Securing the Floor and Raising the Ceiling: A Merging-based Paradigm for Multi-modal Search Agents

Anonymous ACL submission

Abstract

Recent advances in Vision-Language Models (VLMs) have motivated the development of multi-modal search agents that can actively invoke external search tools and integrate retrieved evidence through multi-step reasoning. While promising, existing approaches typically rely on large-scale supervised trajectories or expensive reinforcement learning (RL), leading to high training cost, instability, and a severe cold-start problem for standard VLMs. We propose a training-free paradigm to empower VLMs with autonomous search capabilities via **cross-modal model merging**. By fusing a text-based search agent with a base VLM, we show that multi-modal search capabilities can be effectively composed without any additional multi-modal training data. To mitigate parameter interference during cross-modal integration, we introduce **Optimal Brain Merging (OBM)**, a saliency-aware merging algorithm that identifies task-critical parameters based on their impact on model loss using only a small set of calibration samples. Extensive experiments on search-intensive benchmarks (e.g., InfoSeek, MMSearch) reveal that: (1) Model merging secures a reasonable performance **floor** as a zero-shot agent, with OBM achieving superior search rates; (2) OBM significantly raises the performance **ceiling** as a warm-start strategy, achieving faster convergence and higher peak accuracy than standard VLM initialization.

1 Introduction

Recent advances in vision–language models (VLMs) have enabled systems capable of perceiving complex visual scenes and responding to diverse multi-modal queries. Beyond passive understanding, an emerging research direction focuses on building multi-modal search agents (Wu et al., 2025)—models that can autonomously decide when external information is required, invoke appropriate tools (e.g., image or text search), and integrate retrieved evidence through multi-step reasoning. Such agents play a central role in real-world applications including open-domain visual question answering, visual fact verification, etc. Despite their promise, constructing multi-modal search agents remains technically demanding. Existing approaches typically rely on large-scale supervised traces or reinforcement learning (RL) over carefully designed search environments. Collecting high-quality search-reasoning trajectories is costly, and RL training introduces substantial computational overhead and instability. More importantly, training often suffers from a pronounced cold-start problem: a vanilla VLM lacks intrinsic search behavior, forcing optimization to discover tool-use policies from scratch. These challenges significantly limit the accessibility and reproducibility of multi-modal agent research.

Interestingly, many of the capabilities required for effective multi-modal search agents are already present, but they are distributed across different types of models: vision–language models excel at visual perception and cross-modal grounding, while large language models demonstrate strong multi-step reasoning and tool-use abilities. The core challenge is therefore not acquiring these skills individually, but integrating them into a single system capable of both visual understanding and complex reasoning. A natural approach is parameter-level model fusion, which leverages pre-trained models of different modalities without requiring costly end-to-end multi-modal training. Existing merging methods, such as Task Arithmetic (TA) (Ilharco et al., 2023) or TIES-merging (TIES) (Yadav et al., 2023), provide a starting point for such fusion, but naively applying them may not fully capture the intricate dependencies between vision and language components. This motivates the need for fusion strategies that explicitly respect cross-modal alignment while combining complementary capabilities.

To address this limitation, we propose a cross-modal model merging framework that combines a vision–language model and a language-based search agent at the parameter level. Within this framework, we introduce Optimal Brain Merging (OBM), a saliency-aware parameter fusion algorithm tailored for cross-modal architectures. OBM estimates parameter importance based on their effect on model loss, using forward propagation on a small set of calibration samples to derive **activation-aware** saliency across the vision encoder, projector, and language backbone. By explicitly accounting for the structural dependencies between modalities, OBM enhances the robustness of the merged model and provides a principled way to adapt and extend existing merging techniques for effective cross-modal integration. Building upon OBM, we construct multi-modal search agents by integrating a text-based search agent and a vision–language model starting from a shared pre-trained backbone. This parameter-level fusion enables autonomous tool use without any additional multi-modal training. The resulting agents establish a strong zero-shot performance floor and serve as a reliable warm-start for downstream reinforcement learning, allowing faster convergence and higher peak performance while preserving cross-modal consistency.

Extensive experiments on search-intensive benchmarks—including FVQA-test (Wu et al., 2025), InfoSeek (Chen et al., 2023), MMSearch (Jiang et al., 2024), and LiveVQA (Fu et al., 2025)—demonstrate that model merging yields complementary benefits. On one hand, merged agents exhibit substantially stronger search behavior in zero-shot settings, establishing a higher performance floor than prompt-based approaches. On the other hand, when used as initialization for RL, OBM consistently accelerates convergence and leads to higher peak performance, effectively raising the performance ceiling of multi-modal search agents.

In summary, this work makes the following contributions:

- We present a new perspective on multi-modal search agent construction, reframing it as a problem of capability composition via model merging rather than end-to-end multi-modal training.
- We introduce OBM, a saliency-aware merging algorithm that leverages activation-dependent loss sensitivity to resolve parameter interference in cross-modal architectures.
- We show that merging-based agents not only establish a strong zero-shot performance floor, exhibiting autonomous search behavior without additional training, but also serve as an effective warm-start for downstream reinforcement learning, significantly improving training efficiency and peak performance compared to standard VLM initialization.

2 Related Work

2.1 Model Merging in LLMs and VLMs

Model merging combines specialized capabilities from multiple fine-tuned models without additional training. TA (Ilharco et al., 2023) composes "task vectors" by subtracting pre-trained from fine-tuned weights, while TIES (Yadav et al., 2023) and DARE (Yu et al., 2024) address parameter interference via trimming/sign consensus and random sparsification, respectively. Recent works (Jiang et al., 2025; Chen et al., 2025) apply TA to merge LLMs with VLMs for domain-specific reasoning, but evaluate only on single-turn tasks. To our knowledge, ours is the first to explore model merging for constructing multi-modal search agents, integrating visual perception with multi-step search-reasoning.

2.2 Multi-modal Search Agents

Building agents with visual perception and autonomous tool-use typically requires high-quality reasoning traces and extensive RL. MMSearch-R1 (Wu et al., 2025) establishes a specialized pipeline and benchmark for fine-tuning VLMs to perform search-based reasoning, while WebWatcher (Geng et al., 2025) pursues a **general-purpose vision-language deep research agent** by leveraging massive datasets and a diverse tool set (e.g., Code Interpreter, OCR). In contrast, we focus on the **compositional efficiency** of multi-modal search agents, proposing a **merging-based paradigm** that fuses existing task-specific experts. This yields a training-free solution with reasonable zero-shot performance and serves as an effective starting point for subsequent RL.

2.3 Foundations of Saliency-aware Optimization

OBM is grounded in classic pruning and quantization literature. OBD (LeCun et al., 1989) leverages

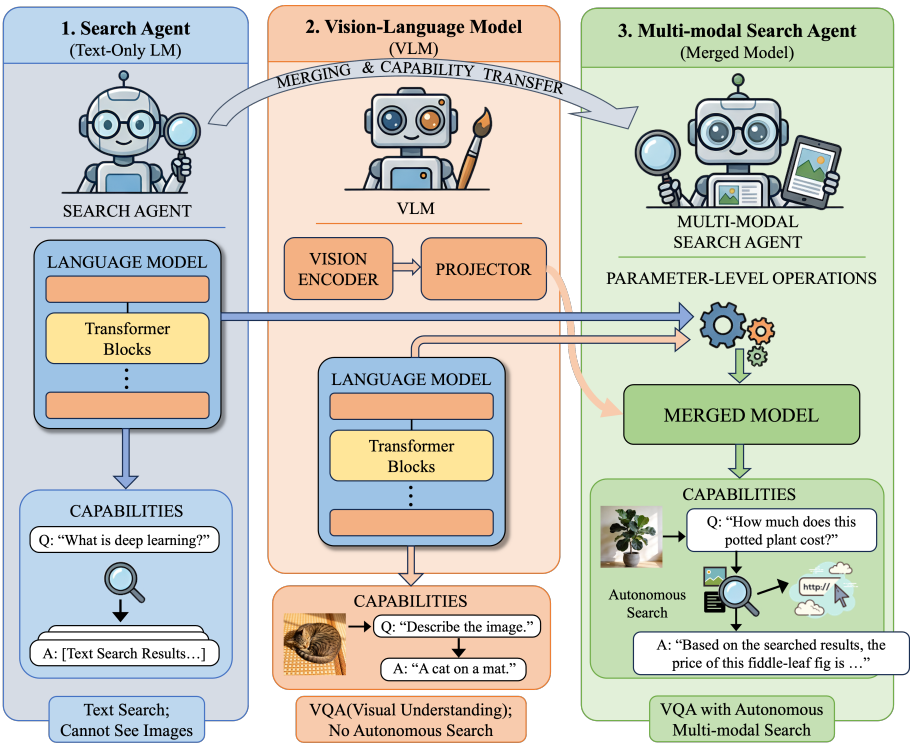

Figure 1: Overview of the merging paradigm for constructing a multi-modal search agent. The merged model retains the frozen vision encoder and projector from the VLM, while its language module is obtained via parameter-level merging between the VLM and the LLM.

second-order Taylor expansions of the loss to identify "salient" parameters, and GPTQ (Frantar et al., 2022) further refines this by computing the Hessian from layer inputs to minimize quantization reconstruction error. We adapt these insights to model merging, identifying task-critical parameters by their contribution to cross-modal reasoning rather than by numerical magnitude.

3 Methodology

In this section, we first introduce the overall framework for composing a multi-modal search agent via model merging, building upon standard formulations of task vectors and existing merging paradigms. We then present OBM, a saliency-aware algorithm designed to resolve parameter interference and facilitate effective cross-modal capability transfer.

3.1 Preliminaries: Model Merging via Task Vectors

Given a pre-trained base model θ_B and a fine-tuned checkpoint θ_T for a specific task T, the *task vector* is defined as $\delta_T = \theta_T - \theta_B$. The goal of model merging is to combine multiple task vectors $\{\delta_1, \delta_2, \ldots, \delta_n\}$ into a single unified model θ_{merged}.

A common baseline is **TA** (Ilharco et al., 2023), which performs a linear summation: $\theta_{merged} = \theta_B + \sum \lambda_i \delta_i$. To address parameter interference, methods like **TIES** (Yadav et al., 2023) introduce a two-stage process: (1) *Trimming*, which keeps only the top-$k\%$ values based on magnitude $|\delta_i|$; and (2) *Sign Consensus*, which resolves direction conflicts via a majority vote. However, these methods rely purely on the weight values in isolation, ignoring the underlying data distribution and inter-module dependencies.

3.2 Model Merging for Multi-modal Search Agent

Our goal is to create an agent that possesses both the visual perception of a VLM and the autonomous tool-use reasoning of a search agent. We leverage the fact that both Search-R1 (Jin et al., 2025) and the language backbone of `Qwen2.5-VL-7B-Instruct` (Bai et al., 2025) share the same ancestry, having been fine-tuned from the `Qwen2.5-7B` (Qwen et al., 2025) base model. Let $\theta_B \in \mathbb{R}^d$ denotes the parameters of the shared base model. We define the task vectors for the text-based search agent (δ_S) and the language module of the

VLM (δ_V) as follows:

$$\delta_S = \theta_{Search} - \theta_B, \quad \delta_V = \theta_{VL} - \theta_B, \tag{1}$$

where θ_{Search} and θ_{VL} represent the weights of Search-R1 [1] and the Qwen-VL [2] 's language module, respectively.

To construct the multi-modal agent, we transplant the vision encoder and the projector from the VLM directly into the fused system. The language module weights are initially combined using Task Arithmetic:

$$\theta_{merged} = \theta_B + \lambda_S \delta_S + \lambda_V \delta_V, \tag{2}$$

where λ are scaling factors. This architecture ensures the model can process visual inputs while inheriting the tool-use reasoning logic.

3.3 Optimal Brain Merging (OBM)

To overcome the **modality-agnostic** limitations discussed in the Introduction and more accurately identify parameters critical for cross-modal reasoning, we propose OBM. OBM introduces a saliency-aware framework consisting of two stages: saliency-driven sparsification and saliency-weighted sign consensus.

Stage 1: Saliency-driven Sparsification. The core principle of OBM is to prune task vector entries that have a negligible impact on the model's loss $\mathcal{L}$. Drawing inspiration from OBD(LeCun et al., 1989), we quantify the *saliency* s_i of the i-th component of a task vector δ_i by the second-order Taylor expansion of the loss change:

$$s_i = \frac{1}{2} \frac{\partial^2 \mathcal{L}}{\partial \theta_i^2} \delta_i^2 = \frac{1}{2} h_{ii} \delta_i^2, \tag{3}$$

where h_{ii} represents the i-th diagonal element of the Hessian matrix. To avoid the prohibitive computational cost of calculating the full Hessian for large-scale models, we adopt a layer-wise Mean Squared Error (MSE) approximation, as motivated by GPTQ(Frantar et al., 2022). For the l-th linear layer with weight matrix $\mathbf{W}^l$ and input activations $\mathbf{X}^l$, the local objective is defined as

$$\Delta \mathcal{L}^l = \|\Delta \mathbf{W}^l \mathbf{X}^l\|_2^2. \tag{4}$$

Under this formulation, the Hessian matrix simplifies to $\mathbf{H}^l = 2\mathbf{X}^l \mathbf{X}^{l\top}$. This enables efficient saliency computation via a single forward pass using a minimal set of calibration samples (multi-modal VQA data for δ_V and search-reasoning data for δ_S). Crucially, since $\mathbf{X}^l$ is dynamically collected during the forward pass, the saliency of δ_V inherently captures the activation-aware dependencies between the language backbone and the frozen vision-projection modules. This effectively bridges the gap between disparate modalities, whereas the saliency of δ_S isolates the logic governing search-based reasoning. For each task vector, we retain only the top-$p\%$ parameters with the highest saliency to eliminate interference while preserving modality-specific expertise:

$$\hat{\delta}_i = \begin{cases} \delta_i & \text{if } s_i \in \text{Top-}p\% \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

For non-linear components such as embedding layers, we assign uniform saliency to all parameters, as task-specific knowledge is primarily concentrated within the Transformer blocks. This allows these layers to follow the same subsequent merging pipeline while maintaining consistent model density.

Stage 2: Saliency-weighted Sign Consensus. In cases where task vectors exhibit conflicting directions (i.e., $\text{sign}(\delta_{S,i}) \neq \text{sign}(\delta_{V,i})$), traditional methods (Yadav et al., 2023; Deep et al., 2024) rely on simple magnitude comparison. We instead propose a **weighted consensus mechanism** that accounts for the relative importance of each task. The consensus direction is determined by:

$$\sigma_i = \text{sign}\left(\sum_{k \in S,V} s_i^{(k)} \cdot \text{sign}(\hat{\delta} i^{(k)}) \right). \tag{6}$$

The final merged task vector is construc ted by aligning the parameters with this consensus sign. Specifically, we preserve the values that agree with σ_i, ensuring that parameters vital to "multi-modal synergy" are retained with their correct task-specific orientations.

4 Experiments

We conducted experiments in two phases, before and after training, to explore the superiority of the model merging paradigm in constructing multi-modal search agent.

4.1 Experimental Settings

Baselines and Answering Paradigms All baselines share a consistent search toolset compris-

[1] Search-R1: `SearchR1-nq_hotpotqa_train-qwen2.5-7b-em-ppo-v0.3`.

[2] Qwen-VL: `Qwen2.5-VL-7B-Instruct`.

ing image-to-image and text-to-text search engines (Appendix B). Prompts are reused from (Wu et al., 2025) with minor tweaks (Appendix A).

- **Direct Answering (DA)**: Qwen-VL answers directly from the image and question, without search.
- **Search-augmented (Qwen-VL-R1)**: A zero-shot prompting baseline where Qwen-VL is instructed to call search tools only when visual information is insufficient.
- **RAG-based Workflow**: A fixed pipeline from (Wu et al., 2025) performing image search, text-query generation, text search, and final answer synthesis.
- **MMSearch-R1**: An RL-trained baseline fine-tuned from Qwen-VL on multi-modal data with both search-required and search-free examples. Since the search tools in the official MMSearch-R1 codebase are not publicly available, we reproduce this baseline using our own search tool implementations (Appendix B).
- **Merging-based Agents (TA, TIES, DARE, OBM)**: Models constructed by merging Search-R1 with Qwen-VL. For OBM, parametric saliency is computed on 128 random samples each from NQ-HotpotQA (Kwiatkowski et al., 2019; Yang et al., 2018) (for Search-R1) and MMStar (Chen et al., 2024) (for Qwen-VL's language module). See Appendix C.1 for hyper-parameters.

Benchmarks We evaluate on four knowledge-intensive and information-seeking VQA datasets. SimpleVQA (Cheng et al., 2025) used in (Wu et al., 2025) is excluded, as our preliminary experiments (Appendix D) show its extremely low dependency on external search.

- **FVQA-test** (Wu et al., 2025): A curated set of examples from FVQA-auto-vc, re-annotated InfoSeek, and additional human-collected samples, covering diverse visual and textual knowledge.
- **InfoSeek** (Chen et al., 2023): A subset of the published MMSearch-R1 test split, focusing on information-seeking VQA grounded in Wikidata triples.
- **MMSearch** (Jiang et al., 2024): Examples from subdomains such as News (2024 events) and rare Knowledge, all requiring multi-modal search.
- **LiveVQA** (Fu et al., 2025): Multi-hop, timely questions collected from global news platforms (e.g., CNN, BBC) requiring reasoning over both visual and textual content.

Metrics

- **Accuracy (Acc):** Exact Match (EM) between prediction and ground-truth, following standard VQA normalization (lowercasing, punctuation removal, article stripping, whitespace trimming).
- **Search Rate (SR):** Percentage of cases where the agent invokes any external search tool (image or text) at least once.
- **Image / Text Search Rates (SR_{img}, SR_{txt}):** Separate invocation rates for image and text search tools. The two are not mutually exclusive—a multi-hop episode (e.g., image search followed by text search) can yield $SR_{img} + SR_{txt} > 100\%$.

4.2 Zero-shot Performance: Securing a High Floor

We first evaluate the performance of the training-free merging paradigm in Table 1. The results provide several key insights into how model merging establishes a superior "floor" for multi-modal search agents.

Limitations of Prompt-based Autonomy Simply relying on sophisticated prompting (**Qwen-VL-R1**) to elicit autonomous tool-use yields sub-optimal results. While Qwen-VL-R1 shows marginal improvements over Direct Answering (**Qwen-VL-DA**) on information-seeking benchmarks—specifically on InfoSeek (19.45% vs. 17.90%) and MMSearch (10.53% vs. 8.77%)—its overall performance remains constrained. This suggests that the inherent reasoning logic for visual search is not fully activated by prompts alone when the base model lacks dedicated search-reasoning parameters.

Merging Transfers Search "Instincts" In contrast, merging-based agents (**TA** and **OBM**) significantly outperform Qwen-VL-R1 across all benchmarks. Notably, on FVQA-test and MMSearch,

Table 1: Benchmark results of our merged models vs. baselines. We compare our merged models—derived via TA, TIES, DARE, and OBM—against several baselines, including the trained MMSearch-R1 and three Qwen-VL variants: Direct Answering (DA), Search-augmented (R1), and RAG-based workflows.

Model	FVQA-test			InfoSeek			MMSearch			LiveVQA		
	Acc.	SR_img	SR_txt	Acc.	SR_img	SR_txt	Acc.	SR_img	SR_txt	Acc.	SR_img	SR_txt
MMSearch-R1	29.39	66.06	47.28	34.60	56.20	36.80	24.56	86.55	67.25	15.60	77.67	53.16
Qwen-VL-DA	15.94	0.00	0.00	17.90	0.00	0.00	8.77	0.00	0.00	12.41	0.00	0.00
Qwen-VL-R1	15.61	8.44	25.50	19.45	11.85	16.50	10.53	16.38	45.03	7.69	12.54	22.43
Qwen-VL-RAG	24.83	100.00	100.00	21.30	100.00	100.00	21.64	100.00	100.00	13.69	100.00	100.00
TA	21.50	19.61	43.11	22.85	35.55	38.80	13.45	15.21	64.92	10.91	38.01	43.20
TIES	15.00	1.94	32.72	15.10	2.45	37.00	9.94	4.09	49.12	11.88	6.47	37.70
DARE	10.06	8.39	29.94	10.65	12.50	35.75	4.09	5.26	32.16	5.47	13.63	17.71
OBM	20.67	16.50	55.55	19.65	31.70	46.15	13.45	21.63	78.94	11.38	19.10	60.05

they achieve an accuracy gain of approximately 5% over Qwen-VL-DA. By examining the search behaviors, we observe that these models exhibit remarkably high search rates. For instance, **OBM achieves a text search rate (SR_{txt}) of 78.94% on MMSearch**. This high activation demonstrates that the integration of Search-R1 parameters successfully transposes search logic into the VLM, enabling the model to autonomously seek external evidence to resolve visual queries.

Narrowing the Gap to RAG Although TA and OBM do not yet consistently surpass the Qwen-VL-RAG baseline in a zero-shot setting, they significantly narrow the performance gap compared to the Qwen-VL-R1. It is important to note that the RAG baseline follows a **rigid, predefined workflow** (sequential image and text search) which, while robust, is not necessarily optimal for every query. In contrast, our merging-based agents attempt to replicate the **autonomous reasoning** seen in trained models like MMSearch-R1. The fact that merged models can relatively approach RAG-level accuracy without a hard-coded pipeline demonstrates their potential. This suggests that model merging provides a much sturdier "floor" for building flexible agents that can eventually surpass fixed workflows through more efficient, targeted tool-use.

Modality Preference and Search Efficiency An analysis of the SR_{img} and SR_{txt} distribution reveals a distinctive "text-centric" bias in zero-shot merged models. While the RL-trained **MMSearch-R1** demonstrates high efficiency by achieving superior accuracy with a more balanced search frequency, zero-shot merged models rely more heavily on textual reasoning. This indicates that while merging successfully implants the "will" to search, the **fine-grained coordination** between visual evidence and specific tool selection—such as knowing exactly when to trigger an image search versus a text search—is a higher-order skill that requires further optimization.

Vulnerability of Unimodal Merging in Cross-modal Contexts Standard algorithms for unimodal model merging, such as **TIES** (Yadav et al., 2023) and **DARE** (Yu et al., 2024), struggle in this cross-modal scenario. Notably, **DARE** suffers from a catastrophic performance drop compared to the base Qwen-VL-DA. We attribute this to DARE's mechanism of randomly dropping task vector entries. In a VLM, the language module parameters are delicately aligned with the vision encoder and projector; DARE's stochastic pruning likely disrupts these cross-modal synergistic dependencies, effectively severing the pathway for visual information. Similarly, magnitude-based methods like **TIES** may inadvertently eliminate small-magnitude parameters that are nonetheless critical for bridging visual perception and search reasoning. These failures reinforce the necessity of **OBM**, which utilizes saliency-aware Hessian information to preserve task-critical parameters while maintaining modality alignment.

Based on these zero-shot observations, we select **TA**, **TIES**, and **OBM** as the base models for the subsequent training phase. While TIES performs slightly worse than TA, it maintains a reasonable search rate and serves as a representative baseline for interference-reduction methods. **OBM**, despite having slightly lower accuracy than TA on some benchmarks, exhibits the highest overall search rate. We hypothesize that this stronger "search-reasoning instinct" indicates a higher performance ceiling when the model is further optimized through reinforcement learning, which we explore in the next section.

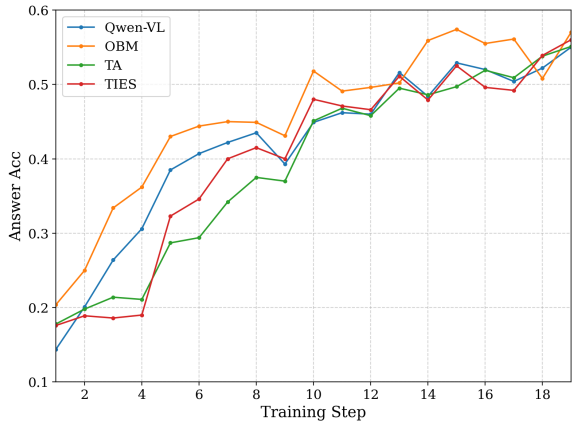

Figure 2: Comparison of training accuracy in the early training phase across different base models.

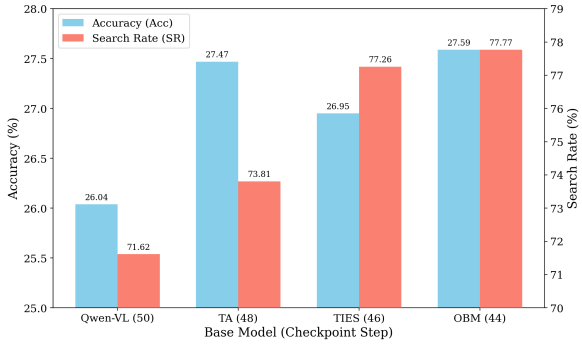

Figure 3: Average accuracy and search rate on benchmarks at each model's best checkpoint.

4.3 Training Dynamics: Raising the Performance Ceiling

To evaluate the impact of model merging on the training process, we performed reinforcement learning using the same training data (5,000 samples) and parameter update frequency as in MMSearch-R1 (Wu et al., 2025). All other training hyper-parameters were also kept identical to those used in MMSearch-R1.

Accelerated Convergence The initial phase of training reveals a significant efficiency gap between standard VLM initialization and our OBM based paradigm. As shown in Figure 2, which tracks the training accuracy over the first two epochs (18 steps), OBM continuously and substantially outperforms all other initialization strategies. Notably, OBM requires only 5 steps to surpass the 1-epoch performance of other models and reaches their 2-epoch accuracy level within just 14 steps. This rapid ascent indicates that the saliency-aware merging in OBM preserves a pre-aligned reasoning-perception space, allowing the model to bypass the "cold-start" phase typical of training multi-modal agents from scratch.

Breakthrough in Peak Performance A central claim of our work is that merging "raises the ceiling" of the agent's ultimate capability. Figure 3 summarizes the average Acc and SR at each model's optimal checkpoint. The results clearly demonstrate that starting from vanilla Qwen-VL leads to a lower performance plateau compared to starting from a merged model. OBM achieves the highest peak performance (27.59% Avg. Acc and 77.77% Avg. SR) at step 44, which is not only better but also earlier than the peak of other variants. This suggests that the "search instincts" embedded during the merging phase provide a more robust foundation for the model to explore and master complex tool-use logic.

Generalization vs. In-domain Fitting We further analyze the late-stage training performance (steps 44–52) in Table 2. Several key observations emerge regarding the stability and generalization of the merged agents:

- **Superiority of Merged Initializations**: Across the entire late-stage window, the pure Qwen-VL baseline (**MMSearch-R1**) consistently lags behind the merged models in both in-domain (FVQA-test, InfoSeek) and out-of-distribution (MMSearch, LiveVQA) benchmarks. This confirms that the benefits of model merging persist throughout the training lifecycle.
- **The Overfitting Risk of TIES**: While **TIES** shows impressive fitting capabilities on in-domain data (leading on FVQA-test and InfoSeek), it suffers from poor generalization. Its performance on the search-intensive MMSearch benchmark is notably weaker than TA and OBM, suggesting that magnitude-based merging may lead to a more rigid parameter distribution that overfits to specific reasoning patterns.
- **Robust Generalization of OBM**: Both **TA** and **OBM** demonstrate superior generalization on OOD benchmarks like MMSearch and LiveVQA. OBM, in particular, maintains a high "performance-to-step" ratio, hitting the global SOTA at step 44 and continuing to exhibit strong results at step 52.

Table 2: Late-stage performance comparison of various base models. We report the benchmark results during the final training phase (steps 44–52).

Step	Model	FVQA-test	InfoSeek	MMSearch	LiveVQA	Avg.
44	Qwen-VL	27.72	33.45	23.98	15.24	25.10
	TA	29.06	34.25	26.90	15.55	26.44
	TIES	**30.28**	**34.85**	25.73	15.71	26.64
	OBM	28.94	34.45	**30.99**	**15.96**	**27.59**
46	Qwen-VL	28.39	33.80	22.22	15.21	24.91
	TA	29.94	34.45	**27.48**	15.46	26.83
	TIES	**30.22**	**35.60**	26.32	15.66	**26.95**
	OBM	28.56	34.60	26.32	**16.05**	26.38
48	Qwen-VL	28.44	34.25	23.98	15.24	25.48
	TA	28.61	34.75	**30.41**	**16.13**	**27.47**
	TIES	**30.17**	**35.75**	23.98	15.94	26.46
	OBM	29.00	34.60	25.15	15.99	26.19
50	Qwen-VL	29.39	34.60	24.56	15.60	26.04
	TA	29.00	34.15	**28.66**	15.63	**26.86**
	TIES	**30.17**	34.80	22.22	**15.91**	25.78
	OBM	29.44	**34.95**	26.32	15.74	26.61
52	Qwen-VL	29.05	34.40	24.56	15.32	25.83
	TA	29.06	34.85	28.07	15.91	26.97
	TIES	**30.06**	34.90	20.47	15.77	25.30
	OBM	28.89	**34.95**	**29.83**	**15.96**	**27.41**

Table 3: Ablation study on calibration data for OBM. We compare the default OBM with OBM (Text-only) (text-based calibration) and Oracle (benchmark-specific calibration).

Variant	FVQA-test & InfoSeek			MMSearch & LiveVQA		
	Acc	SR_img	SR_txt	Acc	SR_img	SR_txt
OBM	**20.16**	24.10	50.85	**12.42**	20.37	**69.50**
OBM (Text-only)	19.48	24.21	**53.29**	12.20	23.96	65.53
Oracle	19.84	**30.40**	52.85	11.38	**27.98**	66.74

In summary, the training-free OBM paradigm serves as a powerful warm-start strategy. By precisely integrating task-specific expertise while mitigating interference, OBM enables the agent to converge faster and ultimately reach a performance ceiling that is unattainable through standard multi-modal training alone.

4.4 Ablation Study

To analyze the impact of calibration data used for saliency computation of Qwen-VL in OBM, we compare the default OBM against two variants: **OBM (Text-only)**—calibrated solely on textual QA data—and **Oracle**—calibrated on samples drawn from the evaluation benchmarks. All variants use 128 calibration samples.

As shown in Table 3, the default OBM consistently outperforms OBM (Text-only). Without visual forward passes during calibration, the Hessian fails to capture activation-aware dependencies between the language backbone and the visual projector, confirming that multi-modal calibration is essential to preserve cross-modal alignment during parameter integration.

Interestingly, despite being distribution-aligned, the Oracle variant yields lower accuracy than OBM.

Table 4: Computational cost of different merging methods. For OBM, saliency computation is performed once using forward propagation and can be cached for reuse.

Method	#Models	GPU	Saliency Computation	Merging Time
TA	2	1×H20 (96G)	–	0'22
TIES	2	1×H20 (96G)	–	7'03
OBM	2	1×H20 (96G)	2'31	6'46

While it significantly boosts SR_{img} , this leads to an "excessive but inefficient" image search behavior at the zero-shot stage: the agent triggers tools more frequently but cannot effectively leverage the surge of image retrieval. Overfitting to task-specific signals disrupts the delicate perception–reasoning balance.

Nonetheless, performance differences among the three variants are minor, underscoring OBM's robustness—it thrives on task-agnostic, modality-aware calibration without requiring benchmark-specific cherry-picking.

4.5 Merging Cost

Table 4 reports the computational cost of different merging methods. All approaches complete model merging in just seconds to a few minutes on a single GPU. OBM introduces a modest additional cost of approximately 2 minutes for saliency estimation, performed once via forward propagation on a small calibration set. This step requires no backpropagation, gradient updates, or iterative optimization, and the computed saliency can be cached and reused for all subsequent merges.

5 Conclusion

In this work, we revisit multi-modal search agents and propose a training-free, cross-modal model fusion paradigm. By integrating pre-trained vision–language models with language-based search agents and using OBM to preserve cross-modal dependencies, our approach enables autonomous tool use without additional multi-modal training. Experiments show that the resulting agents achieve strong zero-shot performance and provide an effective warm-start for reinforcement learning, demonstrating that parameter-level fusion is a practical alternative to end-to-end multi-modal training.

Limitations

Our merging-based paradigm relies on the assumption that the source models share a common pre-trained backbone, which is a natural fit for cur-

rent model families (e.g., Qwen series) but may limit applicability when the candidate experts come from heterogeneous architectures. In addition, while OBM accelerates RL convergence and raises the performance ceiling, the absolute accuracy on challenging benchmarks such as LiveVQA is still modest, suggesting room for further improvement through better search tools, training data, or merging strategies.

References

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and 1 others. 2024. Are we on the right way for evaluating large vision-language models? *Advances in Neural Information Processing Systems*, 37:27056–27087.

Shiqi Chen, Jinghan Zhang, Tongyao Zhu, Wei Liu, Siyang Gao, Miao Xiong, Manling Li, and Junxian He. 2025. Bring reason to vision: Understanding perception and reasoning through model merging. *arXiv preprint arXiv:2505.05464*.

Yang Chen, Hexiang Hu, Yi Luan, Haitian Sun, Soravit Changpinyo, Alan Ritter, and Ming-Wei Chang. 2023. Can pre-trained vision and language models answer visual information-seeking questions? *arXiv preprint arXiv:2302.11713*.

Xianfu Cheng, Wei Zhang, Shiwei Zhang, Jian Yang, Xiangyuan Guan, Xianjie Wu, Xiang Li, Ge Zhang, Jiaheng Liu, Yuying Mai, and 1 others. 2025. Simplevqa: Multimodal factuality evaluation for multimodal large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4637–4646.

Pala Tej Deep, Rishabh Bhardwaj, and Soujanya Poria. 2024. Della-merging: Reducing interference in model merging through magnitude-based sampling. *arXiv preprint arXiv:2406.11617*.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. 2022. GPTQ: Accurate post-training compression for generative pretrained transformers. *arXiv preprint arXiv:2210.17323*.

Mingyang Fu, Yuyang Peng, Dongping Chen, Zetong Zhou, Benlin Liu, Yao Wan, Zhou Zhao, Philip S Yu, and Ranjay Krishna. 2025. Seeking and updating with live visual knowledge. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Xinyu Geng, Peng Xia, Zhen Zhang, Xinyu Wang, Qiuchen Wang, Ruixue Ding, Chenxi Wang, Jialong Wu, Yida Zhao, Kuan Li, and 1 others. 2025. Webwatcher: Breaking new frontier of vision-language deep research agent. *arXiv preprint arXiv:2508.05748*.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*.

Dongzhi Jiang, Renrui Zhang, Ziyu Guo, Yanmin Wu, Jiayi Lei, Pengshuo Qiu, Pan Lu, Zehui Chen, Chaoyou Fu, Guanglu Song, and 1 others. 2024. Mmsearch: Benchmarking the potential of large models as multi-modal search engines. *arXiv preprint arXiv:2409.12959*.

Lingjie Jiang, Shaohan Huang, Xun Wu, Yixia Li, Dongdong Zhang, and Furu Wei. 2025. Viscodex: Unified multimodal code generation via merging vision and coding models. *arXiv preprint arXiv:2508.09945*.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan O Arik, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-r1: Training LLMs to reason and leverage search engines with reinforcement learning. In *Second Conference on Language Modeling*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, and 1 others. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Yann LeCun, John Denker, and Sara Solla. 1989. Optimal brain damage. *Advances in neural information processing systems*, 2.

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, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Jinming Wu, Zihao Deng, Wei Li, Yiding Liu, Bo You, Bo Li, Zejun Ma, and Ziwei Liu. 2025. Mmsearch-r1: Incentivizing lmms to search. *arXiv preprint arXiv:2506.20670*.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. 2023. Ties-merging: Resolving interference when merging models. *Advances in Neural Information Processing Systems*, 36:7093–7115.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering.

In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2369–2380.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*.

A Prompts

The prompts used to reproduce MMSearch-R1 (Wu et al., 2025) are exactly the same as those they released open-source:

First Round User Prompt

Answer the user's question based on the provided image. Examine the image carefully and identify any recognizable entities, such as faces, objects, locations, events, logos, or text. Determine whether you have sufficient knowledge to confidently recognize the main visual element and answer the user's question. If so, first explain your reasoning, then provide a clear and direct answer.
If you are unable to confidently identify the visual element, stop and invoke the image search tool by appending the string `<search><img></search>` at the end of your response. This will trigger a Google Lens search using the original image to retrieve relevant information that can help you confirm the visual content. Once you have sufficient visual understanding, combine it with the user's question and assess whether you can confidently answer. If so, answer the question directly using your own knowledge. If not, invoke the text search tool by generating a concise and specific query, and output it in the format `<text_search>your query here</text_search>` at the end of your response. Carefully craft your query to accurately retrieve the information needed to help answer the question. The text search tool will then use Google Search to return relevant information based on your query.
You must include your reasoning inside `<reason>...</reason>` before taking any action, whether it is calling the image search tool, generating a text search query, or providing a final answer. The reasoning may involve analysis of the original image and question, interpretation of search results, or logical steps leading to the final answer.
All search results will be placed inside `<information>` and `</information>` and returned to you. When you are ready to answer the question, wrap your final answer between `<answer>` and `</answer>`, without detailed illustrations. For example: `<answer>Titanic</answer>`.
Here is the image and the question: `<image>`

After Image Search Prompt

Please extract the visual element relevant to the user's question (such as faces, objects, locations, events, logos, or text) from the search results. Then, combine this information with the user's question and perform reasoning to determine whether a Google text search is needed to retrieve additional information to answer the question.
If a text search is needed, output the string `<text_search>your query here</text_search>` at the end of your response. Please generate a well-crafted query based on the visual element that will help retrieve the most relevant information.
If a text search is not needed, use your own knowledge to directly answer the user's question. You must conduct your reasoning inside `<reason>` and `</reason>` before taking any action, whether it's generating a text search query or providing a final answer. If you decide to give the final answer, place it inside `<answer>` and `</answer>`, without detailed explanation or illustration. For example: `<answer>Titanic</answer>`

After Text Search Prompt

Please analyze the search results and the user's question and continue reasoning inside `<reason>` and `</reason>`.
If you determine that additional knowledge is still required to answer the user's question, stop responding to the question and instead report a warning by outputting the string 'Unable to answer due to lack of relevant information' at the end of your response.
If no further external information is needed, you should provide the final answer by placing it within `<answer>` and `</answer>`. The answer must be concise, clear, and to the point, without any additional explanation or elaboration.

For our merged models, since their lineage traces back to Search-R1 (Jin et al., 2025), we adopt Search-R1's prompt convention by replacing `<reason>` with `<think>` and `<text_search>` with `<search>`.

The prompt used by Qwen-VL-DA and Qwen-VL-RAG are exactly the same as "Prompt used for Direct Answer" and "Prompts used for RAG workflow" in MMSearch-R1 (Wu et al., 2025), respectively.

B Search Tools

Following MMSearch-R1 (Wu et al., 2025), we implement two search tools: a text search tool and an image search tool. When the model invokes the image search tool, it retrieves the top-k visually relevant web pages and returns a sequence of interleaved thumbnails and corresponding titles extracted from those pages. When the model invokes the text search tool with a textual query, it retrieves the top-k relevant web pages and returns a summary of each page's content along with its title. For further details, please refer to MMSearch-R1 (Wu et al., 2025).

C Hyper-parameters

C.1 Model Merging

When applying the TA method (Ilharco et al., 2023) for model merging, we searched for the task vector coefficient λ within the set $\{0.3, 0.5, 0.7\}$, ulti-

mately selecting $\lambda = 0.7$ for Qwen-VL and $\lambda = 0.3$ for Search-R1.

For methods that inherently incorporate sparsification—TIES (Yadav et al., 2023), DARE (Yu et al., 2024), and OBM—we fixed $\lambda = 1$ and instead tuned the parameter density p, which controls the fraction of top-p most important parameters to retain. We evaluated p over the values $\{0.3, 0.5, 0.7, 0.9\}$, and selected $p = 0.7$ for both TIES and OBM, and $p = 0.9$ for DARE.

C.2 Training

Our training hyper-parameters are identical to those used in MMSearch-R1 (Wu et al., 2025). Please refer to that work for implementation details.

D SimpleVQA

As illustrated in Table 5, the specially trained model actually achieved lower accuracy than Qwen-VL, which directly generates answers without search. Moreover, the RAG pipeline with a 100% search rate further degraded performance, suggesting that SimpleVQA may not be an information-seeking dataset and is therefore unsuitable for evaluating the capabilities of multimodal search agents in this work.

Table 5: Performance on SimpleVQA.

Model	Acc	SR
Qwen-VL-DA	26.75	0.00
Qwen-VL-RAG	19.15	100.00
MMSearch-R1	24.58	45.80