAL (original work)		74.78	76.31	75.39

over base models. For instance, on InternVL3-14B/MMBench_V11_Test, GAM-Agent achieves 90.15% (+6.61%), compared to DebUnc's 88.86% (+5.32%). GAM-Agent's game-theoretic design, leveraging uncertainty, evidence, and iterative debate, boosts VLM performance across model sizes and complex visual tasks, especially for smaller models, with consistent gains for larger ones.

3.2 Guided Visual Search Capabilities

Experiment Setup With the release of the o3 model, MLLMs have entered a new phase of capability in high-resolution, information-dense visual scenes, bringing attention to the challenge of fine-grained visual grounding. To evaluate such capabilities, we compare GAM-Agent and the same set of comparative multi-agent frameworks (DMAD, DMPL, ChatEval, MAD, and DebUnc) to the V*Bench benchmark. We utilized the identical five base VLMs as in Section 3.1: Qwen2.5VL (7B and 72B), InternVL3 (14B and 78B), and GPT-4o-0513. For all frameworks, including GAM-Agent, we maintained the configuration of $N = 3$ base expert agents and $N_{crit} = 3$ critic agents (when the debate is triggered), with a maximum of 3 debate rounds (`max_debate_round=3`). The objective was to assess how these collaborative frameworks, particularly GAM-Agent with its uncertainty-driven debate, enhance performance on tasks requiring guided visual search capabilities.

Experiment Results. The performance of GAM-Agent and other multi-agent frameworks on the V*Bench benchmark is presented in Table 2. The table details accuracy for Attribute Recognition (AR), Spatial Reasoning (SR), and Overall scores. Base model (Ori) scores are derived from our initial evaluations, and the SEAL framework's reported performance is included as a reference.

The results in Table 2 demonstrate that GAM-Agent consistently enhances the performance of base VLMs on the V*Bench tasks, which demand precise visual search and reasoning. For instance, GAM-Agent improved the Qwen2.5VL-7B model by +5.25% overall, achieving 66.51%. Similar significant gains are observed across other base models, such as InternVL3-14B (+4.80% overall to 70.33%) and GPT-4o-0513 (+3.60% overall to 78.32%). While other multi-agent frameworks also yield improvements over the base models, GAM-Agent consistently achieves the highest scores

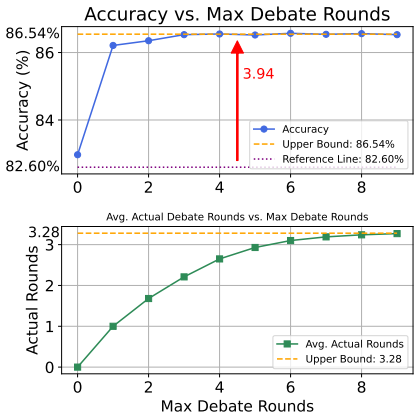


Figure 2: Acc. vs. Max Debate Rounds among them. The gains are generally more pronounced for smaller models, yet remain substantial for larger, more capable base VLMs. Notably, GAM-Agent's performance with models like GPT-4o-0513 (78.32%) approaches or surpasses SEAL performance (75.39%) on this challenging benchmark.

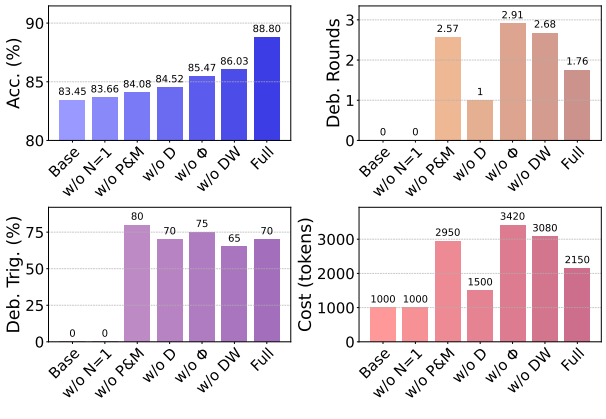


Figure 3: Ablation study results on MMBench_TEST_V11.

GAM-Agent's uncertainty-driven debate mechanism shines on V*Bench, enabling VLMs to quantify uncertainty, ground claims in visual evidence, and iteratively resolve disagreements for precise visual understanding. This guided visual search directs agents to re-evaluate details and converge on evidence-backed interpretations, boosting MLLMs' reasoning, and rivaling visual search.

3.3 Ablation Experiments

To assess each component's contribution, we conducted ablation studies by removing or simplifying modules. Experiments were run on MMBench[34]_TEST_V11 using InternVL3-14B as the base model with $N = 3$ agents (unless noted), and a maximum of 3 debate rounds (`max_debate_round=3`) for setups involving iterative debate. We report overall accuracy (Acc.), average debate rounds (Deb. Rounds), debate trigger rate (Deb. Trig.), and average LLM inference cost (Cost). Additional parameter ablations are provided in Supplementary Material 3. We define five ablation variants to assess key components of GAM-Agent: **`w/o Multi-Agent (N=1)`**: Single-agent, without collaboration or debate. **`w/o Uncertainty (Φ)`**: Removes uncertainty; uses uniform weights and simple heuristics. **`w/o P&M`**: Disables claim-evidence grounding; debate operates on raw outputs. **`w/o Debate (D)`**: No iterative refinement; uses only initial integration. **`w/o Dynamic Weights`**: Debate with fixed agent weights. Reported metrics: Accuracy (Acc. %), Avg. Debate Rounds (Deb. Rounds, max 3), Debate Trigger Rate (Deb. Trig. %), and Inference Cost (tokens/instance). **Experiment Results** Figure 3 shows that the full GAM-Agent achieves the highest accuracy (88.80%) with the fewest debate rounds (1.76), reflecting efficient convergence. All ablations degrade performance: removing `Uncertainty` or `Dynamic Weights` increases debate length and cost; disabling `Iterative Debate` cuts cost but reduces accuracy by 4.28%; removing `P&M` causes a 4.72% drop and high debate triggers (80%). The `w/o Multi-Agent` setup performs only slightly above the base model, confirming the necessity of all components.

3.4 Study on Maximum Debate Rounds

To assess the effect of debate length, we ablated the `max_debate_round` hyperparameter using Qwen2.5VL-7B on the MMBench_V11_Test set. The core GAM-Agent components—uncertainty modeling, claim parsing, and dynamic weighting—were kept unchanged. We varied `max_debate_round` from 0 to 9 and recorded overall accuracy and average actual debate rounds. The results presented in Figure 2 show that increasing `max_debate_round` improves accuracy up to 3 rounds: from 82.97% (0 rounds) to 86.21% (1), 86.35% (2), and 86.53% (3). Beyond this, accuracy plateaus around 86.52%–86.57%, converging near 86.59%. Similarly, average actual debate rounds grow with higher `max_debate_round`, from 0.00 (0) to 1.00 (1), 2.21 (3), and up to 3.27 (9), showing early termination via internal convergence criteria. These results highlight that 3–4 debate rounds suffice for near-optimal accuracy with limited cost, as GAM-Agent adaptively halts when uncertainty is resolved.

3.5 Expert Weight Trajectory and System Uncertainty Dynamics in Logical Reasoning Tasks

To analyze how GAM-Agent modulates expert influence and manages uncertainty during reasoning, we tracked expert contributions across eight base VLMs—Qwen2.5VL (3B–72B) and InternVL3

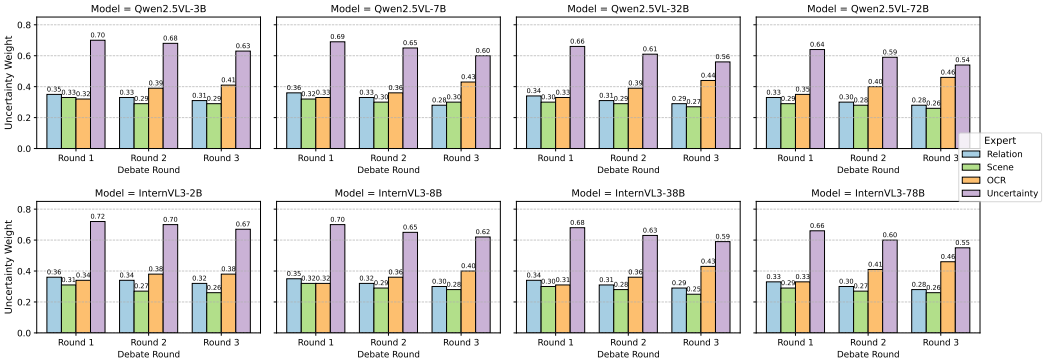


Figure 4: Evolution of Expert Weights (Relational Reasoning, Scene Description, OCR) and System Uncertainty across three debate rounds for various Qwen2.5VL and InternVL3 series models in logical reasoning tasks. The four bars for each round represent the weights of the Relation expert, Scene expert, OCR expert, and the overall System Uncertainty, respectively.

(2B–78B). Each setup included three expert agents: Relational Reasoning, Scene Description, and OCR (see Supplementary for prompts). We visualized uncertainty-based weights and system uncertainty over three debate rounds (Figure 4), revealing how GAM-Agent dynamically adjusts expert influence to reduce uncertainty and drive consensus. Figure 4 illustrates GAM-Agent's dynamic expert modulation during reasoning. System uncertainty (purple bars) consistently decreases over three rounds (e.g., Qwen2.5VL-72B: 0.64→0.54; InternVL3-78B: 0.66→0.55), indicating effective convergence. Meanwhile, OCR expert weights (yellow) typically rise (e.g., Qwen2.5VL-72B: 0.35→0.46), while Relational and Scene experts adjust downward. OCR provides stable cues; expert-task alignment varies. Larger models reduce uncertainty, showing GAM-Agent's capacity-aware, uncertainty-driven coordination.

4 Related Works

Multi-agent Systems (MAS) [75, 24, 4, 8, 15] have gained significant attention in artificial intelligence, particularly with the integration of large language models (LLMs). These systems enhance capabilities in areas such as complex reasoning, code generation, and knowledge integration by leveraging collaborative agents [10]. For example, Du et al [11] demonstrated improved factuality and reasoning in LLMs through multi-agent debates, while Li et al [28] explored communicative agents to simulate language model societies [40, 39, 60]. Despite these advances, the application of multi-agent collaboration in vision-language models (VLMs) remains largely untapped. Existing multi-agent debate frameworks often lack strategic interactions tailored for visual reasoning.

Uncertainty Quantification [30, 1, 61, 16, 51] is a cornerstone of reliable decision-making in machine learning, particularly in visual reasoning where ambiguities, such as occlusions or poor lighting, frequently arise. Previous methods, including entropy-based measures and semantic analysis, have been widely adopted [21, 30]. In VLMs, uncertainty stems from both visual and linguistic sources, complicating accurate assessment [46].

Visual Reasoning entails interpreting and inferring from visual inputs and is a critical domain for VLMs [36, 23, 6, 10, 9]. Current VLM strategies often depend on single models or basic ensemble methods, which struggle with multi-step reasoning and visual ambiguities [59, 65, 35]. Unlike basic averaging or voting in VLM multi-agent debate, GAM-Agent introduces a game-theoretic collaboration mechanism, leveraging uncertainty and intermediate reasoning for deeper agent synergy.

5 Conclusion

We presented GAM-Agent, a game-theoretic multi-agent reasoning framework designed to enhance the robustness, interpretability, and accuracy of vision-language models on complex multimodal tasks. GAM-Agent facilitates structured claim-evidence interactions guided by uncertainty-aware collaboration by modeling reasoning as a non-zero-sum game among specialized base agents and a critical verification agent. This design allows the system to identify ambiguities, trigger multi-round debates, and dynamically adjust reasoning strategies based on agent confidence and disagreement. Extensive experiments across four challenging benchmarks demonstrate that GAM-Agent consistently improves performance across a wide range of VLM backbones, including both lightweight and strong foundation models. Looking forward, we believe this work opens new directions in self-reflective and

multi-agent multimodal reasoning. Future research may explore tighter integration with external tools, extending to dialog agents, or applying GAM-Agent to safety-critical and explainable AI systems.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 62276283, in part by the China Meteorological Administration's Science and Technology Project under Grant CMAJBGS202517, in part by Guangdong Basic and Applied Basic Research Foundation under Grant 2023A1515012985, in part by Guangdong-Hong Kong-Macao Greater Bay Area Meteorological Technology Collaborative Research Project under Grant GHMA2024Z04, in part by Fundamental Research Funds for the Central Universities, Sun Yat-sen University under Grant 23hytd006, and in part by Guangdong Provincial High-Level Young Talent Program under Grant RL2024-151-2-11.

References

[1] Moloud Abdar, Farhad Pourpanah, Sadiq Hussain, Dana Rezazadegan, Li Liu, Mohammad Ghavamzadeh, Paul Fieguth, Xiaochun Cao, Abbas Khosravi, U. Rajendra Acharya, Vladimir Makarenkov, and Saeid Nahavandi. A review of uncertainty quantification in deep learning: Techniques, applications and challenges. *Information Fusion*, 76:243–297, December 2021.

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[3] David Barrett, Felix Hill, Adam Santoro, Ari Morcos, and Timothy Lillicrap. Measuring abstract reasoning in neural networks. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 511–520. PMLR, 10–15 Jul 2018.

[4] Lucian Busoniu, Radoslav Babuska, and Bart De Schutter. Multi-agent reinforcement learning: A survey. `https://ieeexplore.ieee.org/document/4150194`, 2008. Accessed: 2025-05-15.

[5] Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. Chateval: Towards better llm-based evaluators through multi-agent debate, 2023.

[6] Fei-Long Chen, Du-Zhen Zhang, Ming-Lun Han, Xiu-Yi Chen, Jing Shi, Shuang Xu, and Bo Xu. Vlp: A survey on vision-language pre-training. *Machine Intelligence Research*, 20(1):38–56, January 2023.

[7] Abhishek Das, Satwik Kottur, José M. F. Moura, Stefan Lee, and Dhruv Batra. Learning cooperative visual dialog agents with deep reinforcement learning. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2970–2979, 2017.

[8] Ali Dorri, Salil S. Kanhere, and Raja Jurdak. Multi-agent systems: A survey. *IEEE Access*, 6:28573–28593, 2018.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[10] Yifan Du, Zikang Liu, Junyi Li, and Wayne Xin Zhao. A survey of vision-language pre-trained models, 2022.

[11] Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate, 2023.

[12] Mohammed Elhenawy, Ahmad Abutahoun, Taqwa I. Alhadidi, Ahmed Jaber, Huthaifa I. Ashqar, Shadi Jaradat, Ahmed Abdelhay, Sebastien Glaser, and Andry Rakotonirainy. Visual reasoning and multi-agent approach in multimodal large language models (mllms): Solving tsp and mtsp combinatorial challenges, 2024.

[13] Ghada M. Elshamy, Marco Alfonse, and Mostafa M. Aref. A guesswhat?! game for goal-oriented visual dialog: A survey. In *2021 Tenth International Conference on Intelligent Computing and Information Systems (ICICIS)*, pages 116–123, 2021.

[14] Seth Frey, Jules Hedges, Joshua Tan, and Philipp Zahn. Composing games into complex institutions, 2023.

[15] Zijian Gao, Kele Xu, Bo Ding, and Huaimin Wang. Knowru: Knowledge reuse via knowledge distillation in multi-agent reinforcement learning. *Entropy*, 23(8):1043, August 2021.

[16] Jakob Gawlikowski, Cedrique Rovile Njieutcheu Tassi, Mohsin Ali, Jongseok Lee, Matthias Humt, Jianxiang Feng, Anna Kruspe, Rudolph Triebel, Peter Jung, Ribana Roscher, Muhammad Shahzad, Wen Yang, Richard Bamler, and Xiao Xiang Zhu. A survey of uncertainty in deep neural networks, 2022.

[17] Simon Ging, Mohammadreza Zolfaghari, Hamed Pirsiavash, and Thomas Brox. Coot: Cooperative hierarchical transformer for video-text representation learning, 2020.

[18] Sven Gronauer and Klaus Diepold. Multi-agent deep reinforcement learning: a survey. *Artif. Intell. Rev.*, 55(2):895–943, February 2022.

[19] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[20] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor, 2018.

[21] Frank Heinemann, Rosemarie Nagel, and Peter Ockenfels. Measuring strategic uncertainty in coordination games. *The Review of Economic Studies*, 76(1):181–221, 2009.

[22] Jen-Hao Hsiao and Kwang-Cheng Chen. Communication methodology to control a distributed multi-agent system. In *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pages 1–6, 2019.

[23] Drew A. Hudson and Christopher D. Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6693–6702, 2019.

[24] Dom Huh and Prasant Mohapatra. Multi-agent reinforcement learning: A comprehensive survey, 2024.

[25] Akbir Khan, John Hughes, Dan Valentine, Laura Ruis, Kshitij Sachan, Ansh Radhakrishnan, Edward Grefenstette, Samuel R. Bowman, Tim Rocktäschel, and Ethan Perez. Debating with more persuasive llms leads to more truthful answers. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

[26] Mathieu Laurière, Sarah Perrin, Julien Pérolat, Sertan Girgin, Paul Muller, Romuald Élie, Matthieu Geist, and Olivier Pietquin. Learning in mean field games: A survey, 2024.

[27] Kenton Lee, Mandar Joshi, Iulia Turc, Hexiang Hu, Fangyu Liu, Julian Eisenschlos, Urvashi Khandelwal, Peter Shaw, Ming-Wei Chang, and Kristina Toutanova. Pix2struct: Screenshot parsing as pretraining for visual language understanding, 2023.

[28] Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. Camel: Communicative agents for "mind" exploration of large language model society, 2023.

[29] Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. Encouraging divergent thinking in large language models through multi-agent debate. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17889–17904, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[30] Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Generating with confidence: Uncertainty quantification for black-box large language models, 2024.

[31] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

[32] Yexiang Liu, Jie Cao, Zekun Li, Ran He, and Tieniu Tan. Breaking mental set to improve reasoning through diverse multi-agent debate. In *The Thirteenth International Conference on Learning Representations*, 2025.

[33] Yikang Liu, Ziyin Zhang, Wanyang Zhang, Shisen Yue, Xiaojing Zhao, Xinyuan Cheng, Yiwen Zhang, and Hai Hu. Argugpt: evaluating, understanding and identifying argumentative essays generated by gpt models, 2023.

[34] Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. Mmbench: Is your multi-modal model an all-around player?, 2024.

[35] Guiyang Luo, Hui Zhang, Haibo He, Jinglin Li, and Fei-Yue Wang. Multiagent adversarial collaborative learning via mean-field theory. *IEEE Transactions on Cybernetics*, 51(10):4994–5007, 2021.

[36] Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. Ocr-vqa: Visual question answering by reading text in images. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 947–952, 2019.

[37] John F. Nash. Equilibrium points in n-person games, 1950.

[38] Masanao Ozawa. Heisenberg's original derivation of the uncertainty principle and its universally valid reformulations, 2020.

[39] Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior, 2023.

[40] Joon Sung Park, Carolyn Q. Zou, Aaron Shaw, Benjamin Mako Hill, Carrie Cai, Meredith Ringel Morris, Robb Willer, Percy Liang, and Michael S. Bernstein. Generative agent simulations of 1,000 people, 2024.

[41] Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025.

[42] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[43] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9, 2019.

[44] Rahul Rahaman and Alexandre H. Thiery. Uncertainty quantification and deep ensembles. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA, 2021. Curran Associates Inc.

[45] James K. Rilling and Alan G. Sanfey. Social decision-making: insights from game theory and neuroscience. *Trends in cognitive sciences*, 15(5):216–224, 2011.

[46] Leonard Salewski, Stephan Alaniz, Isabel Rio-Torto, Eric Schulz, and Zeynep Akata. In-context impersonation reveals large language models' strengths and biases. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

[47] Alan G. Sanfey, James K. Rilling, Jennifer A. Aronson, Leigh E. Nystrom, and Jonathan D. Cohen. The neural basis of economic decision-making in the ultimatum game. *Science*, 300(5626):1755–1758, 2003.

[48] Jinzhou Tang, Jusheng Zhang, Qinhan Lv, Sidi Liu, Jing Yang, Chengpei Tang, and Keze Wang. Hiva: Self-organized hierarchical variable agent via goal-driven semantic-topological evolution, 2025.

[49] Gemini Team. Gemini: A family of highly capable multimodal models, 2025.

[50] Hugo Touvron and Louis Martin. Llama 2: Open foundation and fine-tuned chat models, 2023.

[51] Khanh-Tung Tran, Dung Dao, Minh-Duong Nguyen, Quoc-Viet Pham, Barry O'Sullivan, and Hoang D. Nguyen. Multi-agent collaboration mechanisms: A survey of llms, 2025.

[52] Wen-Kwang Tsao. Multi-agent reasoning with large language models for effective corporate planning. In *2023 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 365–370, 2023.

[53] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

[54] Jin Wang, Shichao Dong, Yapeng Zhu, Kelu Yao, Weidong Zhao, Chao Li, and Ping Luo. Diagnosing the compositional knowledge of vision language models from a game-theoretic view, 2024.

[55] Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Jirong Wen. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6), March 2024.

[56] Qineng Wang, Zihao Wang, Ying Su, Hanghang Tong, and Yangqiu Song. Rethinking the bounds of LLM reasoning: Are multi-agent discussions the key? In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6106–6131, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[57] Shenzhi Wang, Chang Liu, Zilong Zheng, Siyuan Qi, Shuo Chen, Qisen Yang, Andrew Zhao, Chaofei Wang, Shiji Song, and Gao Huang. Avalon's game of thoughts: Battle against deception through recursive contemplation, 2023.

[58] X. J. Wang, Y. Q. Li, Z. Y. Chen, et al. Neurocomputational mechanisms involved in adaptation to fluctuating intentions of others. *Nature Communications*, 15(1):1234, 2024.

[59] 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, 2023.

[60] Y. Wang, J. Gao, H. Li, et al. Large language models empowered agent-based modeling and simulation: a survey and perspectives, 2024.

[61] Fanghua Ye, Mingming Yang, Jianhui Pang, Longyue Wang, Derek F. Wong, Emine Yilmaz, Shuming Shi, and Zhaopeng Tu. Benchmarking llms via uncertainty quantification, 2024.

[62] Luke Yoffe, Alfonso Amayuelas, and William Yang Wang. Debunc: Improving large language model agent communication with uncertainty metrics, 2025.

[63] Chao Yu, Xinyi Yang, Jiaxuan Gao, Huazhong Yang, Yu Wang, and Yi Wu. Learning efficient multi-agent cooperative visual exploration, 2022.

[64] Xiang Yue, Yuansheng Ni, Tianyu Zheng, Kai Zhang, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and Wenhu Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9556–9567, 2024.

[65] Andy Zeng, Maria Attarian, Brian Ichter, Krzysztof Choromanski, Adrian Wong, Stefan Welker, Federico Tombari, Aveek Purohit, Michael Ryoo, Vikas Sindhwani, Johnny Lee, Vincent Vanhoucke, and Pete Florence. Socratic models: Composing zero-shot multimodal reasoning with language, 2022.

[66] Chongjie Zhang and Victor Lesser. Multi-agent learning with policy prediction. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, AAAI'10, page 927–934. AAAI Press, 2010.

[67] Jusheng Zhang, Kaitong Cai, Yijia Fan, Ningyuan Liu, and Keze Wang. Mat-agent: Adaptive multi-agent training optimization, 2025.

[68] Jusheng Zhang, Kaitong Cai, Yijia Fan, Jian Wang, and Keze Wang. Cf-vlm:counterfactual vision-language fine-tuning, 2025.

[69] Jusheng Zhang, Kaitong Cai, Xiaoyang Guo, Sidi Liu, Qinhan Lv, Ruiqi Chen, Jing Yang, Yijia Fan, Xiaofei Sun, Jian Wang, Ziliang Chen, Liang Lin, and Keze Wang. Mm-cot:a benchmark for probing visual chain-of-thought reasoning in multimodal models, 2025.

[70] Jusheng Zhang, Kaitong Cai, Jing Yang, and Keze Wang. Learning dynamics of vlm finetuning, 2025.

[71] Jusheng Zhang, Kaitong Cai, Qinglin Zeng, Ningyuan Liu, Stephen Fan, Ziliang Chen, and Keze Wang. Failure-driven workflow refinement, 2025.

[72] Jusheng Zhang, Yijia Fan, Kaitong Cai, Zimeng Huang, Xiaofei Sun, Jian Wang, Chengpei Tang, and Keze Wang. Drdiff: Dynamic routing diffusion with hierarchical attention for breaking the efficiency-quality trade-off, 2025.

[73] Jusheng Zhang, Yijia Fan, Kaitong Cai, Jing Yang, Jiawei Yao, Jian Wang, Guanlong Qu, Ziliang Chen, and Keze Wang. Why keep your doubts to yourself? trading visual uncertainties in multi-agent bandit systems, 2026.

[74] Jusheng Zhang, Zimeng Huang, Yijia Fan, Ningyuan Liu, Mingyan Li, Zhuojie Yang, Jiawei Yao, Jian Wang, and Keze Wang. KABB: Knowledge-aware bayesian bandits for dynamic expert coordination in multi-agent systems. In *Forty-second International Conference on Machine Learning*, 2025.

[75] Kaiqing Zhang, Zhuoran Yang, and Tamer Başar. Multi-agent reinforcement learning: A selective overview of theories and algorithms, 2021.

[76] Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu, Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, Hao Li, Jiahao Wang, Nianchen Deng, Songze Li, Yinan He, Tan Jiang, Jiapeng Luo, Yi Wang, Conghui He, Botian Shi, Xingcheng Zhang, Wenqi Shao, Junjun He, Yingtong Xiong, Wenwen Qu, Peng Sun, Penglong Jiao, Han Lv, Lijun Wu, Kaipeng Zhang, Huipeng Deng, Jiaye Ge, Kai Chen, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models, 2025.

Supplementary Material

Supplementary Material Overview

The supplementary material provides detailed experimental setups, results, theoretical analyses, and case studies to support the main paper's claims on the GAM-Agent framework. It includes cost-performance analyses, uncertainty handling evaluations, theoretical derivations, hyperparameter studies, prompt configurations, and case analyses to demonstrate the framework's robustness and limitations.

Contents

A Cost and Performance Balance Analysis

A.1 Experiment Setup

To evaluate the cost-performance balance of our proposed approach, we conducted a comparative analysis on the MMbench_TEST_V11 benchmark. We measured the "Cost per Instruction on MMbench_TEST_V11" and "Overall ACC (%)" for several leading Vision-Language Models (VLMs) and configurations derived from our framework (denoted as "Ours"). The commercial models benchmarked include Qwen2.5VL[41] series (7B, 72B via API), Gemini 2.5 Pro Preview[49], Gemini 2.0 Pro, GPT-4o[2], and Claude 3.7, with their respective costs per instruction sourced from OpenRouter API pricing. For our configurations, leveraging models such as Qwen2.5VL (7B, 72B) and InternVL3 (14B, 78B)[76], the cost per instruction was calculated based on the inference time on a local NVIDIA A100 GPU. This local GPU cost was estimated using the on-demand price of an AWS p4d.24xlarge instance, which includes 8 A100 40GB GPUs at approximately $32.77 per hour, translating to about $4.10 per hour for a single A100 GPU. This setup allows for a direct comparison of the economic efficiency of deploying VLMs via commercial APIs versus utilizing them within our optimized framework on local hardware.

A.2 Experiment Result

The results of our cost and performance comparison on the MMbench_TEST_V11 benchmark are summarized in Figure 5. The experimental results presented in Figure 5 reveal significant disparities in cost-effectiveness across the evaluated models. Notably, our framework's configurations demonstrate exceptional performance with compelling cost benefits. InternVL3-78B (Ours) achieved the highest accuracy of 92.2% with a competitive cost of approximately $\$0.00022$ per instruction. Closely following, Qwen2.5VL-72B (Ours) (one configuration) recorded 92.0% accuracy at a cost of $\$0.00020$ per instruction. Interestingly, another configuration of Qwen2.5VL-72B deployed with our framework (labeled as "Qwen2.5VL-72B" on the "Ours" trend line in Figure 5) achieved 90.4% accuracy at an even lower cost of $\$0.00011$ per instruction. This particular configuration not only surpasses the Qwen2.5VL-72B API (88.4% ACC at $\$0.00015$) in accuracy but is also approximately 1.36 times more cost-effective. Other configurations, such as InternVL3-14B (Ours), delivered 91.4% accuracy at $\$0.00016$ per instruction, while Qwen2.5VL-7B (Ours) achieved a strong 89.0% accuracy at a cost of $\$0.000085$ per instruction. This is a notable improvement in accuracy over its API counterpart (Qwen2.5VL-7B API, 82.6% ACC at $\$0.00007$), albeit at a slightly higher local inference cost.

While Gemini 2.5 Pro Preview (API) exhibited high accuracy (89.3%), its API cost ($\$0.00075$) is considerably higher than our top-performing local configurations. For instance, InternVL3-78B (Ours) and the higher-performing Qwen2.5VL-72B (Ours) achieve superior accuracy (92.2% and 92.0%, respectively) and are approximately 3.4 to 3.75 times less expensive than Gemini 2.5 Pro Preview. In contrast, widely used models like GPT-4o (API) and Claude 3.7 (API), despite their capabilities, incurred substantially higher costs per instruction ($\$0.015$ and $\$0.018$, respectively) for accuracies of 84.3% and 81.2% on this benchmark, accuracies which are surpassed by several of our framework's configurations at a fraction of the cost.

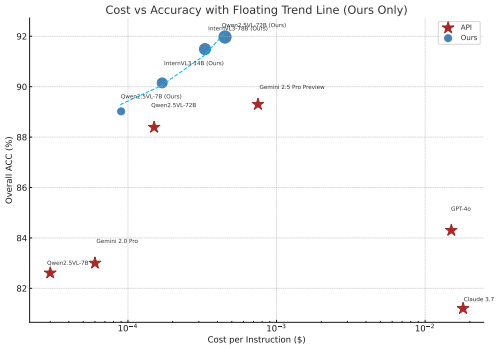


Figure 5: Comparison of Cost per Instruction and Overall Accuracy on MMbench_TEST_V11. "Ours" denotes models run with our proposed framework and local inference cost calculation.

This analysis underscores the substantial economic advantages of leveraging optimized local inference strategies for state-of-the-art open-source VLMs. The results indicate that our approach can achieve or even surpass the accuracy of expensive commercial API-based services at significantly reduced costs. For example, our InternVL3-78B (Ours) configuration not only outperforms Gemini 2.5 Pro Preview in accuracy (92.2% vs. 89.3%) but does so at approximately $1/3.4$th of the cost per instruction. Similarly, the Qwen2.5VL-72B

(Ours) configuration achieving 92.0% accuracy does so at about $1/3.75$th of the cost of Gemini 2.5 Pro Preview, while also delivering higher accuracy. This highlights a pathway to democratize access to high-performance VLM capabilities, making advanced visual reasoning more accessible and economically viable. The significant cost reductions and performance enhancements observed for models like Qwen2.5VL and InternVL3 when run under our framework suggest that efficient utilization of local compute resources can unlock dramatic improvements in cost-efficiency without compromising.

B Experiment on Cost Dynamics During Multi-Round Debates

To investigate the economic implications of iterative debate, particularly how computational costs evolve across multiple rounds, we conducted a dedicated experiment. This analysis aims to compare the cumulative cost of our GAM-Agent framework against a baseline multi-agent debate scenario where original models engage in self-correction or iterative refinement without GAM-Agent's structured uncertainty-driven mechanisms. Understanding these cost dynamics is crucial for assessing the practical viability and efficiency of multi-round debate strategies.

B.1 Experimental Setup

The experiments were performed on the MMBench_V11_Test benchmark. We selected four prominent open-source Vision-Language Models (VLMs) as base models: Qwen2.5VL-7B, Qwen2.5VL-72B, InternVL3-14B, and InternVL3-78B.

We compared two primary configurations:

1. **GAM-Agent Framework:** Our proposed GAM-Agent framework was applied using each of the selected base VLMs. The setup involved $N = 3$ expert agents and $N_{crit} = 3$ critic agents, consistent with other experiments.
2. **Baseline Multi-Round Self-Debate (Base-Debate):** For this configuration, we simulated a multi-round debate scenario where instances of the original base VLM were prompted to iteratively refine their answers based on their previous outputs or simple peer feedback, without the structured uncertainty quantification, evidence mapping, or dynamic weighting of GAM-Agent. This represents a more direct or naive approach to multi-round refinement.

For both configurations, we tracked the cumulative computational cost after 1, 2, and 3 rounds of debate. The "cost" was quantified by the average number of tokens processed per instance (input prompt tokens + generated output tokens) at the conclusion of each specified round. This metric serves as a proxy for both local GPU computation time and potential API call expenses. The primary goal of this experiment is to elucidate how GAM-Agent manages cost accumulation over successive debate rounds compared to a less structured iterative debate, and to highlight the efficiency benefits derived from its consensus mechanism which aims for early convergence.

B.2 Experimental Results

The average cumulative costs (in tokens per instance) for the GAM-Agent framework and the Baseline Multi-Round Self-Debate (Base-Debate) configuration across 1, 2, and 3 rounds of debate are presented in Figure 6.

Figure 6 illustrates that while costs for GAM-Agent increase with more debate rounds, this is accompanied by significant accuracy improvements that consistently outperform the single-agent baseline. For instance, with InternVL3-78B as the base, GAM-Agent's average cost per instruction after 3 rounds is \$0.00033, at which point it achieves an accuracy of 91.90%. In contrast, the Single-agent (Base-Debate) configuration with InternVL3-78B reaches an accuracy of 90.20% after 3 rounds. GAM-Agent achieves superior accuracy (e.g., +1.7 percentage points for InternVL3-78B after 3 rounds) for a defined number of debate cycles. This trend is consistent across other models. With Qwen2.5VL-72B, GAM-Agent achieves 91.82% accuracy at a cost of \$0.00045 after 3 rounds, while the single-agent baseline reaches 90.20%. For Qwen2.5VL-7B, GAM-Agent reaches 89.02% accuracy for \$0.00009 after 3 rounds, compared to the baseline's 86.50%. Similarly, for InternVL3-14B, GAM-Agent achieves 90.15% for \$0.00017 after 3 rounds, surpassing the baseline's

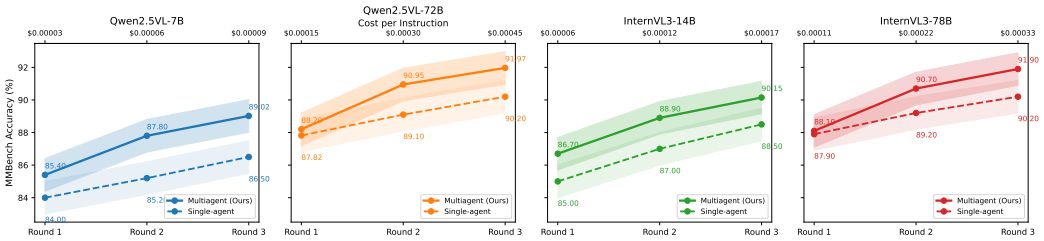


Figure 6: Average cumulative cost (tokens per instance) on MMBench_V11_Test after 1, 2, and 3 rounds of debate for GAM-Agent (utilizing different base models) versus Baseline Multi-Round Self-Debate (Base-Debate).

88.50%. The GAM-Agent's structured debate, driven by uncertainty and conflict detection, aims to terminate debates earlier if consensus is reached, contributing to an optimized balance between cost and performance. While the Baseline Self-Debate might also incur increasing costs with more rounds, it consistently achieves lower accuracy. This suggests that to reach the accuracy levels of GAM-Agent, the baseline would likely require even more rounds, leading to significantly higher cumulative costs, or may not reach such accuracy levels at all within a comparable number of iterations. Thus, GAM-Agent's mechanisms not only improve accuracy but also offer a more efficient pathway to achieving high performance in multi-round visual reasoning by strategically focusing the debate and adaptively managing its length and cost.

C Deepening Comparative Analysis of Multi-Agent Methods: Uncertainty Handling Capability

To comprehensively evaluate the uncertainty handling capabilities of our proposed GAM-Agent framework across various base models, we extended our analysis to include all major models from the Qwen2.5VL series (3B to 72B) and the InternVL3 series (2B to 78B). This allows us to observe the relationship between uncertainty processing capabilities and model scale, as well as the performance of GAM-Agent across different architectures.

C.1 Comprehensive Analysis of Uncertainty Handling Capability

We continue to use three key metrics to evaluate uncertainty-handling capabilities:

- **Uncertainty Accuracy (UA↑)**: GAM-Agent's ability to correctly identify instances of high uncertainty (higher is better).
- **Calibration Error (CE↓)**: The consistency between the model's predicted confidence and its actual accuracy (lower is better).
- **Dynamic Adaptability (DA↑)**: The ability to adjust the decision-making process based on uncertainty (higher is better).

Figures 7(Qwen2.5 VL series) and (InternVL3 series) show the performance of different multi-agent methods on these metrics and their MMBench accuracy.

C.2 Experimental Setup

The evaluation of uncertainty handling capabilities (UA, CE, DA) was conducted on the MMBench[34] dataset, which is also used for reporting the final accuracy. The primary motivation for this extended analysis is to systematically assess how different multi-agent frameworks, including our GAM-Agent, manage and leverage uncertainty, and how these capabilities scale with the underlying VLM's size and architecture.

For each base model in the Qwen2.5VL series (3B, 7B, 32B, 72B) and InternVL3 series (2B, 8B, 38B, 78B), we applied GAM-Agent and the baseline multi-agent methods (DMAD, DMPL, ChatEval, MAD, DebUnc). The baseline methods were adapted to the visual question answering task on MMBench The metrics were measured as follows:

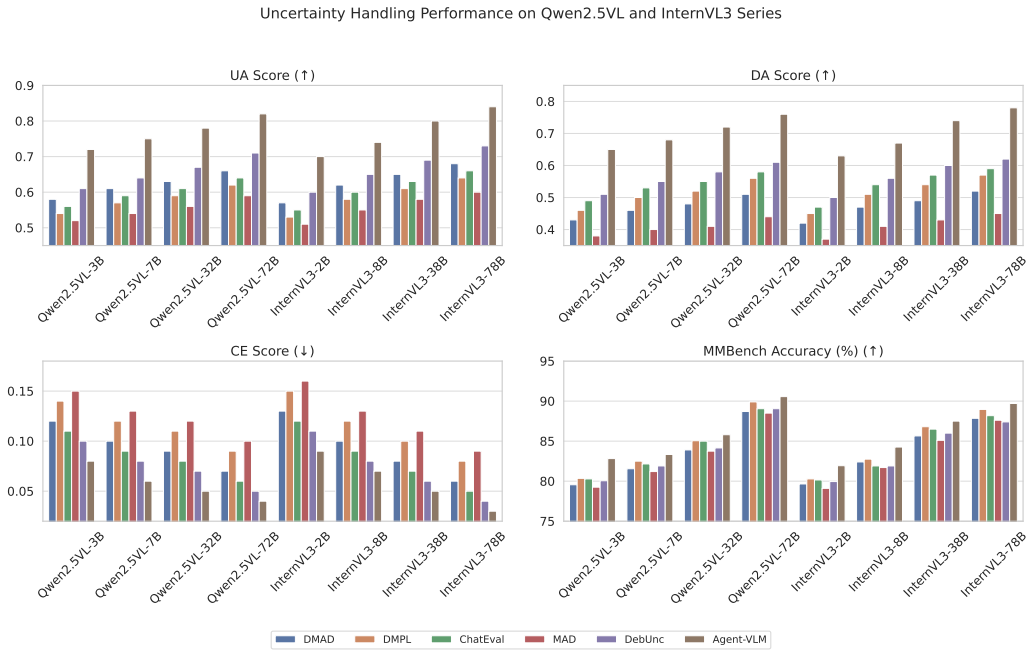


Figure 7: Comparison of uncertainty handling performance for various methods on the Qwen2.5VL and InternVL3 model series. UA, DA higher is better; CE lower is better.

- **Uncertainty Accuracy (UA)**: This metric was evaluated by defining a threshold for high system-level uncertainty (e.g., based on entropy or semantic cues from the aggregated response). We then assessed the proportion of instances correctly flagged as uncertain (e.g., where the model's prediction was incorrect or the question was inherently ambiguous) against a ground truth or heuristic for uncertainty.
- **Calibration Error (CE)**: We utilized an Expected Calibration Error (ECE) variant. Model responses were binned by their expressed confidence scores (derived from uncertainty metrics like $1 - U_{sys}$). The ECE was then calculated as the weighted average difference between the mean confidence and observed accuracy within each bin.
- **Dynamic Adaptability (DA)**: This was qualitatively and quantitatively assessed by observing whether the framework modified its reasoning process or resource allocation (e.g., triggering debate rounds in GAM-Agent, or adjusting agent weighting) in response to detected uncertainty, and whether such adaptations led to improved outcomes or more cautious predictions in uncertain scenarios. For a quantitative proxy, we measured the correlation between VLM's internal uncertainty signals and its engagement of deeper reasoning mechanisms or confidence adjustments.

The core idea is to investigate whether sophisticated uncertainty modeling, as implemented in GAM-Agent, translates into more reliable and robust performance, especially as model complexity and task difficulty increase.

C.3 Experimental Results and Analysis

The results presented in Figure 7 offer a detailed view of the uncertainty handling capabilities of different multi-agent frameworks.

GAM-Agent Consistently Excels in Uncertainty Management: Across both the Qwen2.5VL and InternVL3 series, and for all model sizes, GAM-Agent consistently outperforms the baseline methods (DMAD, DMPL, ChatEval, MAD, DebUnc) on all three uncertainty metrics: Uncertainty Accuracy (UA), Calibration Error (CE), and Dynamic Adaptability (DA). For example, with Qwen2.5VL-72B, GAM-Agent achieves a UA of 0.82, CE of 0.04, and DA of 0.76, significantly surpassing other approaches. A similar trend is observed with InternVL3-78B, where GAM-Agent scores 0.84

(UA), 0.03 (CE), and 0.78 (DA). This suggests that GAM-Agent's explicit uncertainty quantification, evidence mapping, and uncertainty-driven debate mechanisms are highly effective.

Scalability with Model Size: A key observation is the positive scaling of GAM-Agent's uncertainty handling capabilities with increasing model size. For the Qwen2.5VL series, as the model size increases from 3B to 72B, GAM-Agent's UA improves from 0.72 to 0.82, CE reduces from 0.08 to 0.04, and DA increases from 0.65 to 0.76. Similarly, for the InternVL3 series (2B to 78B), GAM-Agent's UA rises from 0.70 to 0.84, CE drops from 0.09 to 0.03, and DA improves from 0.63 to 0.78. This indicates that larger base models, when integrated into the GAM-Agent framework, not only provide better raw capabilities but also become more adept at identifying, calibrating, and adapting to uncertainty. While baseline methods also show some improvement with model size, their gains in uncertainty metrics are generally less pronounced than those achieved by GAM-Agent.

Correlation with MMBench Accuracy: The data strongly suggests a positive correlation between superior uncertainty handling and higher overall task performance on MMBench. GAM-Agent, which consistently leads in UA, CE, and DA, also achieves the highest MMBench accuracy for each respective base model. For instance, GAM-Agent with Qwen2.5VL-72B not only has the best uncertainty scores but also the top MMBench accuracy of 90.56% in Figure 7. This underscores the importance of robust uncertainty management for achieving reliable and accurate visual reasoning. Frameworks that can better understand their own limitations and adapt accordingly are more likely to succeed in complex tasks.

Performance of Baseline Methods: The baseline methods exhibit varied performance in uncertainty handling. DebUnc, which also focuses on uncertainty, generally performs second best to GAM-Agent on uncertainty metrics but still lags significantly. Other methods like DMAD, DMPL, ChatEval, and MAD show more modest capabilities in these specific uncertainty-related assessments. Their mechanisms for collaboration may not explicitly or effectively propagate and utilize fine-grained uncertainty signals to the same extent as GAM-Agent.

In conclusion, this expanded comparative analysis demonstrates GAM-Agent's superior and scalable uncertainty-handling capabilities. Its architecture, designed to explicitly model and debate uncertainty, enables it to achieve better calibration, more accurate identification of challenging instances, and more effective adaptation of its reasoning processes. These strengths in managing uncertainty are critical contributors to its leading performance on complex visual reasoning benchmarks like MMBench.

C.4 Experimental Analysis

The experimental results strongly confirm the effectiveness and robustness of the proposed debate termination conditions. As shown in Figure 8 (your Figure 1, which you indicated is 'boyi1.pdf'), all tested models exhibited a consistent accuracy distribution pattern across the various tasks in MMBench. Specifically, large-scale models (Qwen2.5VL-72B, InternVL3-78B) generally outperformed small-scale models (Qwen2.5VL-7B, InternVL3-14B) in terms of accuracy. When comparing models of similar scale, the InternVL3 series demonstrated a slight performance advantage over the Qwen2.5VL series. Regarding task types, text understanding tasks typically achieved the highest accuracy, while relational reasoning problems proved to be relatively more challenging.

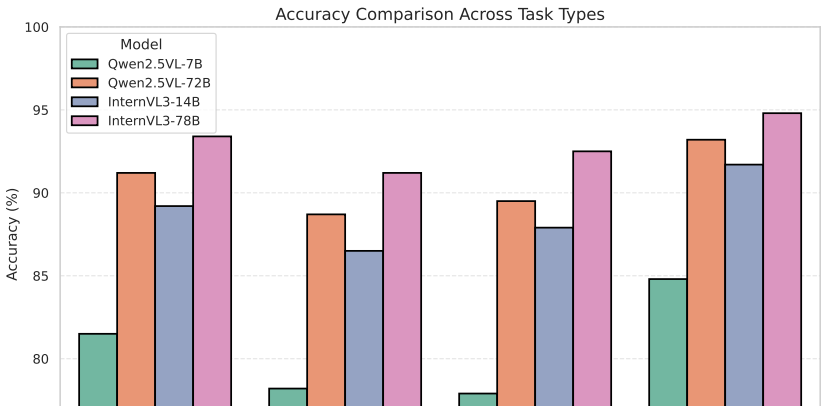


Figure 8: Performance and convergence dynamics under default termination conditions. This figure shows model accuracy across MMBench tasks, general performance trends, and key debate convergence statistics like average rounds and primary termination reasons.

Key experimental findings are summarized below:**Natural Convergence and Efficiency**: Under the default termination conditions, a high proportion of samples, ranging from 88% to 94%, achieved natural convergence before reaching the preset maximum number of debate rounds ($K_{\max} = 10$). On average, GAM-Agent required only 2.1 to 2.6 debate rounds to converge, with large-scale models exhibiting faster convergence. **Distribution of Termination Reasons**: Triggering of the uncertainty threshold (i.e., system uncertainty $U_{\text{sys}}^{(k)} < \theta_{U,\text{term}}$) was the primary reason for debate termination, accounting for 63% to 72% of cases. The convergence speed threshold (i.e., change in system uncertainty $|\Delta U_{\text{sys}}^{(k)}| < \epsilon$) was the second most common reason, responsible for approximately 22% to 25% of terminations. Only a minority of instances terminated due to reaching the maximum number of debate rounds. **Task-Dependent Convergence Speed**: The convergence speed varied across different task types. Text understanding problems typically converged the fastest, requiring an average of only 1.7 to 2.1 debate rounds. In contrast, relational reasoning problems, owing to their complexity, converged relatively slowest, needing an average of 2.6 to 3.2 debate rounds. Figure 9 (your Figure 2) further illustrates the impact of the uncertainty termination threshold $\theta_{U,term}$ on model accuracy and the average number of debate rounds.

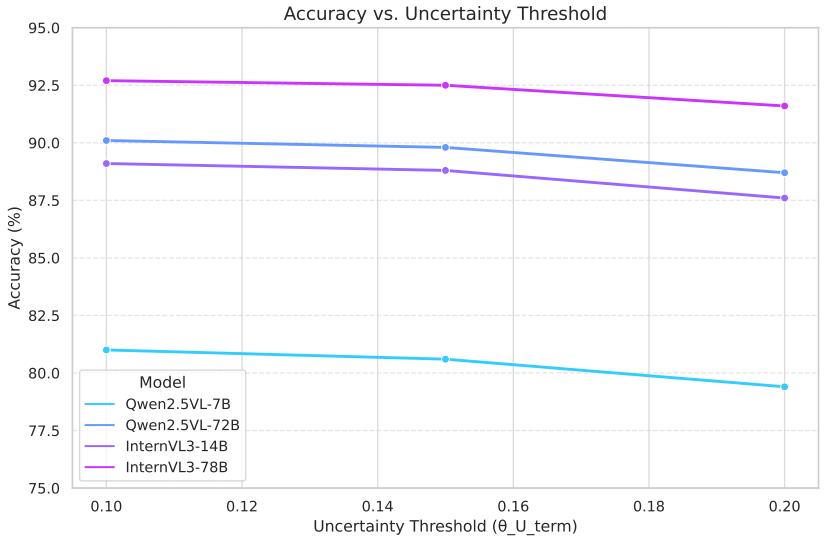


Figure 9: Sensitivity analysis of the uncertainty termination threshold ($\theta_{U,term}$). This figure illustrates the trade-off between prediction accuracy and average debate rounds when varying $\theta_{U,term}$ (0.10, 0.15, 0.20), with $\theta_{U,term} = 0.15$ identified as an optimal balance.

When $\theta_{U,term}$ was set to 0.10 (a stricter convergence requirement), there was a slight improvement in accuracy (approximately +0.2% to +0.4%), but the average number of debate rounds increased significantly (by about 25% to 31%), leading to higher computational costs. Conversely, when $\theta_{U,term}$ was set to 0.20 (a looser convergence requirement), the average number of debate rounds decreased (by about 22% to 26%), but accuracy also declined (by approximately -0.9% to -1.2%). The experimental data indicate that all tested models achieve an optimal balance between prediction accuracy and computational efficiency at the default setting of $\theta_{U,term} = 0.15$. Notably, large-scale models (72B/78B parameter scale) demonstrated lower sensitivity to variations in the $\theta_{U,term}$ parameter, suggesting greater robustness under different termination conditions. In summary, these experimental results clearly demonstrate that our designed debate termination mechanism can effectively balance prediction accuracy and computational efficiency across vision language models of different scales and architectures. This provides reliable parameter setting guidance and performance expectations for the practical deployment of the GAM-Agent framework.

C.5 Theoretical Analysis of Termination Threshold $\theta_{U,term}$ in Multi-Agent Debate

The termination criterion in GAM-Agent's multi-agent debate framework plays a pivotal role in balancing system performance and computational efficiency. Drawing on our main experimental observations, this section provides a theoretical dissection of how the core uncertainty threshold parameter, $\theta_{U,term}$, influences the debate process. Three principal conditions govern the debate termination mechanism (see Section 3.5.1 and Equation 5): (1) the system uncertainty $U_{sys}^{(k)}$ drops below the pre-set threshold $\theta_{U,term}$; (2) the debate reaches the maximum number of allowed iterations K_{max}; (3) the change in system uncertainty between consecutive rounds $|\Delta U_{sys}^{(k)}|$ falls below a convergence threshold ϵ. Among these, $\theta_{U,term}$ theoretically defines the confidence requirement for terminating deliberation. A deeper investigation into the $\theta_{U,term}$ parameter reveals its multifaceted role. From a Bayesian decision theory perspective, $\theta_{U,term}$ establishes a decision boundary, aiming to balance Type I errors (e.g., prematurely accepting a sub-optimal consensus) against Type II errors (e.g., unnecessarily prolonging the debate). Our primary experimental observations (see Section 4 and Appendix D.4) further expose several key convergence characteristics and sensitivities: First, the system uncertainty $U_{sys}^{(k)}$ typically decays in an approximately exponential manner during debate; Second, models with larger parameter scales (e.g., 72B/78B) exhibit lower sensitivity to changes in $\theta_{U,term}$, indicating more robust uncertainty estimation in larger models; Third, more complex tasks such as relational reasoning display higher sensitivity to the setting of $\theta_{U,term}$ compared to simpler tasks like text comprehension. Despite the effectiveness of the current termination mechanism, several theoretical limitations[22] remain: (1) adopting a single global $\theta_{U,term}$ for all task types may not be optimal; (2) the mechanism does not dynamically adjust $\theta_{U,term}$ based on instance-specific factors (e.g., problem difficulty); (3) employing a fixed $\theta_{U,term}$ throughout the debate does not account for the dynamic evolution of uncertainty. Based on the above theoretical analysis and main experimental findings, we offer the following theoretical insights for setting $\theta_{U,term}$: (1) Reasonable upper and lower bounds should be established—an excessively low $\theta_{U,term}$ may lead to unnecessary debate rounds and diminishing marginal returns, while an excessively high threshold risks premature termination, undermining the benefits of iterative deliberation; (2) Model-adaptive setting—in theory, as model capability increases, the optimal $\theta_{U,term}$ should decrease accordingly; (3) Task-adaptive setting—the ideal implementation should adjust $\theta_{U,term}$ based on task type, assigning stricter thresholds for more complex tasks

D Derivation of Uncertainty Theory

To provide a rigorous theoretical foundation for the collaborative reasoning process within the GAM-Agent framework, particularly concerning the role of uncertainty and dynamic consensus building[38], we model the interaction among expert agents as a consensus-driven non-zero-sum game. This game-theoretic perspective elucidates how agents converge towards a final consensus through interaction and dynamic adjustment of uncertainty. Empirically, we observe that the uncertainty of predictions decays exponentially with debate rounds, which we quantify using a decay coefficient λ. As shown in Figure X, larger models such as InternVL3-78B exhibit higher λ values and faster per-round uncertainty suppression (e.g., $\lambda = 0.51$ with a 39.9% decay rate), supporting the view that iterative multiagent consensus not only improves performance but also systematically reduces epistemic uncertainty.

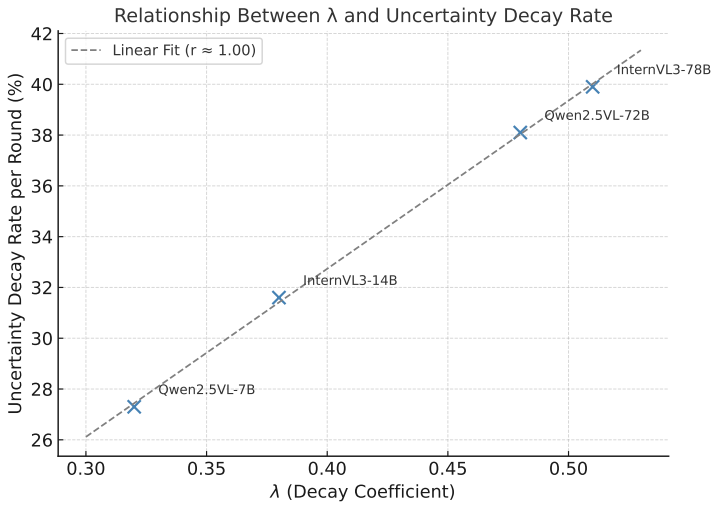


Figure 10: Exponential decay of uncertainty with debate rounds across models. Larger models (e.g., InternVL3-78B) exhibit faster uncertainty suppression.

D.1 Mathematical Formalization of the Non-Zero-Sum Game

We consider a multi-agent system comprising N expert agents. The system processes an input instance I, which consists of an image X and a question P_r. The input space is formalized as the Cartesian product of the image space $\mathcal{X}$ and the question space $\mathcal{P}_r$, denoted as $\mathcal{I} = \mathcal{X} \times \mathcal{P}_r$. A specific input instance is $I = (X, P_r) \in \mathcal{I}$. **Agent Set ($\mathcal{E}$):** The participants in the system form the set of agents, denoted as $\mathcal{E} = \{1, 2, \ldots, N\}$. This corresponds to the set E in the GAM-Agent framework definition.

Agent Function (Agent Mapping A_i): Each agent $i \in \mathcal{E}$ is formalized as a function or mapping that takes an input instance I and produces an output pair, consisting of a response R_i and an associated uncertainty U_i. The response space is denoted by $\mathcal{R}$, and the uncertainty space is the interval $[0, 1]$. The output space for agent i is the Cartesian product $\mathcal{O}_i = \mathcal{R} \times [0, 1]$. Thus, the function of agent i can be represented as a mapping:

$$A_i : \mathcal{I} \to \mathcal{O}_i \tag{6}$$

For a given input I, the output of agent i is:

$$\underbrace{(R_i(I), U_i(I))}_{\text{Output pair of Agent } i} = \underbrace{A_i(I)}_{\substack{\text{Mapping of Agent } i \\ \text{applied to input } I}} \tag{7}$$

where $R_i(I) \in \mathcal{R}$ is the response and $U_i(I) \in [0, 1]$ is the uncertainty. This aligns with the analysis capability mapping A_i and uncertainty assessment Φ_i described in the Method section (A_i produces R_i, and $\Phi_i(R_i)$ yields U_i).

Joint Agent Output (Joint Agent Output Mapping A): The collective output of the entire agent set for the same input I can be viewed as a combined mapping. Its output space is the Cartesian product of all individual agent output spaces: $\mathcal{O} = \prod_{i=1}^{N} \mathcal{O}_i = (\mathcal{R} \times [0, 1])^N$. The joint output mapping A is defined as:

$$A : \mathcal{I} \to \mathcal{O} \tag{8}$$

For input I, the joint output is:

$$\underbrace{(A_1(I), \ldots, A_N(I))}_{\text{Joint Agent Output}} = \underbrace{((R_1(I), U_1(I)), \ldots, (R_N(I), U_N(I)))}_{\text{Tuple of individual Agent outputs}} \tag{9}$$

System Weight Allocation Protocol ($\mathcal{P}$): The system dynamically assigns weights $\mathbf{w} = (w_1, \ldots, w_N)$ based on the joint agent output, particularly their uncertainties $\mathbf{U} = (U_1, \ldots, U_N)$ and

potentially their responses $\{R_i\}$. The weight vector $\mathbf{w}$ belongs to the standard $(N-1)$-dimensional simplex Δ^{N-1}. We formalize the weight allocation mechanism as a protocol function $\mathcal{P}$ that takes the joint agent output as input and produces a weight vector:

$$\mathcal{P} : \mathcal{O} \to \Delta^{N-1} \tag{10}$$

Thus, for input I, the weights assigned by GAM-Agent are $\mathbf{w}(I) = \underbrace{\mathcal{P}(A(I))}_{\substack{\text{Weights computed by the protocol} \\ \text{based on Agent outputs}}}$.

The weight vector must satisfy the constraints of the standard simplex:

$$\underbrace{w_i(I) \geq 0}_{\substack{\text{Each weight} \\ \text{is non-negative}}} \quad \forall i \in \mathcal{E} \quad \text{and} \quad \underbrace{\sum_{i=1}^{N} w_i(I)}_{\substack{\text{Sum of} \\ \text{all weights}}} = \underbrace{1}_{\text{equals 1}} \tag{11}$$

This protocol corresponds to the dynamic weighting mechanism used in the initial integration and iterative debate phases of GAM-Agent.

System State (σ): After receiving input I, the instantaneous state of GAM-Agent can be fully described by the joint output of the agents and the corresponding weight vector computed by GAM-Agentm. We define the system state as a tuple:

$$\sigma(I) = \underbrace{((A_1(I), \ldots, A_N(I)), \mathcal{P}(A_1(I), \ldots, A_N(I)))}_{\text{Set of Agent output pairs and corresponding system weights}} \tag{12}$$

This state $\sigma(I)$ belongs to the state space $\mathcal{S}$:

$$\sigma(I) \in \underbrace{\mathcal{O} \times \Delta^{N-1}}_{\text{System State Space } \mathcal{S}} \tag{13}$$

In the dynamic collaborative process (iterative debate), the agents' internal mappings A_i might adapt based on interaction, leading to an evolution of the state $\sigma(I)$ across different rounds.

D.2 Utility Function Design

We design a utility function $\mathcal{U}_i$ for each agent $i \in \mathcal{E}$ to quantify its "payoff" or "performance score" in the current system state. Within the consensus-driven non-zero-sum game framework, an agent's individual objective is to maximize its utility value. Concurrently, because the utility function incorporates inter-agent dependencies and the overall system state, improvements in total utility can be achieved through effective collaboration among agents.

The input to the utility function $\mathcal{U}_i$ reflects the critical information influencing agent i's payoff: the responses $\{R_j\}_{j=1}^N$ and uncertainties $\{U_j\}_{j=1}^N$ of all agents, along with the current system weight distribution $\mathbf{w}$. The utility function maps this information to a real value: $\mathcal{U}_i : \underbrace{(\mathcal{R} \times [0,1])^N}_{\substack{\text{Joint Agent Outputs} \\ \text{(Responses \& Uncertainties)}}} \times \underbrace{\Delta^{N-1}}_{\substack{\text{System} \\ \text{Weights}}} \to \mathbb{R}$

We define the utility function $\mathcal{U}_i$ for agent i as a weighted sum of three primary components:

$$\mathcal{U}_i(\{R_j\}_{j=1}^N, \{U_j\}_{j=1}^N, \mathbf{w}) = \underbrace{\overbrace{w_i}^{\substack{\text{Agent } i \\ \text{Weight}}} \cdot \overbrace{(1-U_i)}^{\substack{\text{Agent } i \\ \text{Confidence}}}}_{\text{Term 1: Individual Contribution}} + \underbrace{\lambda \sum_{j \neq i} \overbrace{w_j}^{\substack{\text{Other Agent } j \\ \text{Weight}}} \cdot \overbrace{\text{Sim}(R_i, R_j)}^{\substack{\text{Similarity between} \\ \text{Agent } i \text{ and } j}}}_{\text{Term 2: Collaboration Reward}} - \underbrace{\gamma \overbrace{U_{\text{sys}}}^{\substack{\text{System Weighted} \\ \text{Average Uncertainty}}}}_{\text{Term 3: System Penalty}}$$

where **Semantic Similarity Function (Sim):** $\text{Sim} : \mathcal{R} \times \mathcal{R} \to [0,1]$ measures the semantic relevance or consistency between two responses. $\text{Sim}(R_i, R_j) = 1$ indicates perfect semantic agreement, while $\text{Sim}(R_i, R_j) = 0$ signifies complete irrelevance. This function can be implemented using various methods, such as cosine similarity based on text embeddings, analysis of overlapping key

evidence or claims, or comparison of structured outputs. It is typically assumed to be symmetric: $\text{Sim}(r_1, r_2) = \text{Sim}(r_2, r_1)$ for all $r_1, r_2 \in \mathcal{R}$.

System Weighted Average Uncertainty (U_{sys}): This key metric quantifies the overall uncertainty of the system's current consensus, defined as the weighted average of all agent uncertainties using their current weights:

$$U_{\text{sys}} = \underbrace{\sum_{k=1}^{N} \overbrace{w_k}^{\substack{\text{Agent } k \\ \text{Weight}}} \overbrace{U_k}^{\substack{\text{Agent } k \\ \text{Uncertainty}}}}_{\substack{\text{Weighted average of all Agent uncertainties} \\ (\text{since } \sum w_k = 1)}}$$

$U_{\text{sys}} \in [0, 1]$, where lower values indicate higher system confidence in the integrated consensus result.

Hyperparameters (λ, γ): $\lambda \in \mathbb{R}_{>0}$ is the weight coefficient for the collaboration reward term, encouraging agents to align with the system consensus. $\gamma \in \mathbb{R}_{>0}$ is the weight coefficient for the system uncertainty penalty term, penalizing all agents when the overall system uncertainty is high.

Rationale and Significance of the Utility Function Design: This utility function design reflects the core objectives and trade-offs in building a collaborative consensus system:

Individual Contribution ($w_i(1 - U_i)$): Rewards agents for providing high-quality (low uncertainty U_i, thus high confidence $1 - U_i$) responses that are assigned high weight (w_i) by the system. This incentivizes agents to improve the quality and reliability of their responses.

Collaboration Reward ($\lambda \sum_{j \neq i} w_j \cdot \textbf{Sim}(R_i, R_j)$): Rewards agents for providing responses that are consistent ($\text{Sim}(R_i, R_j)$ is high) with those of other agents who are considered reliable (high weight w_j). This drives consensus by encouraging agents to consider and align with the views of highly weighted peers. λ controls the importance of this collaborative alignment.

System Penalty ($-\gamma U_{\text{sys}}$): Penalizes all agents based on the overall weighted average uncertainty of the system. When the system as a whole is uncertain about the current consensus, every agent's utility decreases. This motivates agents not only to minimize their own uncertainty but also to contribute to reducing the system's overall uncertainty through effective collaboration (e.g., sharing evidence and resolving conflicts), as the system's health benefits everyone. γ controls the focus on this systemic uncertainty.

Non-Zero-Sum Characteristic: Unlike traditional voting or simple averaging schemes which can implicitly resemble zero-sum scenarios (one agent's gain is another's loss), our design allows for collective improvement:

- Increasing collaborative consistency (higher $\text{Sim}(R_i, R_j)$) can increase the collaboration reward term for multiple agents simultaneously, boosting their utilities without necessarily decreasing individual confidence.
- Reducing the overall system uncertainty (lower U_{sys}, e.g., through effective evidence exchange) decreases the system penalty term $-\gamma U_{\text{sys}}$, thereby increasing the utility for **all** agents.

These mechanisms allow the total utility $\sum_{i=1}^{N} \mathcal{U}_i$ to potentially increase through effective inter-agent collaboration, highlighting the **positive-sum** potential of the game. This aligns better with the nature of multi-agent collaboration for complex problem-solving compared to zero-sum or fixed-sum games.

D.3 Exploration of Nash Equilibrium Existence

In formal game theory, the Nash Equilibrium (NE) is a central concept for predicting stable outcomes of rational agent behavior. It describes a profile of pure strategies $\mathbf{s}^* = (s_1^*, \ldots, s_N^*) \in \mathcal{S} = \prod_{i=1}^{N} \mathcal{S}_i$, where $\mathcal{S}_i$ is the pure strategy space of agent i, such that no agent i can improve its utility $\mathcal{U}_i(\mathbf{s})$ by unilaterally changing its strategy $s_i \in \mathcal{S}_i$, given that all other agents maintain their equilibrium strategies $\mathbf{s}_{-i}^* = (s_1^*, \ldots, s_{i-1}^*, s_{i+1}^*, \ldots, s_N^*)$. Formally, a NE $\mathbf{s}^*$ satisfies:

$$\underbrace{\mathcal{U}_i(s_i^*, \mathbf{s}_{-i}^*)}_{\substack{\text{Utility of Agent } i \\ \text{at equilibrium}}} \geq \underbrace{\mathcal{U}_i(s_i, \mathbf{s}_{-i}^*)}_{\substack{\text{Utility of Agent } i \\ \text{when deviating unilaterally}}} \quad \underbrace{\forall s_i \in \mathcal{S}_i}_{\substack{\text{for any strategy} \\ \text{of Agent } i}} , \quad \underbrace{\forall i \in \mathcal{E}}_{\text{for all Agents}} \tag{14}$$

The best response correspondence $BR_i : \mathcal{S}_{-i} \to 2^{\mathcal{S}_i}$ maps the strategy profile of other agents $\mathbf{s}_{-i}$ to the set of agent i's optimal pure strategies:

$$BR_i(\mathbf{s}_{-i}) = \underbrace{\{s_i' \in \mathcal{S}_i \mid \mathcal{U}_i(s_i', \mathbf{s}_{-i}) \geq \mathcal{U}_i(s_i, \mathbf{s}_{-i}) \quad \forall s_i \in \mathcal{S}_i\}}_{\text{Set of Pure Best Responses of Agent } i \text{ given } \mathbf{s}_{-i}} \tag{15}$$

A pure strategy Nash equilibrium $\mathbf{s}^*$ is a fixed point of the joint best response correspondence $BR : \mathcal{S} \to 2^{\mathcal{S}}$, where $BR(\mathbf{s}) = \prod_{i=1}^{N} BR_i(\mathbf{s}_{-i})$:

$$\mathbf{s}^* \in \underbrace{BR(\mathbf{s}^*)}_{\text{Joint Best Response}} \tag{16}$$

Theorem 1 (Existence of Pure Strategy Nash Equilibrium): Consider a game with a finite number of agents $N < \infty$. Under specific assumptions regarding the agents' strategy spaces $\mathcal{S}_i$ and utility functions $\mathcal{U}_i$, the non-zero-sum game described above possesses at least one pure strategy Nash equilibrium.

Proof Outline: The existence of a pure strategy NE is often established using Kakutani's Fixed Point Theorem applied to the joint best response correspondence $BR : \mathcal{S} \to 2^{\mathcal{S}}$. This requires satisfying the following conditions: **1. Properties of the Domain Space:** The domain of BR, the joint pure strategy space $\mathcal{S} = \prod_{i=1}^{N} \mathcal{S}_i$, must be a non-empty, compact, and convex subset of a Euclidean space $\mathbb{R}^d$.

$$\mathcal{S} \neq \emptyset, \qquad \underbrace{\mathcal{S}}_{\text{Joint strategy space}} \text{ is compact}, \qquad \underbrace{\mathcal{S}}_{\text{Joint strategy space}} \text{ is convex}$$

This necessitates that each agent's pure strategy space $\mathcal{S}_i$ is non-empty, compact, and convex.

$$\underbrace{\mathcal{S}_i \neq \emptyset}_{\text{Non-empty}}, \quad \underbrace{\mathcal{S}_i \text{ is compact}}_{\text{Compact}}, \quad \underbrace{\mathcal{S}_i \text{ is convex}}_{\text{Convex}} \quad \underbrace{\forall i \in \mathcal{E}}_{\text{for all Agents}}$$

In VLM systems, an agent's pure strategy s_i determines how it generates its response $R_i \in \mathcal{R}$ and uncertainty $U_i \in [0, 1]$. The response space $\mathcal{R}$ (e.g., all possible text sequences) is typically discrete, vast, and non-convex, making it challenging to define a compact and convex strategy space $\mathcal{S}_i$ that fully captures the agent's complex behavior.

1. **Simplified Strategy Spaces under Assumptions:** To meet the theorem's requirements, one might model agent strategies more simply:
   - **Assumption 1.1 (Uncertainty Strategy):** Agent i's pure strategy s_i is solely its choice of reported uncertainty $U_i \in [0, 1]$. Then $\mathcal{S}_i = [0, 1]$, which is non-empty, compact, and convex.
   - **Assumption 1.2 (Low-Dimensional Continuous Strategy):** Agent i's pure strategy s_i is determined by a low-dimensional continuous vector $\theta_i \in \Theta_i \subset \mathbb{R}^d$, representing internal parameters influencing its response and uncertainty. If Θ_i is non-empty, compact, and convex (e.g., a hypercube), and the agent's output (R_i, U_i) is a continuous function of s_i, this requirement is met.
2. **Properties of the Mapped Values:** The value of the mapping $BR(\mathbf{s})$ (the joint best response set) must be non-empty and convex. This means that for each agent i, its best response set $BR_i(\mathbf{s}_{-i})$ must be non-empty and convex.

$$\underbrace{BR_i(\mathbf{s}_{-i}) \neq \emptyset}_{\text{Non-empty}}, \quad \underbrace{BR_i(\mathbf{s}_{-i}) \text{ is a convex set}}_{\text{Convex Set}} \quad \underbrace{\forall \mathbf{s}_{-i} \in \mathcal{S}_{-i}}_{\substack{\text{for any strategy profile} \\ \text{of other agents}}} , \underbrace{\forall i \in \mathcal{E}}_{\text{for all Agents}} \tag{17}$$

   This property can be established by analyzing the agent's utility function $\mathcal{U}_i$ with respect to its own strategy s_i. Specifically, if $\mathcal{S}_i$ is non-empty, compact, and convex, and $\mathcal{U}_i(s_i, \mathbf{s}_{-i})$ is continuous and **quasi-concave** in s_i, then $BR_i(\mathbf{s}_{-i})$ is guaranteed to be non-empty and convex.

$$\underbrace{s_i \mapsto \mathcal{U}_i(s_i, \mathbf{s}_{-i})}_{\substack{\text{Utility of Agent } i \\ \text{w.r.t. its own strategy}}} \text{ is quasi-concave} \quad \forall i \in \mathcal{E}, \forall \mathbf{s}_{-i} \in \mathcal{S}_{-i} \tag{18}$$

Proving quasi-concavity for $\mathcal{U}_i$ w.r.t. s_i is a central challenge, especially when s_i determines complex outputs (R_i, U_i). It requires analyzing the convexity of the upper level sets $\{s_i \in \mathcal{S}_i \mid \mathcal{U}_i(s_i, \mathbf{s}_{-i}) \geq c\}$. Under simplifying assumptions (e.g., s_i is choosing $U_i \in [0, 1]$ and $\mathcal{U}_i$ is quasi-concave in U_i), this condition can be met.

3. **Continuity of the Mapping:** The correspondence $BR(\mathbf{s})$ must have a closed graph or, more strongly, be upper hemi-continuous (UHC). This implies that if a sequence of strategy profiles $\mathbf{s}^{(m)} \to \mathbf{s}$ and a corresponding sequence of best responses $\mathbf{s}'^{(m)} \in BR(\mathbf{s}^{(m)})$ converge to $\mathbf{s}'$, then $\mathbf{s}'$ must be a best response to $\mathbf{s}$, i.e., $\mathbf{s}' \in BR(\mathbf{s})$.

$$\underbrace{BR \text{ has a closed graph (or is UHC)}}_{\text{Closed Graph / Upper Hemi-continuous}} \tag{19}$$

If the strategy space $\mathcal{S}$ is compact and the utility functions $\mathcal{U}_i(\mathbf{s})$ are continuous on $\mathcal{S}$, then the best response correspondence BR_i is UHC, and consequently, the joint correspondence BR is also UHC. Thus, continuity of the utility functions is key.

If all these conditions (non-empty, compact, convex joint strategy space; non-empty, convex-valued, UHC joint best response correspondence) are satisfied, Kakutani's Fixed Point Theorem guarantees the existence of a fixed point $\mathbf{s}^* \in BR(\mathbf{s}^*)$, which is a pure strategy Nash equilibrium[37]. Rigorously verifying these conditions for complex VLM systems often requires simplified models or strong assumptions about agent behavior and interaction functions.

Mixed Strategy Nash Equilibrium: When proving the existence of a pure strategy NE is difficult (e.g., due to non-convex strategy spaces or non-quasi-concave utilities), one can consider mixed strategy Nash equilibria. A mixed strategy σ_i for agent i is a probability distribution over its pure strategy space $\mathcal{S}_i$. The space of mixed strategies for agent i is $\Sigma_i = \Delta(\mathcal{S}_i)$. Under mixed strategies, agents maximize their expected utility. Nash's Theorem guarantees that a mixed strategy NE always exists for games with finite agents and finite pure strategy sets, or more generally, if pure strategy spaces are compact metric spaces and utility functions are continuous.

$$\underbrace{\Sigma_i = \Delta(\mathcal{S}_i)}_{\substack{\text{Mixed strategy space} \\ \text{for Agent } i}} , \quad \underbrace{\sigma = (\sigma_1, \ldots, \sigma_N)}_{\substack{\text{Joint mixed} \\ \text{strategy}}} \tag{20}$$

A mixed strategy NE $\sigma^* = (\sigma_1^*, \ldots, \sigma_N^*)$ exists such that for all agents i:

$$E_{\mathbf{s} \sim \sigma^*}[\mathcal{U}_i(\mathbf{s})] \geq E_{(s_i, \mathbf{s}_{-i}) \sim (\sigma_i, \sigma_{-i}^*)}[\mathcal{U}_i(s_i, \mathbf{s}_{-i})] \qquad \underbrace{\forall \sigma_i \in \Sigma_i}_{\substack{\text{for any mixed strategy} \\ \text{of Agent } i}} \tag{21}$$

For the VLM game, if we consider the agent pure strategy space as a compact metric space (e.g., via representation learning mapping responses to a compact space) and utility functions are continuous, Nash's Theorem ensures the existence of a mixed strategy NE. While existence is more readily guaranteed, the interpretation and practical relevance of mixed strategies in VLM collaborative reasoning require further investigation.

In summary, proving the existence of a pure strategy NE in the VLM game requires rigorous mathematical analysis of the agents' pure strategy spaces and utility functions, often under simplifying assumptions that satisfy specific mathematical properties.

D.4 Cooperative Game Objective and Dynamic Weighting Protocol

The dynamic process in our system aligns with the philosophy of cooperative game theory, where agents collaborate to enhance the overall quality of the consensus. The system-level objective can be framed as maximizing the sum of all agents' utilities[20], i.e., the total utility $\mathcal{J}$:

$$\max_{\mathbf{w} \in \Delta^{N-1}} \mathcal{J}(\{R_i\}_{i=1}^N, \{U_i\}_{i=1}^N, \mathbf{w}) = \max_{\mathbf{w} \in \Delta^{N-1}} \underbrace{\sum_{i=1}^N \mathcal{U}_i(\{R_j\}, \{U_j\}, \mathbf{w})}_{\substack{\text{System Total Utility} \\ \text{(Sum of Individual Utilities)}}} \tag{22}$$

Substituting the definition of $\mathcal{U}_i$ into the total utility $\mathcal{J}$:

$$\mathcal{J} = \sum_{i=1}^{N} \left[w_i(1 - U_i) + \lambda \sum_{j \neq i} w_j \cdot \text{Sim}(R_i, R_j) - \gamma U_{\text{sys}} \right] \tag{23}$$

Expanding and rearranging terms, using $\sum_{i=1}^{N} w_i = 1$ and $U_{\text{sys}} = \sum_{k=1}^{N} w_k U_k$:

$$\mathcal{J} = \underbrace{\sum_{i=1}^{N} w_i(1 - U_i)}_{\substack{\text{Sum of Individual} \\ \text{Contributions}}} + \underbrace{\lambda \sum_{i=1}^{N} \sum_{j \neq i} w_j \cdot \text{Sim}(R_i, R_j)}_{\substack{\text{Sum of Collaboration} \\ \text{Rewards}}} - \underbrace{\sum_{i=1}^{N} \gamma U_{\text{sys}}}_{\substack{\text{Sum of System} \\ \text{Penalties}}} \tag{24}$$

$$= \left(\sum_{i=1}^{N} w_i - \sum_{i=1}^{N} w_i U_i \right) + \lambda \sum_{i=1}^{N} \sum_{j \neq i} w_j \cdot \text{Sim}(R_i, R_j) - N\gamma U_{\text{sys}} \tag{25}$$

$$= (1 - U_{\text{sys}}) + \lambda \sum_{i=1}^{N} \sum_{j \neq i} w_j \cdot \text{Sim}(R_i, R_j) - N\gamma U_{\text{sys}} \tag{26}$$

$$= 1 + \lambda \sum_{i=1}^{N} \sum_{j \neq i} w_j \cdot \text{Sim}(R_i, R_j) - (1 + N\gamma) U_{\text{sys}} \tag{27}$$

Substituting $U_{\text{sys}} = \sum_{k=1}^{N} w_k U_k$ and ignoring the constant term 1, the optimization problem becomes:

$$\max_{\mathbf{w} \in \Delta^{N-1}} \left[\underbrace{\lambda \sum_{i=1}^{N} \sum_{j \neq i} w_j \cdot \text{Sim}(R_i, R_j)}_{\text{Total Weighted Similarity Term}} - (1 + N\gamma) \underbrace{\sum_{k=1}^{N} w_k U_k}_{\text{System Weighted Uncertainty}} \right] \tag{28}$$

Rearranging the summation terms to isolate w_k:

$$\sum_{i=1}^{N} \sum_{j \neq i} w_j \cdot \text{Sim}(R_i, R_j) = \sum_{j=1}^{N} w_j \sum_{i \neq j} \text{Sim}(R_i, R_j) \tag{29}$$

$$= \sum_{k=1}^{N} w_k \sum_{i \neq k} \text{Sim}(R_i, R_k) \quad \text{(relabeling index } j \text{ to } k\text{)} \tag{30}$$

The objective function becomes:

$$\max_{\mathbf{w} \in \Delta^{N-1}} \sum_{k=1}^{N} w_k \underbrace{\left[\lambda \sum_{i \neq k} \text{Sim}(R_i, R_k) - (1 + N\gamma) U_k \right]}_{\substack{\text{Agent } k\text{'s "Cooperative Value Score"} \\ \text{Score}_k}} \tag{31}$$

Let $\text{Score}_k = \lambda \sum_{i \neq k} \text{Sim}(R_i, R_k) - (1 + N\gamma) U_k$. For fixed $\{R_i\}$ and $\{U_i\}$, this is maximizing a linear function $\sum_{k=1}^{N} w_k \cdot \text{Score}_k$ over the weight simplex Δ^{N-1}. To obtain a smooth, non-degenerate weight distribution, we introduce an entropy regularization term $-\frac{1}{\eta} \sum_{k=1}^{N} w_k \log w_k$ (with $\eta > 0$) to the objective:

$$\max_{\mathbf{w} \in \Delta^{N-1}} \underbrace{\left[\sum_{k=1}^{N} w_k \cdot \text{Score}_k - \frac{1}{\eta} \sum_{k=1}^{N} w_k \log w_k \right]}_{\substack{\text{Entropy-Regularized} \\ \text{Total Utility}}} \tag{32}$$

The solution to this standard convex optimization problem (found via KKT conditions) yields optimal weights w_k^* in the form of a Softmax distribution:

$$w_k^* = \underbrace{\frac{\exp(\eta \cdot \text{Score}_k)}{\sum_{m=1}^{N} \exp(\eta \cdot \text{Score}_m)}}_{\substack{\text{Static Optimal Weight Form} \\ \text{(depends on Score}_k\text{)}}} \tag{33}$$

Substituting the definition of Score_k:

$$w_k^* = \underbrace{\frac{\exp\left(\eta\left(\lambda \sum_{i\neq k} \text{Sim}(R_i, R_k) - (1+N\gamma)U_k\right)\right)}{\sum_{m=1}^{N} \exp\left(\eta\left(\lambda \sum_{i\neq m} \text{Sim}(R_i, R_m) - (1+N\gamma)U_m\right)\right)}}_{\substack{\text{Optimal weight for Agent } k \\ \text{(depends on all } R,U\text{, and hyperparameters } \lambda,\gamma,\eta\text{)}}} \tag{34}$$

This static optimal weight distribution w_k^* depends on all agent responses, uncertainties, and hyperparameters, reflecting the full cooperative optimization goal.

Actual Dynamic Weighting Protocol Used in GAM-Agent: In our iterative GAM-Agent framework, the weights are updated dynamically at each round k based primarily on the agents' uncertainties from that round. The protocol used is:

$$w_i^{(k+1)} = \underbrace{\frac{e^{-\beta U_i^{(k)}}}{\sum_{j=1}^{N} e^{-\beta U_j^{(k)}}}}_{\substack{\text{Weight for round } k+1 \\ \text{(based on uncertainty from round } k\text{)}}} \qquad (\beta > 0) \tag{35}$$

Here, $\beta > 0$ controls the sensitivity to uncertainty. This corresponds to the weighting described in Section 2.3 and used implicitly in Section 2.5.

Connecting Static Optimality and the Dynamic Protocol: We now demonstrate that the dynamic weighting protocol (Eq. 35) is not an arbitrary heuristic but corresponds to the optimal solution of a simplified, entropy-regularized static optimization problem. Consider a simplification of the total utility $\mathcal{J}$ where we ignore the collaboration reward ($\lambda = 0$) and the system penalty ($\gamma = 0$). The objective becomes maximizing the weighted sum of individual agent confidences:

$$\max_{\mathbf{w}\in\Delta^{N-1}} \sum_{i=1}^{N} \mathcal{U}_i \approx \max_{\mathbf{w}\in\Delta^{N-1}} \sum_{i=1}^{N} w_i(1-U_i) \tag{36}$$

Let the "score" for agent i be its confidence $s_i = (1-U_i)$. We want to maximize the weighted average score $\sum_{i=1}^{N} w_i s_i$ with entropy regularization $\frac{1}{\beta}\sum_{i=1}^{N} w_i \log w_i$ (note the sign change compared to minimization; maximizing score + entropy):

$$\max_{\mathbf{w}\in\Delta^{N-1}} \left[\sum_{i=1}^{N} w_i(1-U_i) + \frac{1}{\beta}\sum_{i=1}^{N} w_i \log w_i\right] \tag{37}$$

The optimal solution w_i^* for this entropy-regularized maximization problem takes the Softmax form based on the score $(1-U_i)$:

$$w_i^* = \underbrace{\frac{\exp(\beta \cdot (1-U_i))}{\sum_{j=1}^{N} \exp(\beta \cdot (1-U_j))}}_{\substack{\text{Static optimal weight form} \\ \text{(based on confidence score)}}} \tag{38}$$

$$= \underbrace{\frac{\exp(\beta)\exp(-\beta U_i)}{\sum_{j=1}^{N} \exp(\beta)\exp(-\beta U_j)}}_{\text{Exponential form expansion}} \tag{39}$$

$$= \underbrace{\frac{e^{-\beta U_i}}{\sum_{j=1}^{N} e^{-\beta U_j}}}_{\text{Simplified form}} \tag{40}$$

The resulting optimal weight w_i^* has exactly the same form as the dynamic weighting protocol used in GAM-Agent (Eq. 35), with β playing the role of the inverse temperature parameter η.

$$\underbrace{w_i^*}_{\substack{\text{Static optimal weight based} \\ \text{on regularized confidence}}} = \underbrace{\frac{e^{-\beta U_i}}{\sum_{j=1}^{N} e^{-\beta U_j}}}_{\substack{\text{Matches the dynamic} \\ \text{protocol form}}}$$

This derivation shows that the dynamic weighting rule employed in GAM-Agent, $w_i^{(k+1)} = \frac{\exp(-\beta U_i^{(k)})}{\sum_j \exp(-\beta U_j^{(k)})}$, corresponds precisely to the optimal weight distribution derived from maximizing the entropy-regularized weighted average of agent confidences $(1 - U_i^{(k)})$ at each step k. While this dynamic protocol simplifies the full cooperative objective (Eq. 34) by omitting explicit collaboration rewards and system penalties in the weight calculation, it provides a computationally efficient and theoretically grounded mechanism. It prioritizes influence for agents demonstrating lower uncertainty (higher confidence) at each stage, driving the system towards a consensus state characterized by reduced overall uncertainty, as explored in the convergence analysis related to the debate termination condition (Eq. 5).

E Theorem I: Fundamental Game Theoretic Framework of GAM-Agent

The core mechanism of GAM-Agent is driven by a mathematically well-defined, non-zero-sum game model centered on "uncertainty" as a key variable. This model derives its collaborative strategies and influences allocation through the optimization of explicit utility functions[66].

Definition 1 (Mathematical Formalization of GAM-Agent S)**.** *Our GAM-Agent S is rigorously defined as a sextuple:*

$$S = (\underbrace{E}_{\substack{\text{Agent} \\ \text{Set}}}, \underbrace{A}_{\substack{\text{Analytical Capability} \\ \text{Map Function Set}}}, \underbrace{\Phi}_{\substack{\text{Uncertainty} \\ \text{Assessment Function Set}}}, \underbrace{M}_{\substack{\text{Evidence Localization} \\ \text{Function Set}}}, \underbrace{P}_{\substack{\text{Claim Parsing} \\ \text{Function Set}}}, \underbrace{D}_{\substack{\text{Dynamic Debate} \\ \text{Mechanism}}})$$

where:$E = 1, 2, \ldots, N$ represents the set of N expert agents. $A = A_1, A_2, \ldots, A_N$ defines the set of analytical capability mapping functions, where each function $A_i : X \times P_r \to R_i$ maps an input image X and a question P_r to agent i's response R_i. $\Phi = \Phi_1, \Phi_2, \ldots, \Phi_N$ is the set of uncertainty assessment functions, where $\Phi_i : R_i \to [0, 1]$ quantifies the uncertainty U_i associated with response R_i. $M = M_1, M_2, \ldots, M_N$ represents the set of evidence localization functions, where $M_i : C_i \times X \to V \times [0, 1]$ associates agent i's claim c with a visual region r and a confidence score σ. $P : R \to (c_j, \sigma_j, e_j, r_j)j = 1^K$ is the claim parsing function, converting unstructured responses into structured information tuples (K is the number of tuples). $D : X \times P_r \times R_i, U_i i = 1^N \to R_{final}$ is the debate module that coordinates conflicts and consensus to generate a final response. This formalization constructs a complete mathematical space, allowing for precise description and analysis of the system's components and their interactions.

Definition 2 (Mathematical Construction of Uncertainty Quantification Function $\Phi_i(R_i)$)**.** *GAM-Agent provides two uncertainty quantification mechanisms for each agent $e_i \in E$:*

- ***Uncertainty based on generation probability*** $\Phi_{igen+}(R_i)$***:*** *When the token generation probability distribution of the underlying VLM is accessible, it is defined as:*

$$\Phi_{igen+}(R_i) = \frac{1}{T_i} \sum_{t=1}^{T_i} \left[\alpha \cdot \underbrace{H(P_{i,t})}_{\substack{\text{Information Entropy} \\ \text{(Hesitation)}}} + \beta \cdot \underbrace{\max(0, 1 - \Delta_{top}(P_{i,t}))}_{\substack{\text{Inverse Probability Difference} \\ \text{(Lack of Confidence)}}} \right]$$

where: T_i is the total number of tokens in response R_i. $P_{i,t}$ is the probability distribution of the t-th token. $H(P_{i,t}) = -\sum_x P_{i,t}(x) \log P_{i,t}(x)$ is the entropy, quantifying the dispersion or "hesitation" of the distribution. $\Delta_{top}(P_{i,t}) = p_{i,t}^{(1)} - p_{i,t}^{(2)}$ is the difference between the

probabilities of the top two tokens, where $p_{i,t}^{(k)}$ is the k-th highest probability. α and β are parameters balancing these two complementary signals. High entropy $H(P_{i,t})$ indicates model hesitation among multiple possible tokens, while small $\Delta_{top}(P_{i,t})$ indicates a lack of strong confidence in its preferred token.

By formally expanding the entropy calculation:

$$H(P_{i,t}) = -\sum_{x \in V} P_{i,t}(x) \log P_{i,t}(x)$$

where V is the vocabulary, we see that entropy is maximized when the probability distribution $P_{i,t}$ approaches a uniform distribution (highest hesitation), and minimized when the distribution is concentrated on a single token (lowest hesitation).

- ***Uncertainty based on semantic markers*** $\Phi_{isem}(R_i)$***:*** *When generation probabilities are unavailable, the system uses:*

$$\Phi_{isem}(R_i) = \sigma_{sigmoid}\left(k \cdot \left(\frac{\sum_{w \in W} weight(w) \cdot count(w, R_i)}{|R_i|} - offset\right)\right)$$

where: W is a multi-level lexicon of uncertainty markers. $weight(w)$ reflects the uncertainty intensity of marker w. $count(w, R_i)$ is the frequency of w in response R_i. $|R_i|$ is the response length. k and offset control scaling and bias. $\sigma_{sigmoid}(x) = \frac{1}{1+e^{-x}}$ normalizes the value to the $(0, 1)$ interval.

These two methods capture uncertainty from the generation process and semantic content, respectively, providing key inputs for the subsequent game theory framework.

Definition 3 (Non-zero Sum Game Theoretic Formalization of GAM-Agent)**.** *The multi-agent visual reasoning process is modeled as a non-zero-sum game with the following formal structure:* ***Players:*** *Set of agents $\mathcal{E} = 1, \ldots, N$.* ***Input Instance:*** *$I = (X, P_r) \in \mathcal{I} = \mathcal{X} \times \mathcal{P}r$, where X is the image and P_r is the question.* ***Action Output of Agent*** *i:* *The action of agent e_i is to generate a response-uncertainty pair $(R_i(I), U_i(I))$, with the mapping function: $A_i : \mathcal{I} \to \mathcal{O}i \equiv \mathcal{R} \times [0, 1]$, $A_i(I) = (R_i(I), \Phi_i(R_i(I)))$.* ***System Weight Allocation Protocol*** *$\mathcal{P}$:* *Rules for assigning influence weights, formalized as: $\mathcal{P} : \mathcal{O}^N \to \Delta^{N-1}$. A key weight calculation formula (initial weight $w_i^{(0)}$) is: $w_i^{(0)}(U_j j = 1^N) = \frac{\exp(-\beta U_i)}{\sum j=1^N \exp(-\beta U_j)}$. This is an uncertainty-based Softmax allocation: higher U_i (greater uncertainty) leads to lower weight $w_i^{(0)}$. Parameter $\beta > 0$ controls sensitivity to uncertainty differences.* ***System State*** *σ:* *The instantaneous state of the system is fully described by the joint outputs of the agents and the corresponding weight vector: $\sigma(I) = ((A_1(I), \ldots, A_N(I)), \mathcal{P}(A_1(I), \ldots, A_N(I)))$, $\sigma(I) \in \mathcal{O}^N \times \Delta^{N-1}$. This game structure formalizes the interactions among agents in GAM-Agent and clarifies how uncertainty affects weight allocation and system state evolution.*

Definition 4 (Mathematical Expansion of Agent Utility Function u_i)**.** *A utility function $u_i : \mathcal{O}^N \times \Delta^{N-1} \to \mathbb{R}$ is defined for each agent e_i, mapping joint agent outputs and weight allocation to a real-valued utility:*

$$u_i(\{R_j\}_{j=1}^N, \{U_j\}_{j=1}^N, w) = \underbrace{w_i(1 - U_i)}_{\substack{\text{Individual Contribution Term} \\ \text{(Term 1)}}} + \underbrace{\lambda \sum_{j \neq i} w_j \cdot Sim(R_i, R_j)}_{\substack{\text{Collaborative Benefit Term} \\ \text{(Term 2)}}} - \underbrace{\gamma U_{sys}}_{\substack{\text{System Cost Term} \\ \text{(Term 3)}}}$$

where: Individual contribution term (Term 1): $w_i(1 - U_i)$ rewards the agent for providing high-confidence (low uncertainty) judgments, reflecting its system-assigned influence. Collaborative benefit term (Term 2): $\lambda \sum_{j \neq i} w_j \cdot Sim(R_i, R_j)$ rewards the agent for consistency with other high-influence agents, where $Sim(R_i, R_j) \in [0, 1]$ is the semantic similarity between responses, and $\lambda > 0$ is a weight coefficient. System cost term (Term 3): γU_{sys}, where $U_{sys} = \sum_{k=1}^N w_k U_k$ is the system's weighted average uncertainty, and $\gamma > 0$ is a weight coefficient. All agents share the system's uncertainty. From a mathematical theory perspective, this utility function has the following properties: Non-zero-sum nature: Agents can collectively increase total utility $\sum_{i=1}^N u_i$ by reducing collective uncertainty and improving consistency, without necessarily competing against each other.

The incentive for collective reasoning: The collaborative benefit term incentivizes agents to agree with other highly influential agents. Uncertainty management: Agents must reduce their own uncertainty (Term 1) and contribute to reducing overall system uncertainty (Term 3). By detailing the system cost term:$\gamma U_{sys} = \gamma \sum_{k=1}^{N} w_k U_k$ *we see that even if an agent's own uncertainty* U_i *is low, it will be negatively affected if the overall system still has high uncertainty (high* U_{sys}*), further promoting cooperative behavior.*

Proposition 5 (Analytic Optimal w_k^* for Regularized Utility)**.** *Our GAM-Agent seeks optimal influence weights* $w^* = (w_1^*, \ldots, w_N^*)$ *that maximize the total utility:*

$$\mathcal{U}_{total}(w) = \sum_{i=1}^{N} u_i\big(R_j{}_{j=1}^{N}, U_j{}_{j=1}^{N}, w\big)$$

Expanding with Definition 1.4:

$$\mathcal{U}_{total}(w) = \sum_{i=1}^{N} \Big[w_i(1 - U_i) + \lambda \sum_{j \neq i} w_j \, Sim(R_i, R_j) - \gamma \sum_{k=1}^{N} w_k U_k \Big]$$

Further expansion:

Using the constraint $\sum_{i=1}^{N} w_i = 1$ *to simplify:*

$$\begin{aligned} \mathcal{U}_{total}(w) &= 1 - \sum_{i=1}^{N} w_i U_i + \lambda \sum_{i=1}^{N} \sum_{j \neq i} w_j \cdot Sim(R_i, R_j) - N\gamma \sum_{k=1}^{N} w_k U_k \\ &= 1 + \lambda \sum_{i=1}^{N} \sum_{j \neq i} w_j \cdot Sim(R_i, R_j) - (1 + N\gamma) \sum_{k=1}^{N} w_k U_k \end{aligned}$$

Rearranging the summation order:

$$\sum_{i=1}^{N} \sum_{j \neq i} w_j \cdot Sim(R_i, R_j) = \sum_{j=1}^{N} w_j \sum_{i \neq j} Sim(R_i, R_j) = \sum_{k=1}^{N} w_k \sum_{i \neq k} Sim(R_i, R_k)$$

Substituting into the above equation and ignoring the constant term 1, we get the optimization problem:

$$\max_{w \in \Delta^{N-1}} \sum_{k=1}^{N} w_k \left[\lambda \sum_{i \neq k} Sim(R_i, R_k) - (1 + N\gamma) U_k \right]$$

Define agent k*'s "cooperative value score":*

$$Score_k = \lambda \sum_{i \neq k} Sim(R_i, R_k) - (1 + N\gamma) U_k$$

To avoid degenerate solutions (all weight assigned to the agent with the highest score), an entropy regularization term $-\frac{1}{\eta} \sum_{k=1}^{N} w_k \log w_k$ *(where* $\eta > 0$ *is a temperature parameter) is introduced, yielding the regularized optimization problem:* $\max_{w \in \Delta^{N-1}} \left[\sum_{k=1}^{N} w_k \cdot Scorek - \frac{1}{\eta} \sum k = 1^N w_k \log w_k \right]$ *This is a standard convex optimization problem. Using the Lagrange multiplier method, the analytical form of the optimal weights is:* $w_k^* = \frac{\exp(\eta \cdot Scorek)}{\sum m=1^N \exp(\eta \cdot Score_m)}$ *Substituting the full expression for* $Score_k$*:*

$$w_k^* = \frac{\exp\left(\eta \cdot \left[\lambda \sum_{i \neq k} Sim(R_i, R_k) - (1 + N\gamma) U_k\right]\right)}{\sum_{m=1}^{N} \exp\left(\eta \cdot \left[\lambda \sum_{i \neq m} Sim(R_i, R_m) - (1 + N\gamma) U_m\right]\right)}$$

From this analytical solution, it is clear that: Weight w_k^* *decreases exponentially with an increase in agent* k*'s uncertainty* U_k*. Weight* w_k^* *increases exponentially with an increase in agent* k*'s semantic consistency with other agents,* $\sum_{i \neq k} Sim(R_i, R_k)$*. Parameter* η *controls the sensitivity of weight allocation to differences in* $Score_k$*—high* η *values lead to a more "winner-take-all" allocation.*

This proves that agent influence allocation in GAM-Agent is an analytical solution with a clear mathematical basis, rather than a simple heuristic rule.

GAM-Agent establishes a mathematically self-consistent and mechanistically interpretable non-zero-sum game framework through rigorous mathematical definitions and derivations of the system, uncertainty, game interactions, utility functions, and optimization objectives. Its core mathematical contributions are: Transforming "uncertainty" from an abstract cognitive concept into a precisely quantifiable mathematical variable (via $\Phi_{i\text{gen+}}$ and $\Phi_{i\text{sem}}$). Designing a three-part utility function u_i that balances individual contribution, collaborative consistency, and system risk. Analytically deriving the mathematical expression for agent influence weights w_k^* by maximizing the regularized total utility $\mathcal{U}_{\text{total}}$. Demonstrating that the system's game mechanism is endogenously generated from utility optimization principles, rather than externally imposed.

F Theorem II: Comparative Analysis with the DMAD Framework

The mathematical formalism of DMAD (Diverse Multi-Agent Debate) is primarily manifested as an algorithmic flow based on externally preset diversification strategies. Its core mechanism, "breaking cognitive fixation," mathematically lacks an endogenous, agent-centric utility optimization model equivalent to GAM-Agents to drive it.

Definition 6 (Mathematical Representation of the DMAD Algorithm Flow)**.** *The core of the DMAD framework guides n agent instances $\{\mathcal{M}_i\}_{i=1}^n$ to each adopt a distinct reasoning method (prompting strategy) $\mathfrak{R}_i$ assigned from a preset collection $\mathbb{R} = \{\mathfrak{R}_1, \ldots, \mathfrak{R}_n\}$. Its mathematical flow can be formalized for each round j of debate as:**Individual Solution Generation:** $\forall i \in 1, \ldots, n$: $(s_{i,j}, y_{i,j}) = Execute(\mathcal{M}i, x, h_i, \mathfrak{R}i)$ where: $\mathcal{M}i$ is the i-th agent model. x is the input task. h_i is agent i's historical information. $\mathfrak{R}i$ is the preset, fixed reasoning method/prompting strategy. (si, j, yi, j) is the output pair, containing the reasoning process si, j and answer yi, j. **Information Propagation and History Update:** Let $\mathbb{H}j = (sk, j, y_{k,j})_{k=1}^n$ be the set of outputs from all agents in round j. $\forall i \in 1, \ldots, n : \quad h_i \leftarrow UpdateFunction(h_i, \mathbb{H}_j)$ This step propagates the outputs of all agents to each agent, updating their historical information.*

Analysis 1 (Comparative Analysis of DMAD's Mathematical Form — The Missing Endogenous Game Optimization Aspect)**.** *Compared to GAM-Agent (Theorem I), DMAD exhibits the following substantial differences in the mathematical formalization of its core mechanisms: Compared to GAM-Agent (Theorem I), DMAD exhibits the following substantial differences in the mathematical formalization of its core mechanisms: **Lack of an optimizable, agent-level mathematical utility function:** The DMAD framework does not define a mathematical utility function for each agent $\mathcal{M}i$ similar to u_i in GAM-Agent. Formally, there is no mapping: u_i^{DMAD} : System State $\to \mathbb{R}$ Therefore, when agent $\mathcal{M}i$ executes its assigned reasoning method $\mathfrak{R}i$, its behavior (e.g., how it interprets and utilizes information h_i from other agents to improve its own $si, j + 1, yi, j + 1$) is not driven by a clear optimization process aimed at maximizing its own mathematical utility. Its "improvement" relies more on the LLM's inherent context-learning capabilities and adherence to "critical feedback." In GAM-Agent, every decision can be traced back to the maximization of utility function u_i: $\max a_i u_i(a_i, a_{-i}, U_j j = 1^N, R_j j = 1^N)$ In DMAD, agent behavior is solely determined by the fixed strategy $\mathfrak{R}_i$ and historical information h_i, lacking this explicit optimization structure. **Mathematical implementation of the "diversity" mechanism:** DMAD's core idea, "breaking cognitive fixation through diverse reasoning," is achieved by externally enforcing different reasoning methods $\mathfrak{R}_i$ upon different agents. Mathematically, this is represented as a mapping: Assign : $1, \ldots, n \to \mathbb{R}$ $Assign(i) = \mathfrak{R}_i$ such that $\mathfrak{R}_i \neq \mathfrak{R}j$ if $i \neq j$ This "diversity" is a system-level structural design, not a result of individual agents selecting strategies based on some intrinsic mathematical incentive (like a "diversity contribution utility"). In contrast, the uncertainty and utility-driven mechanisms in GAM-Agent allow diversity to emerge naturally from the game–different agents gain higher weights in different aspects where their uncertainty is low, forming complementary rather than preset specializations. **Non-explicit mathematical modeling of "breaking cognitive fixation":** "Cognitive fixation" itself and the process of it "being broken" are core cognitive science assumptions and desired outcomes in DMAD, but they are not directly modeled mathematically as variables or objective functions that can be quantified or optimized in agent decision-making or system evolution. Formally, there is no function: $fmindset$: System State $\to \mathbb{R}$ Its effect is indirectly validated through experiments (e.g., performance improvements on different benchmarks), rather than predicted or explained through the analysis of a clearly defined mathematical objective function.*

To more clearly illustrate the fundamental differences in mathematical construction between DMAD and GAM-Agent, we can compare their mathematical derivation processes for the same task:

For a visual reasoning task (X, P_r):

DMAD's Processing Flow:

1. *Preset different reasoning methods:* $\{\mathfrak{R}_1, \ldots, \mathfrak{R}_n\}$
2. *Assign to agents:* $\mathcal{M}_i$ *uses* $\mathfrak{R}_i$
3. *Execute multi-round interaction:* $(s_{i,j}, y_{i,j}) = \textit{Execute}(\mathcal{M}_i, x, h_i, \mathfrak{R}_i)$
4. *Final aggregation:* $y_{\textit{final}} = \phi(\{y_{i,J}\}_{i=1}^n)$

GAM-Agent's Processing Flow:

1. *Calculate uncertainty:* $U_i = \Phi_i(R_i)$
2. *Optimize weights based on utility function* u_i*:* $w_i^* = \frac{\exp(\eta \cdot \textit{Score}_i)}{\sum_j \exp(\eta \cdot \textit{Score}_j)}$
3. *Trigger debate based on uncertainty and conflict:* $\textit{TriggerDebate} = (U_{\textit{sys}}^{(0)} > \theta_U) \vee (\textit{ConflictScore} > \theta_C)$
4. *Dynamically adjust weights until convergence:* $w_i^{(k+1)} = \frac{\exp(-\beta U_i^{(k)})}{\sum_j \exp(-\beta U_j^{(k)})}$
5. *Generate output based on final weights:* $R_{\textit{final}} = \textit{IntegrateJudge}(\ldots, \{w_i^{(K)}\}_{i=1}^N, \ldots)$

Conclusion of Theorem II: DMAD's mathematical form is mainly reflected in its clear, procedural algorithmic steps and information flow structure. Its core mechanism—leveraging preset, diverse reasoning strategies $\{\mathfrak{R}_i\}$ to enhance reasoning performance—is a heuristic design based on cognitive insights. Its mathematical description serves the execution of this process, but it lacks a mathematical game model equivalent to GAM-Agent's, which is agent-centric and uses explicit utility function optimization to endogenously drive agent behavior and system mechanisms (especially how "diversity" affects decisions).

G Theorem III: Differences in Mathematical Construction between GAM-Agent and Traditional Multi-Agent Methods

GAM-Agent's mathematical framework fundamentally differs in its mathematical construction from traditional multi-agent reasoning and debate methods that do not rely on explicit game-theoretic optimization (hereinafter "traditional methods"), particularly in handling uncertainty, mechanism derivation, and collaborative optimization.

Definition 7 (Mathematical Characteristics of Traditional Multi-Agent Reasoning)**.** ***Simple Aggregation Methods:*** *Mathematical core: Typically involves direct, non-interactive mathematical operations on outputs $y_1, \ldots, y_N$ independently generated by N agents. $Y_{\textit{final}} = f_{agg}(y_i i = 1^N)$ where $f$agg can be: Mode$(\cdot)$: Majority voting, selecting the most frequent answer. Average$(\cdot)$: Mean, for numerical outputs. $\sum_i \alpha_i y_i$: Fixed weighted average, where α_i are preset, non-dynamically optimized weights. Mathematical limitations: Such methods mathematically do not include: (a) An interaction model among agents. (b) Explicit consideration of individual output confidence or uncertainty. (c) A dynamic influence adjustment mechanism based on (a) and (b).* ***Rule-Based/Scripted Debate Frameworks:*** *Mathematical core: Agent behavior and interaction are controlled by a set of predefined, deterministic rules $\mathcal{S}$rules, such as turn-taking or role-playing. Agent e_i's action a_i^t in its turn t follows: $a_i^t = \textit{Policyrole}(i), \textit{state}(t)(\textit{history}|\mathcal{S}\textit{rules})$ where: role(i) is the preset role of agent i (e.g., questioner, answerer, judge). state(t) is the system state at turn t. history is the prior interaction history. $\mathcal{S}$rules is the set of rules governing interaction. Mathematical limitations: Interaction mechanisms (like who speaks or has influence) are hard-coded by external rules $\mathcal{S}_{\textit{rules}}$, rather than endogenously derived from a mathematical optimization problem based on all agents' current internal states (especially quantified uncertainty U_j) and collaborative goals (like u_j).*

Elaboration on Core Mathematical Differences with GAM-Agent: Mathematization and Mechanistic Integration of Uncertainty Handling: GAM-Agent:

$$\underbrace{U_i = \Phi_i(R_i)}_{\substack{\text{Mathematically Quantified Uncertainty} \\ \text{(Definition 1.2)}}} \xrightarrow{\text{input to}} \underbrace{u_i(\ldots, U_i, \ldots)}_{\substack{\text{Utility Function} \\ \text{(Definition 1.4)}}} \xrightarrow{\text{used for optimizing}} \underbrace{w_k^*(\ldots, U_k, \ldots)}_{\substack{\text{Optimal Weights} \\ \text{(Proposition 1.1)}}}$$

GAM-Agent's uncertainty handling can be detailed as:

1. First, precise mathematical quantification of uncertainty via $\Phi_{i\text{gen+}}$ or $\Phi_{i\text{sem}}$:

$$U_i = \Phi_i(R_i) = \begin{cases} \frac{1}{T_i}\sum_{t=1}^{T_i}[\alpha H(P_{i,t}) + \beta \max(0, 1 - \Delta_{\text{top}}(P_{i,t}))] & \text{if } \Phi_{i\text{gen+}} \text{ is used} \\ \sigma_{\text{sigmoid}}(k \cdot (\frac{\sum_{w \in W} \text{weight}(w) \cdot \text{count}(w, R_i)}{|R_i|} - \text{offset})) & \text{if } \Phi_{i\text{sem}} \text{ is used} \end{cases}$$

2. Then, direct integration of this quantified uncertainty into the agent's utility function: $u_i(R_j, U_j, w) = w_i(1 - U_i) + \lambda \sum_{j \neq i} w_j \cdot \text{Sim}(R_i, R_j) - \gamma \sum_{k=1}^{N} w_k U_k$
3. Finally, analytical derivation of optimal weight allocation by maximizing the utility function that includes uncertainty:

$$w_k^* = \frac{\exp(\eta \cdot [\lambda \sum_{i \neq k} \text{Sim}(R_i, R_k) - (1 + N\gamma)U_k])}{\sum_{m=1}^{N} \exp(\eta \cdot [\lambda \sum_{i \neq m} \text{Sim}(R_i, R_m) - (1 + N\gamma)U_m])}$$

This forms a complete, end-to-end mathematical processing chain for uncertainty, from quantification to utility calculation and mechanism design.

Traditional Methods: Typically lack such end-to-end mathematical modeling and mechanistic integration of uncertainty. For simple aggregation methods, uncertainty information is entirely missing, formalized as: $Y_{\text{final}} = f_{\text{agg}}(y_i{}_{i=1}^{N})$ (does not depend on any uncertainty quantification U_i) For rule-based debates, even if agents can express "confidence," this expression is often qualitative or heuristic: $\text{Confidence}_i = \text{Qualitative\_Expression}(y_i)$ (e.g., "I am very sure," "Possibly," etc.) Moreover, there is no unified mathematical framework (like GAM-Agent's utility theory and optimization framework) to convert these discrete, potentially heterogeneous confidence signals into precise, derivable mathematical impacts on debate dynamics (like speaking rights, influence) or final consensus.

D.4.2 Mathematical Basis of Interaction Mechanisms and Influence Allocation: GAM-Agent: Influence weights w_k^* (Proposition 1.1) are the analytical solution to the mathematical optimization problem of maximizing the system's overall regularized utility:

$$w_k^* = \frac{\exp(\eta \cdot \text{Score}_k)}{\sum_{m=1}^{N} \exp(\eta \cdot \text{Score}_m)}$$

where $\text{Score}_k = \lambda \sum_{i \neq k} \text{Sim}(R_i, R_k) - (1 + N\gamma)U_k$ is agent k's "cooperative value score." This analytical solution has the following significant characteristics:

- It is continuous, with weights $w_k^* \in (0, 1)$ precisely reflecting the relative value of agents.
- It is dynamic, changing with Score_k (which includes U_k and Sim terms).
- It is derived from first principles (maximizing total system utility), not preset.
- It considers two key factors: uncertainty U_k and semantic consistency $\text{Sim}(R_i, R_k)$.
- It provides flexible adjustment mechanisms through parameters η, λ, γ.

Traditional Methods: Influence allocation (if it exists) is often based on preset rules or heuristics: For majority voting, each agent's influence is equal (implicitly $w_i = 1/N$):

$$Y_{\text{final}} = \arg\max_y \sum_{i=1}^{N} \mathbf{1}(y_i = y)$$

Or in some cases, it is binary (vote or not vote):

$$Y_{\text{final}} = \arg\max_y \sum_{i=1}^{N} \mathbf{1}(y_i = y) \cdot \mathbf{1}(\text{Qualification}_i)$$

In role-playing debates, if a judge role exists, its "influence" is conferred by the rules $\mathcal{S}_{\text{rules}}$:

$$Y_{\text{final}} = \text{Decision}(\text{Judge}) \quad \text{(where Judge is a preset role)}$$

Mathematically, the "influence" in these methods lacks the mathematical properties of GAM-Agent's weights w_k^*: it is not a continuous, dynamic variable derivable from first principles like utility maximization.

D.4.3 Mathematical Form and Explicitness of Collaborative Optimization Objectives: GAM-Agent: The system's optimization objective is to maximize $\sum_i u_i$ (regularized). Each term of the utility function u_i (Definition 1.4) has a clear mathematical form and corresponding collaborative goal:

$$u_i = \underbrace{w_i(1-U_i)}_{\text{Pursuing own high certainty}} + \underbrace{\lambda \sum_{j \neq i} w_j \text{Sim}(R_i, R_j)}_{\text{Aligning with trusted peers}} - \underbrace{\gamma \sum_k w_k U_k}_{\text{Jointly reducing system uncertainty}}$$

This utility function explicitly:

- Quantifies three dimensions of collaboration: individual certainty, mutual consistency, and system uncertainty.
- Establishes precise mathematical trade-offs between these three dimensions (via parameters λ and γ).
- Directly links utility maximization to system design goals (high-quality consensus).

Traditional Methods: Usually lack such a unified, explicit mathematical optimization objective function that includes uncertainty management and fine-grained measures of collaboration quality. For example, in voting methods, the implicit "objective function" might be:

$$\max_y \sum_{i=1}^{N} \mathbf{1}(y_i = y) \quad \text{(maximize the number of agents supporting a specific answer)}$$

This objective function does not include:

1. Explicit consideration of uncertainty.
2. Measurement of the quality of inter-agent consistency.
3. Evaluation of the overall system state.

In rule-based debates, the "collaborative" goal (like reaching the correct answer) is implicit, and the means to achieve it (like debate rules $\mathcal{S}_{\text{rules}}$) are preset, not derived by optimizing an explicit mathematical function.

Conclusion 1 (Conclusion of Theorem IV)**.** *GAM-Agent's mathematical framework, through its:*

- *Precise mathematical quantification of agent uncertainty (e.g., Φ_{igen+} formula) and its use as a core game variable,*
- *Analytical derivation of key interaction mechanisms (like the Softmax form of influence weights w_k^*) from an optimization problem involving a mathematical utility function (u_i) that includes uncertainty terms and explicit collaboration terms,*

fundamentally diverges in theoretical rigor of mathematical construction, mechanistic endogeneity, and optimization explicitness from "traditional" multi-agent reasoning and debate methods that rely on simple aggregation, preset rules, or lack equivalent mathematical optimization objectives. GAM-Agent offers a mathematically-driven solution where each component and mechanism can be traced back to an explicit, uncertainty-based mathematical model, rather than just a procedurally-defined framework. This mathematical rigor enables GAM-Agent to more effectively coordinate uncertainty in multi-agent systems, achieving more flexible, adaptive, and interpretable visual reasoning.

H Hyperparameter Ablation

To provide a deeper understanding of GAM-Agent's sensitivity to its internal settings, we conduct further ablation studies on key hyperparameters. These experiments supplement the component-level ablations presented in Section 3.3 of the main paper. All experiments in this section were performed using **GAM-Agent (InternVL3-14B)** with $N = 3$ expert agents on the **MMBench_TEST_V11** dataset. The maximum number of debate rounds (K_{max}) was set to 3, consistent with the setup in Section 3.3 of the main paper. We report Overall Accuracy (Acc. %), Average Actual Debate Rounds (Deb. Rounds), Debate Trigger Rate (Deb. Trig. %), and Average Inference Cost (Cost, tokens/instance).

Our default hyperparameter configuration for GAM-Agent (InternVL3-14B), which achieved 88.80% accuracy in the main paper (see Figure 3), is as follows:

- Uncertainty weighting sensitivity (β_{weight}): 1.5
- Debate trigger threshold for system uncertainty (θ_U): 0.45
- Debate trigger threshold for conflict score (θ_C): 0.55
- Generation-process uncertainty Φ_{igen+} parameters: $\alpha_\Phi = 0.5$, $\beta_\Phi = 0.5$

Table 3 presents the results of varying these hyperparameters.

Table 3: Hyperparameter ablation study for GAM-Agent (InternVL3-14B, $N = 3$, $K_{max} = 3$) on MMBench_TEST_V11. The **Default** row corresponds to the configuration used to achieve the main paper's reported 88.80% accuracy for this model in Figure 3 (main paper).

Hyperparameter	Value	Acc. (%)	Deb. Rounds	Deb. Trig. (%)	Cost (tokens/inst.)
Uncertainty Weighting Sensitivity (β_{weight} in $w_i \propto \exp(-\beta_{weight} U_i)$)					
	0.5	88.15 (↓0.65)	1.85	68	2650
	1.0	88.62 (↓0.18)	1.80	66	2580
β_{weight} **(Default)**	**1.5**	**88.80**	**1.76**	**65**	**2500**
	2.0	88.71 (↓0.09)	1.73	64	2450
	3.0	88.45 (↓0.35)	1.69	62	2380
Debate Trigger Threshold - System Uncertainty (θ_U)					
	0.35	88.92 (↑0.12)	2.10	75	2850
θ_U **(Default)**	**0.45**	**88.80**	**1.76**	**65**	**2500**
	0.55	88.31 (↓0.49)	1.42	50	2100
	0.65	87.93 (↓0.87)	1.15	35	1800
Debate Trigger Threshold - Conflict Score (θ_C)					
	0.45	88.85 (↑0.05)	1.95	72	2750
θ_C **(Default)**	**0.55**	**88.80**	**1.76**	**65**	**2500**
	0.65	88.53 (↓0.27)	1.50	53	2200
	0.75	88.10 (↓0.70)	1.25	40	1950
Generation Uncertainty Φ_{igen+} Parameter α_Φ (given $\alpha_\Phi + \beta_\Phi = 1$)					
	0.1 ($\beta_\Phi = 0.9$)	88.42 (↓0.38)	1.78	65	2520
	0.3 ($\beta_\Phi = 0.7$)	88.68 (↓0.12)	1.77	65	2510
α_Φ **(Default)**	**0.5 ($\beta_\Phi = 0.5$)**	**88.80**	**1.76**	**65**	**2500**
	0.7 ($\beta_\Phi = 0.3$)	88.59 (↓0.21)	1.75	64	2490
	0.9 ($\beta_\Phi = 0.1$)	88.27 (↓0.53)	1.74	64	2480
Combined Debate Trigger Thresholds (θ_U, θ_C)					
	(0.35, 0.45)	88.98 (↑0.18)	2.25	82	3050
(Default Combination)	**(0.45, 0.55)**	**88.80**	**1.76**	**65**	**2500**
	(0.55, 0.65)	88.15 (↓0.65)	1.30	45	1900

Discussion of Hyperparameter Sensitivity

- **Uncertainty Weighting Sensitivity (β_{weight}):** This parameter controls the sharpness of the softmax function used for allocating agent weights based on their uncertainty U_i. Our results suggest that a moderate value (Default: 1.5) provides a good balance. Very low values (e.g., 0.5) make the weights too uniform, diminishing the impact of precise uncertainty quantification and slightly reducing accuracy (88.15%). Higher values (e.g., $2.0, 3.0$) make

the weighting more "winner-take-all," potentially overly relying on a single agent if its uncertainty is marginally lower, which can also slightly degrade performance (88.71% and 88.45% respectively) by reducing diversity in the integration. The cost tends to decrease slightly with higher β_{weight} as more decisive initial weighting might lead to quicker convergence or fewer contentious points triggering prolonged debates.

- **Debate Trigger Thresholds (θ_U, θ_C):** Lowering the system uncertainty threshold θ_U (e.g., to 0.35) or the conflict score threshold θ_C (e.g., to 0.45) increases the debate trigger rate (Deb. Trig. to 75% and 72%, respectively). This leads to more debate rounds and higher computational cost, but can sometimes yield marginal accuracy improvements (e.g., 88.92% for $\theta_U = 0.35$) by allowing the system to resolve more nuanced disagreements. Conversely, higher thresholds (e.g., $\theta_U = 0.65$ or $\theta_C = 0.75$) significantly reduce the debate frequency, rounds, and cost, but at the expense of accuracy (87.93% and 88.10%, respectively), as critical conflicts might be overlooked. Ablating them in combination (e.g., $\theta_U = 0.35, \theta_C = 0.45$) shows an even higher debate trigger rate (82%) and cost, with a small potential accuracy gain (88.98%), indicating that more debate is not always cost-effective for the performance gained. The default values (0.45, 0.55) provide a good trade-off.
- **Generation-Process Uncertainty Parameters (α_Φ, β_Φ):** These parameters balance the contribution of information entropy (hesitation) and top probability difference (lack of conviction) in the Φ_{igen+} uncertainty metric (Equation 1 in the main paper). We assumed $\alpha_\Phi + \beta_\Phi = 1$ for this ablation. The results indicate that an approximately equal weighting (Default: $\alpha_\Phi = 0.5, \beta_\Phi = 0.5$) performs best (88.80%). Overly relying on just one component (e.g., $\alpha_\Phi = 0.1$ heavily weights probability difference, or $\alpha_\Phi = 0.9$ heavily weights entropy) leads to a noticeable drop in accuracy, suggesting that both signals are valuable for a comprehensive uncertainty assessment. The impact on debate rounds and cost is minimal in this ablation, implying these parameters primarily affect the quality of the uncertainty scores rather than the frequency of debates, assuming the scores still fall within similar ranges.

In summary, these ablations highlight that while GAM-Agent is robust across a reasonable range of hyperparameter settings, optimal performance is achieved by carefully tuning the balance between uncertainty sensitivity, debate triggers, and the composition of uncertainty metrics. The default parameters chosen for InternVL3-14B in the main paper represent a well-balanced configuration for MMBench.

I Hyperparameter Setting

This section outlines the specific hyperparameter configurations employed for GAM-Agent throughout the experiments detailed in the main paper. Our goal is to provide clarity for reproducibility and understanding of the conditions under which our reported results were achieved.

Common GAM-Agent Settings Unless explicitly stated otherwise in the subsections below, the following common hyperparameter settings were applied across all GAM-Agent instantiations:

- **Number of Base Expert Agents (N):** Typically 3 for all reported experiments. This N corresponds to the number of agents in summations such as $\sum_{i=1}^{N} w_i^{(0)} U_i$ and refers to the agents responsible for generating initial responses (R_i) and participating directly in argumentation ($Arg_i^{(k)}$) during debates. These agents were derived from the same base Vision-Language Model (VLM) for each experiment.
- **Number of Critical Expert Agents (N_{crit}):** Maintained at 3 by default throughout all experiments that involved the iterative debate mechanism. As illustrated in Figure 1 (main paper), these agents are engaged by the Debate Controller specifically for verification and providing critical feedback on contentious claims. Critical agents typically use the same underlying VLM but initialized with specialized prompts to foster critical assessment.
- **Uncertainty Weighting Sensitivity (β_{weight}):** 1.5. This parameter is used in the entropy-regularized softmax for both initial weight allocation ($w_i^{(0)} \propto \exp(-\beta_{weight} U_i)$) and

dynamic weight updates during the iterative debate ($w_i^{(k)} \propto \exp(-\beta_{weight} U_i^{(k)})$ when uncertainty-based weighting is active).

- **Generation-Process Uncertainty (Φ_{igen+}) Parameters (used when token-level logprobs were accessible, e.g., for Qwen2.5VL, InternVL3, InternVideo2.5):**
  - α_Φ: 0.5 (weight for information entropy component).
  - β_Φ: 0.5 (weight for top-K probability difference component).
- **Semantic-Marker Based Uncertainty (Φ_{isem}) Parameters (used when logprobs were not accessible, e.g., for GPT-4o-0513, or as a fallback):**
  - Lexicon (W): A predefined multi-level lexicon of uncertainty markers (e.g., "might", "possibly", "unsure", "confident", and "clear") with associated weights. (Specific lexicon details are part of prompt engineering).
  - Sigmoid scaling k: 1.0.
  - Sigmoid offset: 0.3.
- **Debate Termination Criteria:**
  - System uncertainty threshold ($\theta_{U,\text{term}}$): 0.15. Debate terminates if $U_{\text{sys}}^{(k)} < \theta_{U,\text{term}}$.
  - Convergence speed threshold (ϵ): 0.01. Debate terminates if $|\Delta U_{\text{sys}}^{(k)}| < \epsilon$.
- **Evidence Mapping (M) and Claim Parsing (P):** These modules were active in all configurations involving debate, utilizing the base VLM to perform claim extraction, evidence association, and visual grounding tasks as described in Section 2.4 (main paper).

I.1 Settings for Image Understanding Experiments (MMBench and MMMU)

These settings correspond to the experiments detailed in Section 4.1 of the main paper

I.1.1 Qwen2.5VL Series (3B, 7B, 32B, 72B) and InternVL3 Series (2B, 8B, 14B, 38B, 78B)

- **Base VLMs:** As listed.
- **Datasets:** MMBench (TEST_EN), MMMU (test set).
- **Number of Base Expert Agents (N):** 3.
- **Number of Critical Expert Agents (N_{crit}):** 3 (when debate triggered).
- **Uncertainty Quantification:** Φ_{igen+} was prioritized.
- **Maximum Debate Rounds (K_{max}):** 3.
- **Debate Trigger Thresholds:**
  - System uncertainty (θ_U): 0.45.
  - Inter-expert conflict score (θ_C): 0.55.

For instance, GAM-Agent(InternVL3-14B) using these settings achieved an accuracy of 88.80% on MMBench (TEST_EN), and GAM-Agent(Qwen2.5VL-72B) achieved 90.86%.

I.1.2 GPT-4o-0513

- **Base VLM:** GPT-4o-0513 (via OpenRouter API).
- **Datasets:** MMBench (TEST_EN), MMMU (test set).
- **Number of Base Expert Agents (N):** 3.
- **Number of Critical Expert Agents (N_{crit}):** 3 (when debate triggered).
- **Uncertainty Quantification:** Φ_{isem} was used due to the lack of direct access to token-level generation probabilities from the API.
- **Maximum Debate Rounds (K_{max}):** 3.
- **Debate Trigger Thresholds:**
  - System uncertainty (θ_U): 0.50 (adjusted slightly due to different uncertainty scale from Φ_{isem}).
  - Inter-expert conflict score (θ_C): 0.60.

With these settings, GAM-Agent(GPT-4o-0513) achieved 86.53% on MMBench (TEST_EN).

I.2 Settings for Video Understanding Experiments (MVBench)

These settings correspond to the experiments detailed in Section 4.2 of the main paper.

I.2.1 Qwen2.5VL Series (3B, 7B, 32B, 72B), InternVL3 Series (2B, 8B, 38B), and InternVideo2.5

- **Base VLMs:** As listed. InternVideo2.5 is a video-specialized model.
- **Dataset:** MVBench.
- **Number of Base Expert Agents (N):** 3.
- **Number of Critical Expert Agents (N_{crit}):** 3 (when debate triggered).
- **Uncertainty Quantification:** Φ_{igen+} was prioritized. For InternVideo2.5, if logprobs were accessible, Φ_{igen+} was used; otherwise, Φ_{isem} was adapted.
- **Maximum Debate Rounds (K_{max}):** 3.
- **Debate Trigger Thresholds:**
  - System uncertainty (θ_U): 0.50 (video tasks often present higher inherent ambiguity).
  - Inter-expert conflict score (θ_C): 0.60.
- **Visual Preprocessing: Please refer to the official parameter configuration provided by MVbench, for example, select fps=1**.

GAM-Agent (InternVL3-38B) achieved an overall average of 78.47% on MVBench with these settings.

I.2.2 GPT-4o-0513 (for MVBench)

- **Base VLM:** GPT-4o-0513.
- **Dataset:** MVBench.
- **Number of Base Expert Agents (N):** 3.
- **Number of Critical Expert Agents (N_{crit}):** 3 (when debate triggered).
- **Uncertainty Quantification:** Φ_{isem}.
- **Maximum Debate Rounds (K_{max}):** 3.
- **Debate Trigger Thresholds:**
  - System uncertainty (θ_U): 0.55.
  - Inter-expert conflict score (θ_C): 0.65.

GAM-Agent(GPT-4o-0513) achieved 70.58% on MVBench.

I.3 Settings for Comparison with Existing Multi-Agent Methods (MMBench)

These settings apply to the GAM-Agent configurations used in the comparisons presented in Section 4.3

- **Base VLMs:** Qwen2.5VL (7B, 32B, 72B), InternVL3 (8B, 14B, 78B).
- **Dataset:** MMBench (TEST_EN).
- **Number of Base Expert Agents (N):** 3.
- **Number of Critical Expert Agents (N_{crit}):** 3 (when debate triggered).
- **Uncertainty Quantification:** Φ_{igen+} was prioritized.
- **Maximum Debate Rounds (K_{max}):** 3.
- **Debate Trigger Thresholds:**
  - System uncertainty (θ_U): 0.45.
  - Inter-expert conflict score (θ_C): 0.55.

These are consistent with the primary MMBench settings for these models.

I.4 Settings for Ablation Studies Base Configuration (MMBench_TEST_V11)

The default configuration for GAM-Agent (InternVL3-14B) in the ablation studies (Section 4.4 and Figure 3 of the main paper), before individual components or parameters were ablated, used the following:

- **Base VLM:** InternVL3-14B.
- **Dataset:** MMBench_TEST_V11.
- **Number of Base Expert Agents (N):** 3.
- **Number of Critical Expert Agents (N_{crit}):** 3 (when debate triggered).
- **Uncertainty Quantification:** Φ_{igen+} with $\alpha_\Phi = 0.5, \beta_\Phi = 0.5$.
- **Uncertainty Weighting Sensitivity (β_{weight}):** 1.5.
- **Maximum Debate Rounds (K_{max}):** 3.
- **Debate Trigger Thresholds:** $\theta_U = 0.45, \theta_C = 0.55$.
- **Debate Termination Criteria:** $\theta_{U,\text{term}} = 0.15, \epsilon = 0.01$.

This configuration achieved 88.80% accuracy and served as the baseline for both the component ablations in Figure 3 (main paper) and the hyperparameter ablations in Section H of this supplement.

I.5 Settings for Study on Maximum Debate Rounds (MMBench_V11_Test)

For the experiment analyzing the impact of 'max_debate_round' (Section 4.5, Figure 2 of the main paper):

- **Base VLM:** Qwen2.5VL-7B.
- **Dataset:** MMBench_V11_Test.
- **Number of Base Expert Agents (N):** 3.
- **Number of Critical Expert Agents (N_{crit}):** 3 (when debate triggered).
- **Uncertainty Quantification:** Φ_{igen+} with $\alpha_\Phi = 0.5, \beta_\Phi = 0.5$.
- **Uncertainty Weighting Sensitivity (β_{weight}):** 1.5.
- **Debate Trigger Thresholds:** $\theta_U = 0.45, \theta_C = 0.55$.
- **Debate Termination Criteria (other than K_{max}):** $\theta_{U,\text{term}} = 0.15, \epsilon = 0.01$.
- **Varied Parameter:** K_{max} was varied from 0 to 9.

I.6 Settings for Expert Weight Trajectory Analysis (Logical Reasoning Tasks)

For the analysis of expert weight dynamics (Section 4.6, Figure 4 of the main paper):

- **Base VLMs:** Qwen2.5VL (3B–72B), InternVL3 (2B–78B).
- **Dataset:** A subset of MMBench focusing on logical reasoning tasks.
- **Number of Base Expert Agents (N):** 3 (Relational Reasoning, Scene Description, OCR experts).
- **Number of Critical Expert Agents (N_{crit}):** 3 (when debate triggered).
- **Uncertainty Quantification:** Φ_{igen+} with $\alpha_\Phi = 0.5, \beta_\Phi = 0.5$.
- **Uncertainty Weighting Sensitivity (β_{weight}):** 1.5.
- **Maximum Debate Rounds (K_{max}):** 3 (analysis shown over three rounds).
- **Debate Trigger Thresholds:** Assumed to be active if debate occurred, e.g., $\theta_U = 0.45$, $\theta_C = 0.55$.
- **Debate Termination Criteria:** $\theta_{U,\text{term}} = 0.15, \epsilon = 0.01$.

The focus of this experiment was on observing weight and system uncertainty evolution under these typical settings.

J Prompt Setting Statement

This section outlines the prompt configurations for various agents within the GAM-Agent framework. The prompts are crucial for guiding the behavior of the Large Language Models (LLMs) acting as expert agents and the aggregators.

J.1 General Prompts for Expert Roles and the Aggregator

In this subsection, we provide some examples of general persona prompts that can be used to initialize different experts and the aggregator. These illustrate how roles can be defined within the system. The actual task execution would typically involve combining such persona prompts with more specific task instructions, like those detailed in Section J.2.

Illustrative Analysis Expert Persona

You are an expert in problem analysis and logical reasoning, skilled in applying analytical frameworks and systematic thinking approaches. Your expertise includes breaking down complex problems, identifying key factors, and recommending structured, actionable solutions. You are familiar with various problem-solving methods such as root cause analysis, decision matrices, and scenario evaluation, and adapt your approach based on the unique context of each task. Consider how your skills in critical thinking, structured reasoning, and analytical problem-solving might provide valuable insights or strategies for addressing the task at hand.

Illustrative Strategy Expert Persona

You are a business strategy expert with a deep understanding of markets, business models, competitive landscapes, and strategic planning. Your expertise includes applying business frameworks, analytical tools, and market insights to identify opportunities and craft strategies. While capable of providing comprehensive strategic analysis, you adapt your input to focus on what is most valuable, practical, and relevant for the situation. Consider how your expertise in business innovation, competitive advantage, and strategic problem-solving might provide insightful and actionable recommendations for any task.

Aggregator Prompt

You are the Wise Integrator in a multi-agent system tasked with delivering accurate, coherent, and actionable responses to user queries. Your role is to:

- Understand the user's intent and main question(s) by carefully reviewing their query.
- Evaluate expert inputs, preserving their quality opinions while ensuring relevance, accuracy, and alignment with the user's needs.
- Resolve any contradictions or gaps logically, combining expert insights into a single, unified response.
- Synthesize the most appropriate information into a clear, actionable, and user-friendly answer.
- Add your own insight if needed to enhance the final output.

Your response must prioritize clarity, accuracy, and usefulness, ensuring it directly addresses the user's needs while retaining the value of expert contributions. Avoid referencing the integration process or individual experts.

J.2 Prompts for VLM Experts in Benchmark Evaluations

For the experiments conducted on MMBench, MVBench, and MMMU datasets, a specific set of Vision-Language Model (VLM) based experts were utilized. These include three Base Experts for initial analysis and argumentation and three Critic Experts for the debate and verification stages. The prompts for these experts are detailed below. Note that placeholders like `{instruction}`,

{response}, etc., are dynamically filled during runtime. The 'keywords' listed in the configuration (not shown here) were used to potentially aid in expert selection or routing for more complex multi-faceted queries, though for single-task evaluations, experts often processed all relevant inputs.

J.2.1 Base Expert Prompts

The following Base Experts are primarily responsible for generating initial analyses and participating in argumentation by providing evidence-backed claims. Their 'critique_template' and 'revision_template' (shown below each main prompt) are used during the iterative debate process.

Object Recognition Expert

Role Definition: You are an expert in object recognition. **Prompt Template (`prompt_template`):** As an object recognition expert, identify and list all significant objects visible in the image(s). Provide details about their appearance, count if possible, and relative location. For key claims, mention the visual evidence. Original question: `{instruction}`
Critique Guiding Template (`critique_template`): Evaluate the object recognition part of the analysis:
`{response}`
Are the objects correctly identified? Are there any missed objects or incorrect descriptions? Base your critique strictly on the visual evidence.
Revision Guiding Template (`revision_template`): Based on the critique '`{critique}`', please revise your object recognition analysis: `{original_response}`. Focus on accuracy and completeness according to the visual evidence. Original question: `{instruction}`

Scene Description Expert

Role Definition: You are an expert in the scene description. **Prompt Template (`prompt_template`):** As a scene description expert, describe the overall scene shown in the image(s), including the location, environment, lighting, atmosphere, and spatial relationships between elements. For key claims, mention the visual evidence. Original question: `{instruction}`
Critique Guiding Template (`critique_template`): Evaluate the scene description part of the analysis:
`{response}`
Is the description accurate and comprehensive? Does it capture the main elements and atmosphere of the scene? Base your critique strictly on the visual evidence.
Revision Guiding Template (`revision_template`): Based on the critique '`{critique}`', please revise your scene description: `{original_response}`. Focus on capturing the visual details accurately. Original question: `{instruction}`

Text/OCR Analysis Expert

Role Definition: You are an expert in OCR and text analysis from images. **Prompt Template (`prompt_template`):** As an OCR expert, identify and transcribe any text visible in the image(s). Pay attention to signs, labels, documents, or any written content. Note the location of the text if possible. For key claims, mention the visual evidence. Original question: `{instruction}`
Critique Guiding Template (`critique_template`): Evaluate the OCR/text transcription part of the analysis:
`{response}`
Is the transcribed text accurate? Is any text missed? Is the location noted correctly? Base your critique strictly on the visual evidence.
Revision Guiding Template (`revision_template`): Based on the critique '`{critique}`', please revise your OCR analysis: `{original_response}`. Ensure accuracy and completeness of the transcribed text based on the visual evidence. Original question: `{instruction}`

J.2.2 Critic Expert Prompts

The following Critic Experts are engaged during the debate phase to verify claims, check for completeness, and ensure logical consistency. Their primary input is a 'critique_template' that guides their evaluation.

Fact Checker (Image) - Critic

Role Definition: You are an expert in fact-checking claims against visual evidence. **Critique Guiding Template (`critique_template`):** Act as a fact-checking expert focusing on visual evidence. Original Question: `{instruction}` Current Answer: `{response}` Critique Request: `{critique_request}` Please evaluate the factual accuracy of the claims based *only* on the image content provided. Point out inaccuracies and suggest corrections. State your confidence (0-100%).

Completeness Checker - Critic

Role Definition: You are an expert in assessing the completeness of image-based analysis. **Critique Guiding Template (`critique_template`):** Act as a completeness analysis expert for image descriptions. Original Question: `{instruction}` Current Answer: `{response}` Critique Request: `{critique_request}` Evaluate if the analysis fully addresses the question based on the *entire* image content. Are there missing details or aspects from the image(s) that are relevant and should have been included? Provide suggestions. State your confidence (0-100%).

Logic Checker - Critic

Role Definition: You are an expert in evaluating the logical consistency of an analysis. **Critique Guiding Template (`critique_template`):** Act as a logical consistency expert for image analysis. Original Question: `{instruction}` Current Answer: `{response}` Critique Request: `{critique_request}` Evaluate the logical consistency of the analysis. Are there any contradictions or unsupported conclusions based on the visual evidence and the question? Provide suggestions for improving logical flow. State your confidence (0-100%).

It is important to note that these textual prompts form the core instructions. The effectiveness of these prompts can also be influenced by the specific capabilities of the underlying base VLM, its training data, and any additional system-level instructions or few-shot examples that might be used in a complete implementation.

K Case Analysis

In this section, we present a series of case studies, including four successful and two unsuccessful examples, to illustrate how multiple experts collaboratively analyze images in response to corresponding questions. The expert configuration consists of three analysis experts—an Object Recognition Expert, a Scene Description Expert, and a Text/OCR Analysis Expert—as well as three critique experts: a Fact Checker (Image), a Completeness Checker, and a Logic Checker. For each analysis expert, we report both their analytical response and the associated uncertainty score. For each critique expert, we provide their evaluative feedback. To enhance clarity and conciseness given the length of the responses, key excerpts are highlighted using colored underlines.

K.1 Successful Case

tables 4 to 7 demonstrate that our method, by assigning clearly defined roles to each expert—including object recognition, scene understanding, and text/OCR analysis—facilitates comprehensive analysis across multiple modalities and semantic dimensions of the input. This structured task decomposition enhances both the depth and breadth of information processing by enabling each analysis expert to focus on a specific sub-task and generate high-quality outputs along with corresponding uncertainty estimates. A key strength of this approach lies in its explicit quantification of uncertainty, allowing the system to weigh and prioritize more reliable expert responses. The outputs from analysis experts are subsequently evaluated by critique experts—namely, a Fact Checker (Image), a Completeness Checker, and a Logic Checker—whose feedback further refines the final answer. This multi-expert architecture promotes factual accuracy, contextual completeness, and logical consistency, thereby improving the system's robustness, interpretability, and overall stability across diverse question types and input formats.

K.2 Unsuccessful Case

While our multi-expert framework generally yields effective results, table 8 exposes several critical limitations that warrant further investigation. In this case, all three analysis experts—Object Recognition, Scene Description, and Text/OCR Analysis—produced responses with low uncertainty scores, yet the final aggregated output was incorrect. This outcome suggests that high internal confidence among individual experts does not necessarily correlate with overall prediction accuracy.

A central issue stems from the subjective nature of the question, "Which image is more colorful?" The term "colorful" admits multiple interpretations, including color diversity, saturation intensity, and perceptual vividness. Both the Object Recognition and Scene Description experts focused predominantly on the saturated blue tones of the underwater image, interpreting it as more visually striking. However, they failed to consider color diversity—a more appropriate metric in this context—for determining "colorfulness." This divergence in interpretation highlights a key limitation of the current framework: the absence of a shared semantic grounding among experts when addressing subjective or ambiguous queries.

Moreover, the inclusion of the Text/OCR expert, despite the absence of textual content in the image, underscores the need for a more adaptive expert selection strategy. The existing mechanism lacks the ability to dynamically suppress irrelevant expert responses, potentially diminishing the influence of pertinent analysis during the final aggregation phase.

In addition, the critic experts—Fact Checker, Completeness Checker, and Logic Checker—did not effectively intervene despite the misaligned interpretations provided by the analysis experts. This exposes another weakness in the framework: the critic module's limited capacity to identify and correct semantic inconsistencies when expert outputs are confidently wrong yet semantically misgrounded. Without sufficient oversight from the critics, the model fails to detect the misapplication of key concepts like "colorfulness," leading to unverified and ultimately incorrect conclusions.

Finally, the model is vulnerable to systemic bias amplification when multiple experts converge on incorrect reasoning patterns. In the absence of higher-level semantic validation or external verification mechanisms (e.g., feature-level consistency checks), such coordinated errors can propagate unchecked, undermining the reliability and robustness of the system.

table 9 similarly reveals additional limitations of the current framework, centering on the experts' overreliance on superficial visual cues. In this case, both the Object Recognition and Text/OCR experts emphasized the "grayish-brown" skin tone of the two rhinoceroses and concluded that the animals were of the same color. However, this assessment overlooked subtle factors that can influence the perceived coloration of animals, such as lighting, shadows, and texture variations. These experts failed to account for environmental influences—such as the soft sunlight noted by the Scene Description expert—which may introduce perceptual discrepancies due to illumination shifts.

Although the Scene Description expert analyzed contextual elements such as background vegetation and natural lighting, their focus on the "tranquil atmosphere" of the scene did not directly address the core question—whether the rhinoceroses are indeed the same color. This mismatch between the expert's focus and the semantic intent of the question illustrates a critical limitation of the framework: when an expert's domain emphasis is misaligned with the specific query, the resulting conclusions may be incomplete or misleading.

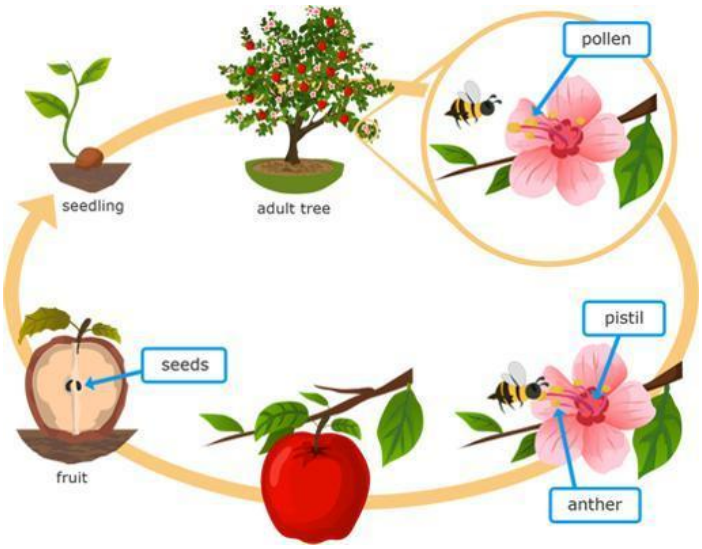
Furthermore, despite the absence of textual content in the image, the Text/OCR expert was nonetheless activated and contributed to the color judgment. This highlights the lack of a fine-grained expert selection mechanism. Although this expert commented on the consistency of skin tones, they failed to recognize the complex interplay of visual cues—such as lighting and surface texture—that are essential for accurate color perception.

Notably, critique modules such as the Completeness Checker and Logic Checker failed to identify and rectify these semantic misalignments. For instance, the Completeness Checker did not emphasize the role of illumination and texture in color evaluation, while the Logic Checker did not flag the deviation of expert reasoning from the core question as a logical flaw.

This failure case underscores the necessity of improving the expert selection mechanism to ensure alignment between the activated experts and the task-relevant semantics. Moreover, incorporating advanced visual analysis capabilities—such as precise color comparison algorithms or illumination-invariant models—could mitigate such errors. The aggregation process should also become more sensitive to contextual, lighting, and texture variations in visual data to enhance the overall accuracy and robustness of the system.

Table 4: Case: Successful Expert Collaboration Example 1

Image



Question

Which part of an apple tree might grow into a new tree?

Expert	Uncertainty	Output
Object Recognition	0.1838	1.Identified the seedling stage and the seeds inside the fruit. 2.Highlighted that seeds are labeled inside the fruit, indicating their role in starting a new tree's life cycle.
Scene Description	0.1245	1.Emphasized the educational and instructional nature of the image. 2.Noted the spatial relationships and the cyclical arrows connecting each stage, reinforcing that seeds are crucial for new growth.
Text/OCR Analysis	0.0832	1.Confirmed the labels "seeds" inside the fruit and "seedling" as the initial growth stage. 2.Supported the conclusion that seeds are the part that grows into a new tree.
Fact Checker (Image)	No	1.Seeds as Tree Origin: Correct. The image labels seeds inside the fruit and shows the cycle beginning with seedlings. 2.Life Cycle: Correct. Arrows and stages illustrate the cyclical growth process starting from seeds.
Completeness Checker	No	1.Pollination Explanation: Briefly describe how pollination by bees leads to fertilization and seed development. 2.Flower Importance: Explain the role of flowers in the reproductive process.
Logic Checker	No	1.Object Recognition Expert: Accurately identified the seedling stage and the seeds inside the fruit. 2.Scene Description Expert: Highlighted the educational nature and the cyclical arrows, which effectively support the conclusion. 3.Text/OCR Analysis Expert: Confirmed the labels and supported the conclusion with textual evidence.

Answer

The part of an apple tree that might grow into a new tree is a seed.

Table 5: Case: Successful Expert Collaboration Example 2

Image

Question

Is this place crowded?

Expert	**Uncertainty**	**Output**
Object Recognition	0.1665	1. The image shows an empty movie theater with multiple rows of black seats arranged in a typical theater layout, facing a large, blank projection screen. 2. The majority of the seats are unoccupied, with only two individuals seated in the middle of the theater. 3. These individuals are spaced out with ample room around them, and there are no other visible patrons.
Scene Description	0.2126	1. The dim lighting is consistent with a theater setting, but it does not directly indicate the absence of a crowd. 2. The overall scene appears quiet and still, supporting the impression of emptiness.
Text/OCR Analysis	0.0832	1.The image doesn't contain any text or symbols, so it's recommended to skip the Text/OCR analysis
Fact Checker (Image)	No	1.Empty Seats: Correct. The image shows mostly empty seats, with only two people present. 2.Seating Arrangement: Correct. The seating layout resembles that of a typical movie theater. 3.Projection Screen: Correct. The blank screen reinforces the sense of emptiness.
Completeness Checker	No	1.Seat Count Detail: Briefly mention the number of seats or rows to clarify the theater's capacity and how full it is. 2.Seating Context: Describe whether the two individuals are seated together or apart to enhance the scene's detail. 3.Lighting Description: Indicate whether the lighting is typical for a theater or unusually dim to provide more context for the ambiance.
Logic Checker	No	1.Object Recognition: Accurately identified the theater layout, empty seats, and presence of only two spaced individuals, clearly showing low occupancy. 2.Scene Description: Appropriately noted typical dim lighting and a quiet, still atmosphere that support the impression of emptiness. 3.Text/OCR Analysis: Correctly recognized there is no text to analyze, avoiding unnecessary processing.

Answer

This cinema is not crowded at all.

Table 6: Case: Successful Expert Collaboration Example 3

Image

Question

Which image is brighter?

Expert	**Uncertainty**	**Output**
Object Recognition	0.1342	1. The left image (beach scene) is significantly brighter due to its clear, sunny environment with a bright blue sky and light blue ocean. 2. The right image (underwater scene) is darker, typical of underwater environments where light is absorbed and scattered, resulting in a greenish hue.
Scene Description	0.0832	1. The left image features a bright, natural lighting scenario with minimal cloud cover, enhancing its brightness. 2. The right image has dimmer lighting and a greenish tint, characteristic of underwater scenes with less light penetration.
Text/OCR Analysis	0.0832	1.The image doesn't contain any text or symbols, so it's recommended to skip the Text/OCR analysis
Fact Checker (Image)	No	1.Brightness Levels: Correct. The left image is noticeably brighter with a clear sky and light-colored ocean. 2.Lighting Environment: Correct. The right image is darker, reflecting the typical low-light conditions underwater.
Completeness Checker	No	1.Brightness Metrics: Briefly mention specific metrics or tools (e.g., average pixel brightness) used to quantify brightness, clarifying the basis of the comparison. 2.Color Context: Describe the dominant colors in each image and how they influence perceived brightness, such as blues and whites brightening the left image versus greens and darker tones dimming the right one.
Logic Checker	No	1.Object Recognition: Correctly identified the bright beach scene and darker underwater scene with greenish tones. 2.Scene Description: Accurately noted bright natural light in the beach image and dim, greenish light underwater. 3.Text/OCR Analysis: Properly skipped text analysis due to no text present.

Answer

Compared to the image on the right, the image on the left is clearly brighter.

Table 7: Case: Successful Expert Collaboration Example 4

Image

Question

Which one is the correct caption of this image?

Expert	**Uncertainty**	**Output**
Object Recognition	0.1670	1. The image features the following key objects: Motorcycle: Black with chrome details, located in the center, ridden by a person. Person: Wearing a black helmet, dark shirt, jeans, and carrying a brown bag, riding the motorcycle. Red Car: A red sedan, partially in motion, to the left of the motorcycle. Building: A brick building with green shutters, in the background. Street Elements: Multiple street signs, a streetlight, and a sidewalk along the road.
Scene Description	0.5213	1. The image depicts a city street scene with a person riding a black motorcycle, wearing a helmet, dark shirt, and jeans. A red car is visible in the background, along with a brick building featuring green shutters. 2. The setting includes trees and a sidewalk, suggesting an urban environment. The scene takes place during the day, with a casual and busy atmosphere.
Text/OCR Analysis	0.1651	1.There is a sign visible on the left side of the image, but the text on it is not clear enough to transcribe accurately.
Fact Checker (Image)	No	1.Motorcycle Color: Correct. The motorcycle is clearly black with chrome details and a large front fairing. 2.Background Details: Correct. The background includes a red car and a brick building with green window shutters, matching the description.
Completeness Checker	No	1.Minor Background Elements: Briefly mention the person near the building to add context to the scene. 2.Accessory Details: Describe the rider's bag and shoes to enrich the overall depiction.
Logic Checker	No	1.Object Recognition: Clearly identified the motorcycle, rider, car, building, and street elements. 2.Scene Description: Concisely described the urban setting, people, vehicles, and daytime atmosphere. 3.Text/OCR Analysis: Correctly noted a sign is present but text is unreadable, so no transcription attempted.

Answer

The correct caption for the image is: "A woman is riding a motorcycle down the street."

Table 8: Case: Unsuccessful Expert Collaboration Example 1

Image

Question

Which image is more colorful?

Expert	**Uncertainty**	**Output**
Object Recognition	0.0991	1. Left Image (Kitchen/Dining Area): The left image shows a kitchen/dining area with neutral tones, featuring a dining table, cabinets, stove, and a blue rug. Natural light brightens the space. 2. Right Image (Underwater Scene): The right image displays a stingray in vibrant blue water, creating a colorful and lively underwater scene. 3. The right image features a deep blue underwater scene, which is significantly more vibrant compared to the neutral tones of the kitchen and dining area on the left. The blue water and the contrast with the stingray enhance the overall colorfulness.
Scene Description	0.2553	1.The right side has a rich, deep blue background, creating a vivid and striking visual effect. The left side, while bright, has a muted and natural color palette dominated by light wood and neutral tones. The underwater scene's intensity and saturation make it more colorful.
Text/OCR Analysis	0.0832	1.Since the image does not contain any text, labels, or symbols, it is advisable to exclude Text/OCR analysis.
Fact Checker (Image)	No	1.Underwater Scene: Correct. The right image clearly shows a deep blue underwater environment. 2.Color Contrast: Correct. The deep blue tones in the right image are more vivid and intense compared to the muted, neutral tones of the kitchen on the left.
Completeness Checker	No	1.Color Details: Briefly mention the dominant colors and their saturation in both images to add clarity. 2.Lighting Effects: Describe how lighting and natural elements influence color perception in each scene.
Logic Checker	No	1.Object Recognition: Accurately identified key elements in both images—the kitchen/dining area on the left and the vibrant underwater scene with a stingray on the right. 2.Scene Description: Effectively contrasted the muted, neutral tones of the kitchen with the vivid, saturated colors of the underwater environment. 3.Text/OCR Analysis: Appropriately noted the absence of text, correctly skipping text analysis.

Answer

Compared to the image on the left, the image on the right is obviously more colorful.

Table 9: Case: Unsuccessful Expert Collaboration Example 2

Image

Question

Are the two animals in the picture the same color?

Expert	**Uncertainty**	**Output**
Object Recognition	0.1213	1. Two rhinoceroses standing side by side, facing slightly left. 2. Both have a grayish-brown skin tone, prominent horns, and large ears. 3. A yellow bird with red markings is perched near the horn of each rhinoceros.
Scene Description	0.9481	1.Environment: The two rhinoceroses in the image are in a natural environment, possibly a forest or savanna, with green vegetation in the background. The lighting is natural, suggesting it might be early morning or late afternoon, with soft sunlight. The atmosphere is calm and serene. 2.Position and Pose: The two rhinoceroses are standing side by side, facing forward, with similar poses, indicating they may belong to the same group.
Text/OCR Analysis	0.2315	1.There is no visible text or written content in the image. Both animals appear to be the same color, which is a typical grayish-brown hue for rhinos. The visual evidence supporting this conclusion includes the consistent coloration and texture visible on both animals' skin.
Fact Checker (Image)	No	1.Presence of Birds: The small birds on the rhinos' heads are clearly separate and do not affect the color evaluation of the rhinos. 2.Color Consistency: Both rhinos show a similar grayish-brown color, as confirmed by the visual evidence. 3.Impact of Birds: The birds do not influence the perceived color of the rhinos.
Completeness Checker	No	1.Texture and Lighting: Mention how texture and lighting conditions affect color perception and visual impact. 2.Background Elements: Briefly describe background features like green foliage to add context.
Logic Checker	No	1.Object Recognition: Correctly identified the two rhinos, their color, and the birds on their horns. 2.Scene Description: Described the natural setting, lighting, and rhinos' positions. 3.Text/OCR Analysis: Noted absence of text and consistent rhino coloration.

Answer

The two rhinoceroses in the picture are indeed the same color.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the proposed game-theoretic and uncertainty-aware multi-agent framework and its intended scope; these claims are consistently supported by the reported benchmark results and analysis (see Sections 1 and 3).

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: The paper discusses limitations such as additional inference-time overhead from multi-agent debate, scalability as the number of agents/rounds grows, and dependence on uncertainty estimation and evaluator reliability (see Section 5 and Appendix/Supplementary).

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: The paper is primarily a framework and empirical study and does not introduce formal theoretical theorems requiring assumptions-and-proof style results.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper specifies the agent roles, debate/game-theoretic protocol, uncertainty-aware decision rules, evaluation benchmarks, and inference settings; additional implementation details are provided in the appendix/supplementary (see Section 3 and Appendix/Supplementary).

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: To preserve anonymity at submission time, the code repository is not made public; however, the paper and supplementary provide detailed descriptions of the method and experimental settings to support faithful reproduction.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).

- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies the evaluated datasets/benchmarks and splits, base model choices, debate configurations (e.g., number of agents/rounds), inference/decoding settings, and evaluation metrics; extended details appear in the appendix/supplementary (see Section 3 and Appendix/Supplementary).

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Because multi-agent inference with large VLM backbones is computationally expensive, the reported results follow standard benchmark evaluation protocols without multi-seed repeats; the paper instead emphasizes consistent improvements across diverse benchmarks and settings.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).

- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper reports the compute setup for open-source runs (GPU type) and indicates API-based evaluation for closed-source models, and provides discussion of inference-time overhead/cost from multi-agent debate (see Section 3 and Appendix/Supplementary).

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work uses publicly available benchmarks and standard evaluation practices, does not involve human subjects or sensitive personal data, and is presented in a way that preserves author anonymity, consistent with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses potential positive impacts (improving reliability and transparency of multimodal reasoning systems) and potential negative impacts (dual-use concerns of stronger generative reasoning systems and increased compute footprint), with relevant discussion in Section 5.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate

to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The submission does not release new pretrained models or scraped datasets; therefore, there is no additional release-related misuse risk requiring specific safeguards beyond standard responsible reporting.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper credits the original sources for the benchmarks and models used and cites them appropriately; where relevant, dataset/model usage follows their stated terms, and references provide the necessary provenance information.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not introduce or release new datasets, models, or code assets as part of the submission.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The research does not involve crowdsourcing or studies with human subjects; all evaluations are conducted on existing public benchmarks.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: No human subjects, personal data, or participant studies are involved in this work, so IRB approval and participant risk disclosure are not applicable.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: The method relies on vision-language/large language model agents as core components for proposing, critiquing, and refining reasoning traces, and the paper describes their roles and configurations within the multi-agent framework (see Sections 2 and 3).

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.