OSCAR: Optimization-Steered Agentic Planning for Composed Image Retrieval

Anonymous ACL submission

Abstract

Composed image retrieval (CIR) requires complex reasoning over heterogeneous visual and textual constraints. Existing approaches largely fall into two paradigms: *unified embedding retrieval*, which suffers from *single-model myopia*, and *heuristic agentic retrieval*, which is limited by *suboptimal, trial-and-error orchestration*. To this end, we propose **OSCAR**, an optimization-steered agentic planning framework for composed image retrieval. We are the first to reformulate agentic CIR from a heuristic search process into a principled trajectory optimization problem. Instead of relying on heuristic trial-and-error exploration, OSCAR employs a novel offline-online paradigm. In the offline phase, we model CIR via atomic retrieval selection and composition as a two-stage mixed-integer programming problem, mathematically deriving optimal trajectories that maximize ground-truth coverage for training samples via rigorous boolean set operations. These trajectories are then stored in a golden library to serve as in-context demonstrations for online steering of VLM planner at online inference time. Extensive experiments on three public benchmarks and a private industrial benchmark show that OSCAR consistently outperforms SOTA baselines. Notably, it achieves superior performance using only **10%** of training data, demonstrating strong generalization of planning logic rather than dataset-specific memorization. Our code is available[1].

1 Introduction

With the evolution of modern retrieval systems, real-user queries have become increasingly complex and multimodal, which often involve compositional descriptions, multiple constraints, and implicit reasoning over both visual and textual information (Song et al., 2025; Datta et al., 2008). In response, a growing body of research has emerged

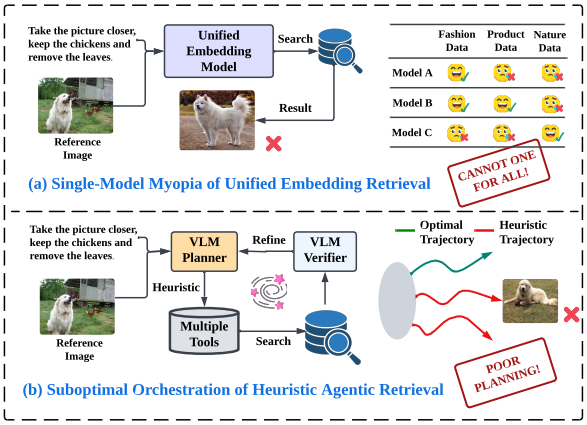

Figure 1: The illustration of limitations of existing image retrieval methods, *i.e.*, (a) single-model myopia of unified embedding retrieval, and (b) suboptimal orchestration of heuristic agentic retrieval.

to address composed image retrieval, generally falling into two paradigms: *Unified Embedding Retrieval* and *Heuristic Agentic Retrieval*. While both have yielded promising advances, they exhibit fundamental limitations that increasingly bottleneck real-world performance, as shown in Figure 1.

Firstly, **unified embedding retrieval** attempts to resolve complex queries via a single, monolithic representation model, whether using a general-purpose multimodal encoder (Jiang et al., 2024a; Xue et al., 2025) or a domain-specific model fine-tuned for a particular setting (Yang et al., 2024b; Radford et al., 2021). However, they suffer from **single-model myopia**. Concretely, this paradigm implicitly assumes that a *single latent space* can act as a universal solution, whereas real-world queries are inherently heterogeneous. User intents vary drastically in granularity, ranging from high-level stylistic shifts (*e.g.*, more formal) to fine-grained attribute constraints (*e.g.*, specific color, texture, or pattern), and the optimal visual evidence shifts across domains (Baldrati et al., 2023a; Liu et al., 2021). Under such diversity, relying on a single embedding space inevitably leads to short-sighted behaviors. As illustrated in Figure 1(a), a model

[1] https://anonymous.4open.science/r/OSCAR-3A55/README.md

optimized for natural images may fail to resolve fashion-specific attributes, while a fashion-oriented model may struggle with natural scene reasoning. Consequently, even state-of-the-art single-model systems remain brittle in practice, as they are unable to adapt to intents that evolve or drift across diverse domains.

Secondly, **heuristic agentic retrieval** seeks to overcome the representation bottleneck by decomposing the task into multi-step workflows invoked by Large Language Models (LLMs) (Achiam et al., 2023; Yang et al., 2025) or Vision–Language Models (VLMs) (Bai et al., 2025a). These methods leverage external tools (*e.g.*, captioners, rewriters, and retrievers) to handle complex queries (Cheng et al., 2025; Yang et al., 2026). However, despite their flexibility, these approaches struggle with **sub-optimal orchestration**, as shown in Figure 1(b). Current pipelines rely heavily on the agent's internal heuristics (*e.g.*, ReAct-style loops (Yao et al., 2023)), where the model makes greedy, iterative decisions on which tool to call next based on intermediate outputs. This lack of a global objective leads to unstructured and inefficient trajectories characterized by redundant calls, poor logical ordering, and unreliable handling of set-theoretic constraints (inclusion/exclusion). Furthermore, the reliance on iterative interaction incurs significant computational overhead without offering any guarantees towards the optimal retrieval path.

To this end, we propose **OSCAR**, an optimization-steered agentic planning framework for composed image retrieval, which, for the first time, reframes agentic CIR from a heuristic search process into a principled trajectory planning problem. Unlike prior methods that rely on heuristic, trial-and-error LLM exploration, OSCAR introduces a novel *offline-online* paradigm. During the offline stage, we treat each individual retriever invocation as the fundamental unit, and derive optimal tool-call planning trajectories for training samples by solving a two-stage mixed-integer programming (MIP) formulation. These optimal trajectories are then stored in a *golden library* to serve as reasoning demonstrations during inference. At test time, OSCAR acquires relevant planning patterns to *steer* the VLM agent, enabling it to replicate optimal logical reasoning for CIR without requiring expensive iterative search or human annotation.

In summary, our contributions are as follows:

- **Optimization Perspective.** To the best of our knowledge, we are the first to formulate agentic CIR as a MIP problem. By shifting from heuristic search to global optimization, we mathematically derive *optimal* planning trajectories for training samples that maximize ground-truth coverage while minimizing computational redundancy, providing strong demonstration signals without human annotation.

- **Set-Theoretic Composition Logic**. We introduce a rigorous logic for composing CIR results via boolean set operations (*i.e.*, union, intersection, and difference). This enables the agent to perform explicit inclusion and conservative exclusion reasoning, a capability that is mathematically intractable for single-embedding models and heuristic agentic methods.

- **The OSCAR Framework**. We propose a novel offline-online paradigm that bridges MIP optimization and agentic planning. By storing MIP solutions into a golden library as demonstrations, OSCAR effectively steers VLMs to perform complex compositional planning for CIR in a single inference pass.

- **Empirical Superiority**. On three public benchmarks and a private industrial benchmark, OSCAR consistently outperforms SOTA single-embedding and agentic baselines. Notably, it achieves these gains with only **10%** of the training data for library construction, demonstrating strong generalization of abstract planning logic rather than simple memorization.

2 Related Works

Composed image retrieval (CIR) aims to retrieve target images that satisfy specific textual modifications while preserving the visual context of a reference image. This task demands fine-grained compositional reasoning over multimodal heterogeneous attributes with increasing semantic complexity (Song et al., 2025; Baldrati et al., 2023a).

Unified Embedding Retrieval for CIR. A dominant line of work formulates CIR by collapsing a composed query, which consists of a reference image and a modification text, into a single embedding, followed by nearest-neighbor retrieval. Composition is typically achieved via cross-modal fusion, feature editing, or language-aligned representations in a shared latent space (Baldrati et al.,

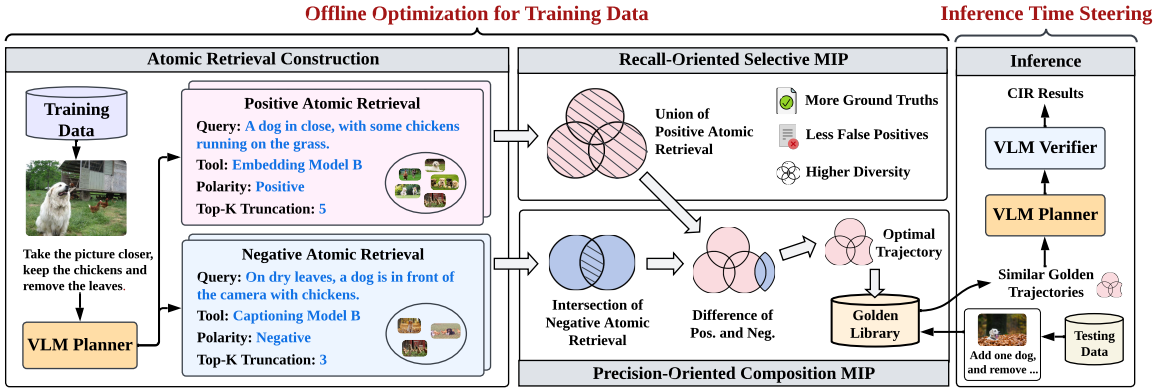

Figure 2: The overall framework of our proposed OSCAR.

2023b; Saito et al., 2023). Moreover, recent large-scale benchmarks (Jiang et al., 2024a) have driven rapid progress in multimodal embedding models for CIR, with a series of methods achieving increasingly strong performance (Xue et al., 2025). However, different models exhibit varying strengths across different domains and datasets, rendering the single-model myopia problem.

Heuristic Agentic Retrieval for CIR. Recent agentic methods decompose CIR into multi-step workflows, such as query rewriting, retrieval, and correction (Cheng et al., 2025). However, they are typically driven by heuristic exploration, often requiring repeated model interactions, incurring high computational cost, and lacking global optimality guarantees. In contrast, our proposed OSCAR replaces this heuristic trial-and-error with optimization-steered planning, using an offline MIP formulation to guide the agent towards provably efficient and accurate retrieval trajectories.

More detailed related work is provided in Appendix A.

3 Methodology

In this section, we present **OSCAR**, an Optimization-Steered Agentic Planning Framework for Composed Image Retrieval. The overall framework of OSCAR is shown in Figure 2. All the prompts for VLM invocation are provided in Appendix M.

3.1 Preliminaries

Problem Formulation. Let $q = \{q_{img}, q_{txt}\}$ denote a composed query, where q_{img} is the reference image and q_{txt} is the modification text. The goal of CIR is to retrieve a set of ground-truth images $I^+ \subset \mathcal{I}$ from an image gallery $\mathcal{I}$ based on the composed query q.

Standard approaches typically learn a monolithic retrieval model to perform CIR in a unified latent space, thereby suffering from single-model myopia. In response, agentic methods reframe these existing retrieval models as a set of **atomic tools**, denoted as $\mathcal{T} = \{f_1, f_2, ..., f_m\}$. Each tool accepts the composed query input and returns a set of candidate images. This formulation allows us to treat the retrieval process not as a single inference step, but as a *compositional planning problem* over the space of available tools.

From Heuristic Exploration to Global Optimization. Existing agentic methods typically navigate this tool space via *heuristic exploration* (*e.g.*, ReAct loops (Yao et al., 2023)), where an LLM/VLM iteratively decides the next step based on local context. While flexible, they lack a global view of the solution space: agents may redundantly query overlapping concepts or fail to enforce strict exclusion constraints. We observe that for any given query q with *known* ground truth $\mathcal{I}^+$, the ideal tool usage can be mathematically formulated as a **Set Cover** (Balas and Padberg, 1972) variation: we seek a compact subset of atomic tool outputs whose union covers $\mathcal{I}^+$ while minimizing the overlap between this union and non-targets $\mathcal{I}^- = \mathcal{I} \setminus \mathcal{I}^+$. This realization allows us to reframe the planning process not as a trial-and-error search, but as a MIP problem that explicitly solves for the global optimal trajectory.

The OSCAR Framework. As illustrated in Figure 2, since the ground-truth set is *unknown* during the retrieval, OSCAR bridges the gap via a distinct *offline-online* paradigm that steers the retrieval process using golden demonstrations derived from training data.

- **Offline Optimization**. For each training sample, we first construct and perform *atomic retrievals*

by varying the query and top-k truncation. Then, we optimize the global planning trajectory via two-stage MIPs: (1) a *recall-oriented* MIP that identifies the optimal subset of atomic retrievals to maximize ground-truth coverage, and (2) a *precision-oriented* MIP that combines rigorous set-theoretic operations (union, intersection, difference) to filter irrelevant images. The resulting optimal trajectories are stored in a *golden library* to steer the online agentic planning for CIR.

- **Online Steering**. For a test query with unknown ground truth, we select optimal trajectories associated with similar queries from the golden library as in-context demonstrations. In this way, we effectively generalize the high-level logic derived offline and steer the agent toward near-optimal planning for CIR.

3.2 Atomic Retrieval Construction

To enable fine-grained planning, OSCAR first decomposes the complex user intent into a search space of discrete, executable actions. We define an **atomic retrieval** r as the fundamental unit of this space, formalized as a four-tuple:

$$r = (f,\ \hat{q},\ p,\ k), \tag{1}$$

where $f \in \mathcal{T}$ is the retrieval tool (*e.g.*, a caption-based searcher or a multimodal encoder), $\hat{q} = \{q_{img}, \hat{q}_{txt}\}$ is derived by rewriting the textual query, $p \in \{+, -\}$ indicates the polarity (positive for inclusion, negative for exclusion), and k is the number of candidate images to return (*i.e.*, top-k truncation). Each atomic retrieval r represents an indivisible planning unit that returns a specific candidate image set $\mathcal{S}_r \subset \mathcal{I}$.

Query Decomposition and Polarity. Rather than solely relying on the original query, we employ a VLM to generate a diverse set of rewritten queries. The VLM is prompted to decompose the modification instructions into specific visual attributes or semantic constraints. Crucially, each generated query $\hat{q}$ is assigned a polarity p. As illustrated in Figure 2, positive queries ($p = +$) target attributes that must be present in the result (*e.g.*, chickens), while negative queries ($p = -$) target attributes that must be explicitly excluded (*e.g.*, leaves). This decoupling allows the subsequent optimization stage to leverage boolean set logic for precise inclusion and exclusion, a capability that a single retrieval model lacks.

Top-k Truncation. The truncation parameter k dictates the trade-off between recall and precision for a given tool f. To enable the planning over different granularities of scope, we discretize the truncation parameter k into a finite set of levels:

$$\mathcal{K} = \{k_1, k_2, \ldots, k_{max}\}. \tag{2}$$

For a fixed tool f and query $\hat{q}$, the retrieved results are deterministic and monotonic w.r.t. k (*i.e.*, $\mathcal{S}_{k_1} \subset \mathcal{S}_{k_2}$ for $k_1 < k_2$). To ensure computational efficiency, we execute the retrieval once with the maximum truncation $k_{\max}$ and generate the smaller top-k variants via slicing, avoiding redundant model inferences.

By taking the Cartesian product over available tools f, rewritten queries $\hat{q}$, polarities p, and truncations k, we construct a diverse global set of atomic retrievals, denoted as $\mathcal{R}$ (1,182 atomic retrievals per sample). This set serves as the decision space for the subsequent offline optimization stages, where we select and compose the optimal subset of $\mathcal{R}$ to satisfy the global retrieval objective.

3.3 Recall-Oriented Selection MIP

The first-stage recall-oriented MIP aims to identify a compact subset of *positive* atomic retrievals $\mathcal{R}^+$ that maximizes the coverage of ground-truth images $\mathcal{I}^+$ while minimizing the inclusion of irrelevant noise $\mathcal{I}^-$. This selection process effectively prunes the massive atomic retrieval space constructed in Section 3.2, resulting in a recall-oriented candidate set of images $\mathcal{U} \subset \mathcal{I}$ for the subsequent fine-grained composition stage.

Optimization Variables. We define binary decision variables $x_r \in \{0, 1\}$ to indicate whether a positive atomic retrieval $r \in \mathcal{R}^+$ is selected. We further define auxiliary state variables, which are logically determined by the decision variable x_r, to track image coverage and tool diversity:

- $c_i \in \{0, 1\}$ indicates whether the image $i \in \mathcal{I}$ is covered by *at least one* selected atomic retrieval.
- $t_f \in \{0, 1\}$ indicates whether the tool f is used by *at least one* selected atomic retrieval, *i.e.*, active.

Moreover, to prevent tool redundancy, we group atomic retrievals into different *families* $\mathcal{F}$. A family $F \in \mathcal{F}$ consists of retrievals that share the same tool f, query $\hat{q}$, and polarity p, differing only in their truncation threshold k. Since larger k values

strictly subsume smaller ones (monotonicity), selecting multiple members from the same family provides diminishing returns.

MIP Formulation. We formulate the selection problem as a MIP that balances three competing objectives: (1) maximizing the ground-truth coverage $\sum_{i\in\mathcal{I}^+} c_i/|\mathcal{I}^+|$, (2) minimizing the irrelevant noise $\sum_{i\in\mathcal{I}^-} c_i/|\mathcal{I}^-|$, and (3) encouraging the tool diversity $\sum_{f\in\mathcal{T}} t_f$. The optimization formulation is written as:

$$\max_{\{x_r\}_{r\in\mathcal{R}^+}} \quad \frac{w_R}{|\mathcal{I}^+|}\sum_{i\in\mathcal{I}^+} c_i - \frac{w_P}{|\mathcal{I}^-|}\sum_{i\in\mathcal{I}^-} c_i + \lambda_{\text{div}}\sum_{f\in\mathcal{T}} t_f \tag{3}$$

$$\text{s.t.} \quad \sum_{r\in F} x_r \le 1, \quad \forall F\in\mathcal{F}, \tag{4}$$

$$x_r,\, c_i,\, t_f \in \{0,1\}. \tag{5}$$

The objective function in Eq. 3 prioritizes high recall (w_R) to ensure the ground truth is present in the candidate universe, while imposing a penalty (w_P) on expanding the search space with irrelevant images. The term λ_{div} serves as a regularizer to prevent over-reliance on a single tool, *i.e.*, single-model myopia. The constraint in Eq. 4 enforces *truncation exclusivity*: for any given tool and query configuration, the solver must pick at most one optimal k threshold, preventing redundant coverage of nested subsets.

Optimization Output. The solution to this MIP yields the optimal set of positive atomic retrievals, denoted as $\mathcal{R}^+_*$, which produces the recall-oriented candidate set of images $\mathcal{U} = \bigcup_{r\in\mathcal{R}^+_*}\mathcal{S}_r$. While $\mathcal{U}$ ensures high coverage of the ground truth, it inevitably contains irrelevant images since we omit the negative atomic retrieval. Hence, the next stage performs precision-oriented filtering to remove the remaining noise.

3.4 Precision-Oriented Composition MIP

While the first stage ensures high recall, the resulting candidate image set $\mathcal{U}$ inevitably contains significant noise since the negative atomic retrievals are not considered yet. Hence, we perform logical filtering and refine $\mathcal{U}$ by composing atomic retrievals into a rigorous boolean expression. To ensure stability and interpretability, we restrict the search space to a fixed two-clause structure:

$$\mathcal{S}_{final} = \underbrace{\Big(\bigcup_{r\in\mathcal{R}^+_{**}}\mathcal{S}_r\Big)}_{\text{Positive Union}} \setminus \underbrace{\Big(\bigcap_{r\in\mathcal{R}^-_*}\mathcal{S}_r\Big)}_{\text{Negative Intersection}} \tag{6}$$

where $\mathcal{R}^+_{**}\subseteq\mathcal{R}^+_*\subseteq\mathcal{R}^+$ and $\mathcal{R}^-_*\subseteq\mathcal{R}^-$ denote the optimal subsets of positive and negative atomic retrievals selected by this stage. $\mathcal{R}^+_*$ is the optimal solution from the previous stage. Crucially, we employ **Intersection** for the negative clause to enforce *conservative exclusion*: an image is removed only if *all* selected negative tools agree it is irrelevant, thereby preventing accidental deletion of ground-truth samples due to single-tool hallucinations.

Optimization Variables. We define binary decision variables $x_r \in \{0,1\}$ to indicate whether an atomic retrieval $r\in\mathcal{R}^+_*\cup\mathcal{R}^-$ is selected. Based on Eq. 6, these decisions determine the state of each candidate image $i\in\mathcal{U}$ via three logically coupled variables:

- $u_i \in \{0,1\}$ indicates if image i is covered by the *Positive Union*, *i.e.*, retrieved by *at least one* selected positive atomic retrieval.
- $v_i \in \{0,1\}$ indicates if image i falls into the *Negative Intersection*, *i.e.*, retrieved by *all* selected negative atomic retrievals.
- $z_i \in \{0,1\}$ indicates if image i remains in the final set $\mathcal{S}_{final}$. Logically, $z_i = 1$ if and only if $u_i = 1$ AND $v_i = 0$ (set difference).

MIP Formulation. The objective is to maximize the retention of ground-truth images in the final set $\mathcal{S}_{final}$. We introduce a regularization term λ_{reg} to mitigate the false positives and thereby prevent trivial solutions, *e.g.*, selecting no negative tools to maximize recall. The MIP is formulated as follows:

$$\max_{\{x_r\}_{r\in\mathcal{R}^+_*\cup\mathcal{R}^-}} \quad \sum_{i\in\mathcal{U}\cap\mathcal{I}^+} z_i - \lambda_{reg}\sum_{i\in\mathcal{I}^-} z_i \tag{7}$$

$$\text{s.t.} \quad \sum_{r\in\mathcal{R}^+_*} x_r \ge 1, \tag{8}$$

$$x_r, z_i \in \{0,1\}. \tag{9}$$

The constraint in Eq. 8 ensures that at least one positive atomic retrieval is selected, enforcing the positive union to be non-empty.

Optimization Output. The solution yields an optimal plan $(\mathcal{R}^+_{**}, \mathcal{R}^-_*)$ consisting of the selected

positive and negative atomic retrievals. Together, they define an optimal trajectory of agentic composed image retrieval for the training sample, *i.e.*, a sequence of specific tool calls associated with the rewritten queries and top-k truncations, and the corresponding set operations leading to the ground truth. These trajectories are stored in a *Golden Library* for online steering during test-time inference for agentic composed image retrieval.

3.5 Optimization-Steered Inference for CIR

The optimization pipeline described in Sections 3.3 and 3.4 is computationally intensive and relies on ground-truth availability, making it inapplicable during inference. However, the trajectories produced by these stages represent *golden demonstrations* of ideal planning logic. To transfer this optimized reasoning to the CIR agent, we construct a **Golden Library** that turns offline insights into retrieval-augmented in-context demonstrations.

For each training instance, we employ `Qwen3-Embedding-8B` to encode the problem context, which is the concatenation of the modification query q_{txt} and the *caption* of reference image q_{img}. The caption is produced by `Qwen3-VL-32B`. The resulting problem-context vector serves as the key for indexing, and the value is the corresponding two-stage MIP solution. At test time, as shown in Figure 2, given a composed query, we first compute its problem-context embedding and then retrieve the top-N most similar cases from the golden library by cosine similarity. These trajectories serve as the in-context demonstrations for the VLM planner , whose output is fed to the VLM verifier for final ranking.

Crucially, this process does not merely provide the answers to similar questions, but rather transfers **generalized planning logic**. By observing how the optimal planner handled similar semantic structures, the planner learns to replicate the underlying reasoning strategy, *e.g.*, selecting the appropriate tool types, determining the correct polarity for exclusion, and calibrating the truncation threshold k. The robustness of this logic transfer is evidenced by our experiments: OSCAR achieves state-of-the-art performance even when the golden library is constructed using only **10%** of the available training data, demonstrating that the system learns abstract meta-strategies rather than memorizing dataset-specific solutions.

We briefly discuss key design choices of OSCAR, with full implementation details and rigorous formulations provided in Appendix B.

4 Experiment

4.1 Experiment Setups

Datasets and Metrics. Our main experiments are conducted on on three representative public datasets for CIR: *CIRCO* (Baldrati et al., 2023a), *CIRR* (Liu et al., 2021) and *FashionIQ* (Wu et al., 2021). We report Recall@K on CIRR and FashionIQ, and mAP@K on CIRCO since each CIRCO query has multiple ground-truth images, following standard practice (Hou et al., 2025; Cheng et al., 2025). Detailed dataset statistics and splits are provided in Appendix C.

Baselines. We compare OSCAR with four groups of baselines:

- **Multimodal embedding models:** RzenEmbed-v2-7B (Jian et al., 2025), VLM2Vec (Jiang et al., 2024b), B3-Qwen2-7B (Thirukovalluru et al., 2025), Ops-MM-embedding-v1-7B (OpenSearch-AI, 2025), and QQMM-embed-v2 (Xue et al., 2025).
- **Caption-based text embedding models:** Qwen3-VL-32B (Bai et al., 2025a) for captioning, with bge-m3 (Chen et al., 2024) and Qwen3-Embedding (0.6B, 4B, 8B) (Li et al., 2026) for retrieval.
- **CIR-dedicated methods:** Pic2Word (Saito et al., 2023), SEARLE (Baldrati et al., 2023a), SEARLE-XL-OTI (Baldrati et al., 2023a), CIReVL (Karthik et al., 2024), LinCIR (Gu et al., 2024), LDRE (Yang et al., 2024b), and FiRE (Hou et al., 2025).
- **Agentic CIR methods:** MRA-CIR (Tu et al., 2025a), AutoCIR (Cheng et al., 2025), and X^R (Yang et al., 2026). OSCAR also belongs to this category.

Implementation Details Due to the page limitation, we provide the implementation details in Appendix D.

4.2 Main Results

Table 1 and Table 2 report the main results on standard CIR benchmarks. We also provide Recall@k performance between the baselines and our methods on CIRCO dataset in Appendix G.4. From the tables we can have the following observations:

- **Overall Performance.** Generally, OSCAR achieves the best performance across all datasets,

Table 1: Performance comparison on *CIRCO* and *CIRR* datasets. m@K and R@K denote mAP@K and Recall@K, respectively. Best results are highlighted in **bold**, while the second best are underlined. Relative improvement is calculated against the best baseline result. "-" means that the result is not reported in the original paper.

Type	Method	No Parameter Update?	CIRCO				CIRR			
			m@5	m@10	m@25	m@50	R@1	R@5	R@10	R@50
Multimodal Embedding	Ops-MM-v1-7B	✓	13.56	15.96	18.61	19.68	1.90	50.29	66.68	91.64
	RzenEmbed-v2-7B	✓	32.20	34.19	37.41	38.61	19.30	70.36	83.08	**96.77**
	VLM2Vec	✓	3.40	4.07	5.08	5.74	0.10	26.48	41.35	71.74
	B3_Qwen2_7B	✓	3.67	4.55	5.53	6.13	0.80	37.95	54.39	81.01
	QQMM-embed-v2	✓	45.92	47.13	50.39	51.45	28.98	73.66	82.98	96.72
Text Embedding	bge-m3	✓	8.15	8.61	9.62	10.21	12.53	34.00	48.63	75.06
	Qwen3-Embed-0.6B	✓	9.55	10.35	11.61	12.33	14.87	39.11	54.53	82.51
	Qwen3-Embed-4B	✓	13.55	14.60	16.33	17.20	22.17	51.57	64.84	88.19
	Qwen3-Embed-8B	✓	16.45	17.23	18.98	19.88	23.21	52.48	66.51	89.71
CIR Dedicated	Pic2Word	✗	8.72	9.51	10.46	11.29	23.90	51.70	65.30	87.80
	SEARLE	✗	11.68	12.73	14.33	15.12	24.24	52.48	66.29	88.84
	SEARLE-XL-OTI	✗	10.18	11.03	12.72	13.67	24.87	52.31	66.29	88.58
	CIReVL	✓	18.57	19.01	20.89	21.80	24.55	52.31	64.92	86.34
	LinCIR	✗	12.59	13.58	15.00	15.85	25.04	53.25	66.68	-
	LDRE	✓	23.35	24.03	26.44	27.50	26.53	55.57	67.54	88.50
	FiRE	✗	31.03	32.08	34.40	35.50	43.33	74.02	83.51	95.83
Agentic	MRA-CIR	✗	27.14	28.85	31.54	32.63	37.98	67.45	78.07	93.98
	AutoCIR	✓	24.05	25.14	27.35	28.36	31.81	61.95	73.86	92.07
	X^R	✓	31.38	32.88	35.46	36.50	43.13	73.59	83.09	94.05
	OSCAR (Ours)	✓	**56.54**	**58.53**	**61.92**	**62.67**	**51.18**	**79.50**	**87.45**	96.56
Relative Improvement (%)			23.13%	24.19%	22.88%	21.81%	18.67%	7.40%	5.25%	-0.22%

Table 2: Performance comparison on *FashionIQ* dataset.

Method	Dress		Shirt		Toptee		Average	
	R@10	R@50	R@10	R@50	R@10	R@50	R@10	R@50
Ops-MM-v1-7B	19.39	38.13	31.45	48.87	27.69	46.51	26.18	44.50
RzenEmbed-v2-7B	37.38	61.97	45.63	64.97	46.56	67.57	43.19	64.84
VLM2Vec	4.76	14.77	15.60	30.62	11.37	22.95	10.58	22.78
B3_Qwen2_7B	8.53	22.31	19.53	34.49	14.99	29.88	14.35	28.89
QQMM-embed-v2	36.44	60.09	**46.07**	65.65	46.35	68.49	42.95	64.74
bge-m3	10.81	23.75	20.66	33.66	15.96	27.33	15.81	28.25
Qwen3-Embed-0.6B	9.32	19.83	21.00	34.69	15.96	27.84	15.43	27.45
Qwen3-Embed-4B	12.69	27.02	26.69	40.68	21.88	36.05	20.42	34.58
Qwen3-Embed-8B	12.64	29.20	28.70	43.13	23.41	36.97	21.58	36.43
Pic2Word	20.20	40.20	26.20	43.60	27.90	47.40	24.77	43.73
SEARLE	20.48	43.13	26.89	45.58	29.32	49.97	25.56	46.23
SEARLE-XL-OTI	21.57	44.47	30.37	47.49	30.90	51.76	27.61	47.91
CIReVL	24.79	44.76	29.49	47.40	31.36	53.65	28.55	48.60
LinCIR	20.92	42.44	29.10	46.81	28.81	50.18	26.28	46.48
LDRE	22.93	46.76	31.04	51.22	31.57	53.64	28.51	50.54
FiRE	29.60	50.87	39.84	60.06	35.64	57.83	35.03	56.25
MRA-CIR	31.87	54.23	40.43	60.20	41.25	62.51	37.85	58.98
AutoCIR	24.94	45.81	34.00	53.43	33.10	55.58	30.68	51.61
X^R	28.71	52.50	38.91	56.82	43.91	62.57	37.18	57.30
OSCAR (Ours)	**38.47**	**65.15**	44.50	**67.52**	**48.24**	**71.24**	**43.73**	**67.97**
Rel. Improv. (%)	2.92%	5.13%	-3.40%	2.84%	3.61%	4.02%	1.25%	4.82%

Table 3: Ablation study on different variants of OSCAR. *FIQ.Avg* denotes average results over FashionIQ subsets.

Variants	CIRCO		CIRR		FIQ.Avg	
	m@25	m@50	R@10	R@50	R@10	R@50
OSCAR (Ours)	**61.92**	**62.67**	**87.45**	**96.56**	43.73	**67.97**
w/o Demo.	59.72	59.84	75.01	82.02	46.57	61.62
w/o Set.Diff.	59.54	59.62	87.32	93.80	46.46	62.08
w/o Set.Diff. & Demo.	58.63	58.66	39.11	54.53	**49.44**	53.96

consistently outperforming other baseline methods, with demonstrations derived from only **10%** training data. Notably, OSCAR is a training-free framework based on open-sourced models and requires no parameter update, even surpassing other baselines that are based on domain-specific-tuning or close-sourced LLMs.

- **Comparison with single-embedding methods.** OSCAR consistently outperforms both multimodal and text embedding baselines, showing that optimization-steered planning and structured tool composition are more effective than relying on a single unified representation.

- **Comparison with CIR-dedicated methods.** OSCAR outperforms CIR-dedicated methods, while our framework does not rely on specialized fusion, rewriting, or learned transformation modules. This suggests that explicit planning over general-purpose tools can be a reusable alternative to dataset-specific engineering.

- **Comparison with agentic approaches.** OSCAR consistently outperforms prior agentic retrieval methods, while avoiding iterative agent interaction or multi-round reasoning. Instead of relying on heuristic exploration, OSCAR guides VLMs with optimization-derived trajectories from the golden library, leading to more effective tool-call planning.

4.3 In-depth Analysis

4.3.1 Ablation Study

Table 3 analyzes the impact of golden trajectory demonstrations and set-theoretic operations in OSCAR, from which we can have the following observations:

Table 4: Performance of different VLM backbones on *CIRCO* dataset.

Model	m@25	m@50	R@25	R@50
QQMM-embed-v2	50.39	51.45	90.25	95.26
Qwen3-Embed-8B	18.98	19.88	53.12	65.62
FiRE	34.40	35.50	-	-
X^R	35.46	36.50	-	-
Qwen3-VL-4B	59.22	59.33	88.62	88.75
w/ OSCAR	**61.18**	**61.91**	**93.75**	**96.12**
Rel.Improv. (%)	3.31%	4.35%	5.79%	8.30%
Qwen3-VL-8B	59.22	59.31	88.12	88.25
w/ OSCAR	**61.65**	**62.40**	**94.88**	**97.12**
Rel.Improv. (%)	4.10%	5.21%	7.68%	10.05%
Qwen3-VL-32B	59.72	59.84	86.88	87.00
w/ OSCAR	**61.92**	**62.67**	**94.62**	**97.50**
Rel.Improv. (%)	3.68%	4.73%	8.91%	12.07%
InternVL3.5-38B	60.29	60.62	90.88	91.75
w/ OSCAR	**61.20**	**61.58**	**92.38**	**93.75**
Rel.Improv. (%)	1.51%	1.58%	1.65%	2.18%

- **Trajectory demonstrations**. Removing the optimal trajectory demonstrations at inference time (**w/o Demo.**) generally degrades the performance, indicating that the optimization-derived trajectories provide valuable signals about how retrieval tools should be selected and composed. Moreover, the golden library implicitly captures dataset- and query-specific patterns, including query intent styles, preferred composition strategies, and top-k selection biases, enabling the planner to adapt its trajectory planning across distributions.

- **Set-theoretic operations**. Performance degrades substantially when removing the set-difference or demo guidance. In particular, removing both makes the process close to a simple union of retrieved results, which is akin to multi-channel retrieval. As shown in Table 3, this variant generally performs poorly, with particularly large drops at Recall@50 across datasets. Naively aggregating outputs from multiple atomic retrievals is insufficient, as union-only composition cannot filter out items that match negative attributes or properly control the candidate scope.

4.3.2 Generalization of OSCAR

We further investigate the generalization of our proposed OSCAR in terms of different VLM planner backbones. We evaluate with Qwen3-4B/8B/32B and InternVL3.5-38B on the CIRCO dataset, and report the results in Table 4. We obtain the following observations:

- OSCAR generalizes well across different VLM backbones and consistently delivers substantial performance gains. This suggests that our offline-derived golden library captures broadly applicable planning logic, enabling OSCAR to function as a training-free, plug-and-play framework with strong model compatibility.

- OSCAR's gains over the VLM backbone largely rely on the VLM's agentic reasoning and tool-calling abilities. Accordingly, a stronger VLM (*e.g.*, Qwen3-VL-32B) may better exploit the golden library and yield larger improvements. This trend is also evident in our preliminary experiments with Qwen2.5-VL (Bai et al., 2025b) and MiniCPM-V (Yao et al., 2024): models with weaker tool-calling capabilities often fail to follow tool-use instructions and cannot reliably execute the required calls, leading to degraded performance.

4.4 Additional Analysis

Due to the page limitation, we move all other experiments to Appendix G, including the case study, robustness to the number of demonstrations, industrial-gallery evaluation, CIRCO Recall evaluation, OOD generalization analysis, tool combination analysis, and the controlled experiment.

5 Conclusion

We present OSCAR, an optimization-steered planning framework that reframes agentic composed image retrieval from a heuristic search process into a principled trajectory planning problem. By solving a two-stage MIP formulation offline, OSCAR mathematically derives optimal planning strategies, including query rewriting, truncation selection, and boolean set composition, without human annotation. These optimal trajectories are stored as golden demonstrations to effectively steer the VLM planner during inference to replicate complex reasoning logic in a single pass. Extensive experiments on three public benchmarks and private industrial user galleries demonstrate that OSCAR consistently outperforms SOTA baselines, offering both superior accuracy and robust generalization. Future work will explore extending this optimization-steered paradigm to other complex reasoning tasks, such as multi-hop question answering and open-ended tool use.

6 Limitations

OSCAR is designed for retrieval tasks, especially those that can be naturally decomposed into atomic retrievals and set-theoretic composition. For non-retrieval tasks, applying OSCAR would require redefining the action space, optimization objective, and composition logic, which is beyond the scope of this work.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Lorenzo Agnolucci, Alberto Baldrati, Marco Bertini, and Alberto Del Bimbo. 2024. isearle: Improving textual inversion for zero-shot composed image retrieval. *arXiv preprint arXiv:2405.02951*.

Anthropic. 2024. Introducing the model context protocol. Accessed: 2026-01-29.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, and etc. 2025a. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*.

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

Egon Balas and Manfred W Padberg. 1972. On the set-covering problem. *Operations Research*, 20(6):1152–1161.

Alberto Baldrati, Lorenzo Agnolucci, Marco Bertini, and Alberto Del Bimbo. 2023a. Zero-shot composed image retrieval with textual inversion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15338–15347.

Alberto Baldrati, Marco Bertini, Tiberio Uricchio, and Alberto Del Bimbo. 2023b. Composed image retrieval using contrastive learning and task-oriented clip-based features. *ACM Transactions on Multimedia Computing, Communications and Applications*.

Alberto Baldrati, Marco Bertini, Tiberio Uricchio, and Alberto Del Bimbo. 2022. Effective conditioned and composed image retrieval combining clip-based features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21466–21474.

Dimitris Bertsimas and John N Tsitsiklis. 1997. *Introduction to linear optimization*, volume 6. Athena scientific Belmont, MA.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *Preprint*, arXiv:2402.03216.

Zhangtao Cheng, Yuhao Ma, Jian Lang, Kunpeng Zhang, Ting Zhong, Yong Wang, and Fan Zhou. 2025. Generative thinking, corrective action: User-friendly composed image retrieval via automatic multi-agent collaboration. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 334–344.

Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z Wang. 2008. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys (Csur)*, 40(2):1–60.

Ginger Delmas, Rafael S Rezende, Gabriela Csurka, and Diane Larlus. 2022. Artemis: Attention-based retrieval with text-explicit matching and implicit similarity. In *International Conference on Learning Representations*.

Dongdong Ge, Qi Huangfu, Zizhuo Wang, Jian Wu, and Yinyu Ye. 2022. Cardinal optimizer (copt) user guide. *arXiv preprint arXiv:2208.14314*.

Geonmo Gu, Sanghyuk Chun, Wonjae Kim, , Yoohoon Kang, and Sangdoo Yun. 2024. Language-only training of zero-shot composed image retrieval. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Bohan Hou, Haoqiang Lin, Xuemeng Song, Haokun Wen, Meng Liu, Yupeng Hu, and Xiangyu Zhao. 2025. Fire: Enhancing mllms with fine-grained context learning for complex image retrieval. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 803–812.

Weijian Jian, Yajun Zhang, Dawei Liang, Chunyu Xie, Yixiao He, Dawei Leng, and Yuhui Yin. 2025. Rzenembed: Towards comprehensive multimodal retrieval. *arXiv preprint arXiv:2510.27350*.

Ziyan Jiang, Rui Meng, Xinyi Yang, Semih Yavuz, Yingbo Zhou, and Wenhu Chen. 2024a. Vlm2vec: Training vision-language models for massive multimodal embedding tasks. *arXiv preprint arXiv:2410.05160*.

Ziyan Jiang, Rui Meng, Xinyi Yang, Semih Yavuz, Yingbo Zhou, and Wenhu Chen. 2024b. Vlm2vec: Training vision-language models for massive multimodal embedding tasks. *arXiv preprint arXiv:2410.05160*.

Shyamgopal Karthik, Karsten Roth, Massimiliano Mancini, and Zeynep Akata. 2024. Vision-by-language for training-free compositional image retrieval. *International Conference on Learning Representations (ICLR)*.

Jongseok Kim, Youngjae Yu, Hoeseong Kim, and Gunhee Kim. 2021. Dual compositional learning in interactive image retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1771–1779.

Mingxin Li, Yanzhao Zhang, Dingkun Long, Keqin Chen, Sibo Song, Shuai Bai, Zhibo Yang, Pengjun Xie, An Yang, Dayiheng Liu, and 1 others. 2026. Qwen3-vl-embedding and qwen3-vl-reranker: A unified framework for state-of-the-art multimodal retrieval and ranking. *arXiv preprint arXiv:2601.04720*.

Haoqiang Lin, Haokun Wen, Xuemeng Song, Meng Liu, Yupeng Hu, and Liqiang Nie. 2024. Fine-grained textual inversion network for zero-shot composed image retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 240–250.

Zheyuan Liu, Cristian Rodriguez-Opazo, Damien Teney, and Stephen Gould. 2021. Image retrieval on real-life images with pre-trained vision-and-language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2125–2134.

Rui Meng, Ziyan Jiang, Ye Liu, Mingyi Su, Xinyi Yang, Yuepeng Fu, Can Qin, Zeyuan Chen, Ran Xu, Caiming Xiong, and 1 others. 2025. Vlm2vec-v2: Advancing multimodal embedding for videos, images, and visual documents. *arXiv preprint arXiv:2507.04590*.

OpenSearch-AI. 2025. Opensearch-ai/ops-mm-embedding-v1-7b. https://huggingface.co/OpenSearch-AI/Ops-MM-embedding-v1-7B.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

Kuniaki Saito, Kihyuk Sohn, Xiang Zhang, Chun-Liang Li, Chen-Yu Lee, Kate Saenko, and Tomas Pfister. 2023. Pic2word: Mapping pictures to words for zero-shot composed image retrieval. *Preprint*, arXiv:2302.03084.

Xuemeng Song, Haoqiang Lin, Haokun Wen, Bohan Hou, Mingzhu Xu, and Liqiang Nie. 2025. A comprehensive survey on composed image retrieval. *Preprint*, arXiv:2502.18495.

Yucheng Suo, Fan Ma, Linchao Zhu, and Yi Yang. 2024. Knowledge-enhanced dual-stream zero-shot composed image retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26951–26962.

Yuanmin Tang, Jing Yu, Keke Gai, Jiamin Zhuang, Gang Xiong, Yue Hu, and Qi Wu. 2024. Context-i2w: Mapping images to context-dependent words for accurate zero-shot composed image retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 5180–5188.

Yuanmin Tang, Jue Zhang, Xiaoting Qin, Jing Yu, Gaopeng Gou, Gang Xiong, Qingwei Lin, Saravan Rajmohan, Dongmei Zhang, and Qi Wu. 2025. Reason-before-retrieve: One-stage reflective chain-of-thoughts for training-free zero-shot composed image retrieval. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 14400–14410.

Raghuveer Thirukovalluru, Rui Meng, Ye Liu, Karthikeyan K, Mingyi Su, Ping Nie, Semih Yavuz, Yingbo Zhou, Wenhu Chen, and Bhuwan Dhingra. 2025. Breaking the batch barrier (b3) of contrastive learning via smart batch mining. *Preprint*, arXiv:2505.11293.

Rong-Cheng Tu, Wenhao Sun, Hanzhe You, Yingjie Wang, Jiaxing Huang, Li Shen, and Dacheng Tao. 2025a. Multimodal reasoning agent for zero-shot composed image retrieval. *Preprint*, arXiv:2505.19952.

Rong-Cheng Tu, Wenhao Sun, Hanzhe You, Yingjie Wang, Jiaxing Huang, Li Shen, and Dacheng Tao. 2025b. Multimodal reasoning agent for zero-shot composed image retrieval. *arXiv preprint arXiv:2505.19952*.

Nam Vo, Lu Jiang, Chen Sun, Kevin Murphy, Li-Jia Li, Li Fei-Fei, and James Hays. 2019. Composing text and image for image retrieval-an empirical odyssey. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6439–6448.

Teng Wang, Wing Yin Yu, Zhenqi He, Zehua Liu, HaileiGong HaileiGong, Han Wu, Xiongwei Han, Wei Shi, Ruifeng She, Fangzhou Zhu, and Tao Zhong. 2025. BPP-search: Enhancing tree of thought reasoning for mathematical modeling problem solving. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Laurence A Wolsey and George L Nemhauser. 1999. *Integer and combinatorial optimization*. John Wiley & Sons.

Hui Wu, Yupeng Gao, Xiaoxiao Guo, Ziad Al-Halah, Steven Rennie, Kristen Grauman, and Rogerio Feris. 2021. The fashion iq dataset: Retrieving images by combining side information and relative natural language feedback. *CVPR*.

Youze Xue, Dian Li, and Gang Liu. 2025. Improve multi-modal embedding learning via explicit hard negative gradient amplifying. *arXiv preprint arXiv:2506.02020*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, and etc. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Zhenyu Yang, Shengsheng Qian, Dizhan Xue, Jiahong Wu, Fan Yang, Weiming Dong, and Changsheng Xu. 2024a. Semantic editing increment benefits zero-shot composed image retrieval. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 1245–1254.

Zhenyu Yang, Dizhan Xue, Shengsheng Qian, Weiming Dong, and Changsheng Xu. 2024b. Ldre: Llm-based divergent reasoning and ensemble for zero-shot composed image retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 80–90.

Zhongyu Yang, Wei Pang, and Yingfang Yuan. 2026. Xr: Cross-modal agents for composed image retrieval. *arXiv preprint arXiv:2601.14245*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*.

Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, Qianyu Chen, Huarong Zhou, Zhensheng Zou, Haoye Zhang, Shengding Hu, Zhi Zheng, Jie Zhou, Jie Cai, Xu Han, and 4 others. 2024. Minicpm-v: A gpt-4v level mllm on your phone. *Preprint*, arXiv:2408.01800.

A Extended Related Works

A.1 Unified Embedding Retrieval for CIR

TIRG (Vo et al., 2019) introduces a gated residual composition mechanism, where the reference image embedding is modified via text-conditioned residual and gating functions. Building on VLMs, Baldrati et al. (2022) propose a conditioned composition framework that combines CLIP-based visual and textual features through a learned fusion module. CLIP4CIR (Baldrati et al., 2023b) builds upon CLIP by fine-tuning the vision–language encoders and learning a dedicated combiner network to fuse the reference image and modification text into a unified retrieval representation. ARTEMIS (Delmas et al., 2022) focuses on relative image descriptions and contrastive supervision to improve sensitivity to semantic changes, while still collapsing the composed query into a unified embedding. DC-Net (Kim et al., 2021) adopts a dual-branch transformation strategy, separately processing visual and textual features before fusing them into a single retrieval representation.

Pic2Word (Saito et al., 2023) projects image features into the textual embedding space as pseudo-word tokens, enabling image–text composition through pretrained text encoders. SEARLE (Agnolucci et al., 2024) further extends this idea by learning task-specific textual inversion or composition modules that encode visual modifications implicitly within a single embedding. Context-I2W (Tang et al., 2024) further maps images to context-dependent pseudo-word tokens conditioned on the modification intent, improving zero-shot composed retrieval under diverse edits. FTI4CIR (Lin et al., 2024) introduces fine-grained textual inversion by mapping an image into a subject-oriented token together with multiple attribute-oriented tokens, capturing finer visual details.

Beyond explicit composition mechanisms, some works focus on improving embedding learning strategies for CIR without altering the underlying retrieval formulation. Suo et al. (2024) enrich pseudo-word token mapping with an external database and an additional concept-alignment stream, improving fine-grained semantic alignment in the text embedding space. In contrast to token-based mapping, SEIZE (Yang et al., 2024a) adopts a training-free textual reasoning pipeline that generates diverse captions and leverages LLM-based compositional reasoning before retrieval.

Some works improve embedding learning for CIR without altering the composition mechanism. LinCIR (Gu et al., 2024) adopts a language-only self-supervised training strategy for zero-shot CIR, while FiRE (Hou et al., 2025) fine-tunes VLMs to enhance multimodal retrieval representations.

Within the unified embedding paradigm, CIReVL (Karthik et al., 2024) proposes a training-free framework following this paradigm, while LDRE (Yang et al., 2024b) explores LLM-based divergent reasoning and ensemble-style aggregation to improve coverage of possible semantic interpretations. Caption-based approaches make intermediate reasoning steps explicit in the language space; however, their effectiveness depends heavily on the quality of caption generation and rewriting, and may suffer from information loss when compressing visual content into text.

Recent advances in large-scale multimodal embedding models have substantially improved zero-shot retrieval performance. However, evaluations such as MMEB (Jiang et al., 2024a; Meng et al., 2025) show that no single embedding consistently performs well across diverse CIR scenarios; instead, models such as QQMM-embed (Xue et al., 2025), Rzenembed (Jian et al., 2025), Ops-MM-embedding (OpenSearch-AI, 2025) and VLM2Vec (Jiang et al., 2024a) display complementary strengths, motivating the use of multiple embedding tools rather than a single retriever.

Despite architectural and training differences, all these approaches ultimately reduce CIR to the unified-embedding model search, where complex compositional constraints such as inclusion and exclusion are implicitly encoded in a continuous representation.

A.2 Heuristic Agentic Retrieval for CIR

Recent work has explored agentic formulations for CIR, where the retrieval process is decomposed into multiple steps with specialized roles (*e.g.*, planning, retrieval, and correction), often involving iterative refinement or feedback. AutoCIR (Cheng et al., 2025) exemplifies this approach by structuring CIR as a multi-component pipeline that iteratively revises queries and retrieval results. MRA-CIR (Tu et al., 2025b) further advances this paradigm by introducing a multimodal reasoning agent that explicitly decouples intent reasoning from retrieval execution, bridging LLM-based logic and retrieval tools. Tang et al. (2025) propose a reflective reasoning-before-retrieval paradigm for

training-free zero-shot CIR, while X^R (Yang et al., 2026) introduces a cross-modal agent framework that coordinates specialized tools through synergistic planning across visual and textual modalities. While this line of work demonstrates the potential of multi-step reasoning, including generating alternative hypotheses and correcting failure cases, such agentic pipelines are typically iterative and computationally expensive due to repeated model interactions. Furthermore, these methods largely rely on heuristic exploration rather than explicit planning with optimality guarantees, leaving open the challenge of how to formalize and optimize tool execution under explicit global constraints.

A.3 Optimization-Based Planning

From a different perspective, Operations Research formulations model decision-making problems by explicitly defining decision variables, constraints, and objective functions, and solving them via combinatorial optimization such as MIP (Bertsimas and Tsitsiklis, 1997; Wolsey and Nemhauser, 1999; Wang et al., 2025). This paradigm enables principled and interpretable decision-making with explicit optimization objectives, rather than relying on heuristic or trial-and-error search.

Our work adopts this optimization perspective by formulating tool-call trajectory generation for CIR as a two-stage MIP problem. Each atomic retrieval corresponds to a discrete decision variable, while inclusion, exclusion, and composition constraints are explicitly modeled through set-level constraints. This formulation allows retrieval trajectories to be generated with explicit optimality objectives, distinguishing our approach from heuristic agentic planning.

B Additional Method Discussions

We clarify key design choices and implementation details to ensure the reproducibility and scientific rigor of the OSCAR framework.

Rigorous Enforcement of Logical Dependencies. In Sections 3.3 and 3.4, we described the relationships between primary decision variables (*i.e.*, x_r) and state variables (*e.g.*, c_i, t_f, z_i) using logical implications to prioritize conceptual clarity. In our actual implementation, these logical dependencies are strictly enforced via standard linear constraints (*e.g.*, Big-M formulations and logical inequalities). The complete, mathematically rigorous formulations, including all auxiliary constraints and bounds, are provided in Appendix I.

Table 5: The dataset statistics.

Split	Num	CIRCO	CIRR	FashionIQ		
				Dress	Shirt	Toptee
Training	#Query	-	28,225	5,985	5,988	6,027
	#Image	-	16,939	11,452	19,036	16,121
Validation	#Query	220	4,181	2,017	2,038	1,961
	#Image	123,403	2,297	3,817	6,346	5,373
Testing	#Query	800	4,148	-	-	-
	#Image	123,403	2,315	-	-	-

Rationale for Two-Stage Optimization. Theoretically, the recall and precision objectives could be unified into a monolithic MIP that solves for the global optimum in a single stage. However, in practice, a single-stage formulation suffers from a severe *combinatorial explosion*, rendering it computationally intractable for large-scale datasets. Our two-stage design adopts a *prune-and-refine* strategy: the first MIP acts as a high-recall filter to reduce the entire search space to a manageable candidate set $\mathcal{U}$, making the second composition MIP computationally tractable.

Negative Results and Failed Explorations. To provide a comprehensive view of the problem landscape and benefit the research community, we include a detailed discussion of negative results in the Appendix J, such as optimal-but-inefficient formulation for composition MIP and alternative trajectory retrieval strategies. By documenting these explorations, we aim to highlight not only the effective design of OSCAR but also the non-trivial challenges inherent in optimizing agentic retrieval trajectories.

C Dataset Statistics and Splits

Table 5 summarizes the statistics of the datasets used in our experiments. Specifically, CIRR is split into a training set, a validation set, and a testing set, while CIRCO has no training set. FashionIQ is composed of three subsets (dress, shirt and toptee), each containing a training set and a validation set. Following previous works (Cheng et al., 2025), we evaluate on the FashionIQ validation set, and on the CIRR and CIRCO testing sets through the official remote evaluation server. We leverage the validation set of CIRCO, the training set of CIRR and FashionIQ to build the golden library (only 10% of them).

D Implementation Details

For the tool planning process, we mainly use Qwen3-VL-32B (Bai et al., 2025a) as the planner agent. We adopt COPT (Ge et al., 2022) as

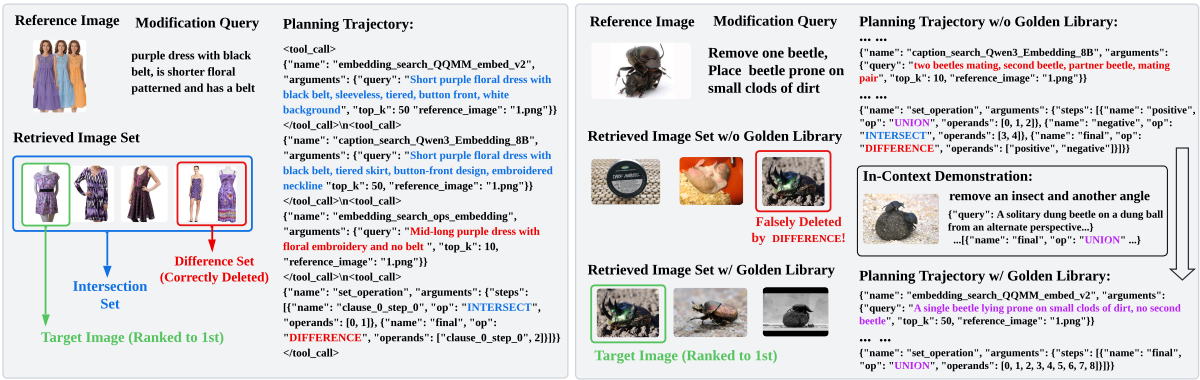

Figure 3: Case Studies on *FashionIQ* (left) and *CIRR* (right) datasets. The left part shows the correct tool call trajectory with the ground truth image ranked to the first place. The right part illustrates the effectiveness of golden library, with whose help the agent can avoid previous wrong tool calls and finally retrieve the ground truth image.

the MIP solver and discretize the top-k parameter described in Section 3.2 into the range from 5 to 50 with a step size of 5. The detailed prompt for the planner is demonstrated in Appendix M. We adopt Qwen3-VL-32B as the verifier agent to perform a binary relevance judgment between the candidates and queries. A detailed explanation of this binary scoring formulation is provided in Appendix L.

For the tools used by the planner, we select Ops-MM-v1-7B (OpenSearch-AI, 2025), RzenEmbed-v2-7B (Jian et al., 2025), QQMM-embed-v2 (Xue et al., 2025) as embedding-based retrieval tools, and bge-m3 (Chen et al., 2024), Qwen3-Embed-4B (Li et al., 2026), Qwen3-Embed-8B (Li et al., 2026) as caption-based retrieval tools. In addition, we introduce a dedicated tool that performs structured set operations on retrieved result sets (details are provided in Appendix F). All retrieval and composition tools are deployed via MCP servers (Anthropic, 2024). All experiments are conducted on A100 GPUs.

To emphasize generalization, we construct the golden library using only a limited subset of the available training data: specifically, we generate 220 cases for CIRCO, and sample 10% of the training set to construct cases for CIRR and FashionIQ.

E Statistics of Candidate Pruning

Table 6 reports the average number of candidates removed by the difference operation across different datasets. When applied appropriately, the difference operation can effectively reduce the size of the candidate set, thereby alleviating the computational burden on the verifier.

Table 6: Statistics of candidate pruning by the difference operation. Avg. Removed denotes the average number of candidates excluded when difference is used.

Dataset	Avg. Removed
CIRR	10.20
CIRCO	3.69
FashionIQ-Shirt	3.67
FashionIQ-Dress	3.05
FashionIQ-Toptee	2.51

F Set Operations

We consider three types of set operations: union, intersection, and difference. These operations enable explicit composition of results returned by multiple retrieval tools, producing a concise candidate set for subsequent ranking. Specifically, union is used to improve recall by aggregating candidates from different tools, intersection increases precision by retaining only commonly retrieved results, and difference filters out candidates that are likely to be irrelevant.

G More Experiments

G.1 Case Study

We present the case study on FashionIQ and CIRR datasets in Figure 3. The left panel shows a successful planning trajectory where structured set operations retrieve the target image and rank it in the first place. The right panel highlights the effect of the golden library: based on the in-context golden demonstration, the planner is steered towards more appropriate tool-call and set-operation choices and consequently retrieves the ground-truth image.

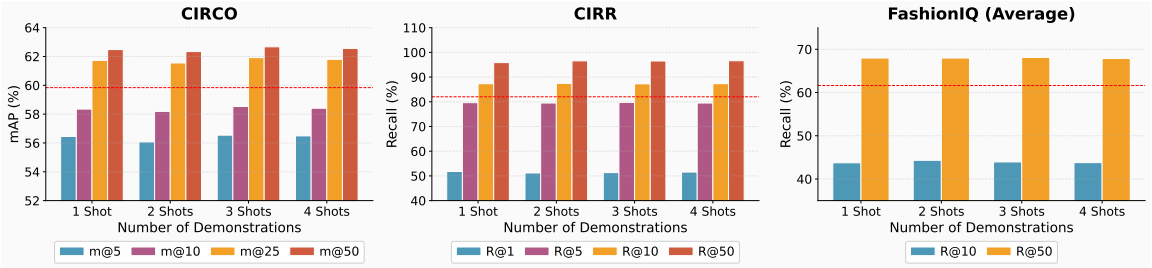

Figure 4: Performance comparison w.r.t. different numbers of demonstrations (*i.e.*, the number of shots) for inference-time steering. "m" and "R" denotes mAP and Recall. The red dashed line denotes the zero-shot performance of OSCAR (*i.e.*, mAP@50 on *CIRCO*, Recall@50 on *CIRR* and *FashionIQ*).

Table 7: Performance of real-world text-to-image retrieval on private industrial photo galleries. N@K and R@K denote NDCG@K and Recall@K, respectively. Best results are highlighted in bold, while the second best are underlined.

Model	Gallery1		Gallery2		Gallery3		Avg	
	N@10	R@10	N@10	R@10	N@10	R@10	N@10	R@10
VLM2Vec	44.29	47.78	44.91	48.69	38.96	41.86	42.72	46.11
B3_Qwen2_7B	42.87	47.32	41.84	45.71	37.49	40.76	40.73	44.60
Ops-MM-v1	51.78	53.74	48.95	51.44	43.98	46.99	48.24	50.72
RzenEmbed-v2-7B	51.04	53.79	49.62	51.93	46.23	47.59	48.96	51.10
QQMM-embed-v2	51.64	54.43	48.42	51.29	44.60	46.63	48.22	50.78
bge-m3	39.61	41.98	38.36	41.47	39.87	41.41	39.28	41.62
Qwen3-Embed-0.6B	40.73	42.78	40.11	42.54	39.60	41.11	40.15	42.14
Qwen3-Embed-4B	43.06	44.93	44.55	46.80	41.48	43.07	43.03	44.93
Qwen3-Embed-8B	42.92	45.05	44.57	46.86	41.89	43.13	43.13	45.01
X^R	52.49	56.17	50.23	48.72	49.74	56.91	50.82	53.93
OSCAR (Ours)	**56.01**	**65.28**	**53.12**	**57.43**	**58.70**	**63.92**	**55.94**	**62.21**
Rel.Improve. (%)	6.71%	16.22%	5.75%	10.59%	18.01%	12.32%	10.08%	15.36%

A common failure mode we observe is a *polarity mismatch in negative evidence*. The negative branch is intended to retrieve the undesired concept (*e.g.*, "more than one beetle" in Figure 1) and remove it via SET DIFFERENCE. However, the planner may instead retrieve the negated concept (*e.g.*, "one beetle") and mistakenly use it as the set to subtract. This reverses the intended set semantics and can remove valid targets. With our optimization-derived guidance, the planner is steered towards consistent negative evidence and set composition, preventing such erroneous exclusion.

G.2 Evaluation on Industrial Photo Galleries

To further assess the robustness of our OSCAR under real-world industrial scenarios, we evaluate it on a private in-the-wild dataset consisting of multiple user photo galleries. Each gallery corresponds to an individual user's photo collection and is associated with text queries for within-gallery image retrieval. This setting reflects practical album search scenarios, and the task becomes general text-to-image retrieval, where image distributions are highly diverse and queries often require non-trivial reasoning over user intentions beyond simple visual descriptions. Specifically, we test on three galleries composed of 1,069/1,466/1,047 real-world images and 483/336/369 real-user queries.

We report the performance in Table 7. We can observe that OSCAR consistently outperforms strong baselines across all galleries in terms of both metrics. On average, OSCAR achieves a relative improvement of 6.71% in NDCG@10 and 16.22% in Recall@10 over the best baseline. These results demonstrate that our optimization-steered tool-call trajectory planning and structured set-theoretic composition remain effective beyond public benchmarks, and generalize well to real-world user photo collections with natural data distributions and complex user intentions.

G.3 Robustness to the Number of Demonstrations

We study how the number of retrieved demonstrations from the golden library (*i.e.*, the number of shots) affects inference-time steering of the VLM planner. Specifically, we vary the number of demonstrations in {1, 2, 3, 4} and evaluate CIR performance under otherwise identical settings. As shown in Figure 4, OSCAR is largely insensitive to the number of demonstrations. The performance remains stable across all datasets as the shot count increases, with only minor fluctuations. This again suggests that the golden library primarily provides *generalizable planning trajectories*, rather than merely memorizing answers to similar queries. Consequently, adding more demonstrations yields limited additional information beyond a single trajectory.

G.4 CIRCO Recall

In addition to the widely used mAP metric on CIRCO, we also report Recall@K in Table 8. As shown in the table, OSCAR consistently achieves higher recall than unified embedding baselines, indicating its ability to retrieve a larger portion of

Table 8: Recall comparison on *CIRCO* dataset. Best results are highlighted in bold, while the second best are underlined. *Rel.Impr* denotes the relative improvement over the best baseline.

Method	R@5	R@10	R@25	R@50
Ops-MM-v1-7B	37.88	54.25	69.75	78.00
RzenEmbed-v2-7B	60.75	72.38	85.75	92.25
VLM2Vec	11.00	16.38	26.75	38.62
B3_Qwen2_7B	12.00	18.00	29.00	38.75
QQMM-embed-v2	71.12	79.75	90.25	95.26
bge-m3	14.62	20.62	32.25	40.62
Qwen3-Embed-0.6B	17.12	25.00	38.00	50.62
Qwen3-Embed-4B	25.62	37.38	51.25	62.62
Qwen3-Embed-8B	27.25	38.75	53.12	65.62
OSCAR (Ours)	**75.50**	**85.50**	**94.62**	**97.50**
Rel.Impr (%)	6.16	6.72	4.84	2.35

Table 9: Offline time cost analysis.

Dataset	Atomic Retrieval Construction	Recall-Oriented Selection MIP	Precision-Oriented Composition MIP	Total
CIRCO	12.03s	0.58s	2.62s	15.23s
CIRR	13.45s	0.21s	0.28s	13.94s
FashionIQ	19.44s	0.45s	2.00s	21.89s

target images.

G.5 Efficiency Analysis

We provide a breakdown of the time cost for both the offline and online stages of OSCAR, averaged per sample. The experiments are conducted on 4 A100 GPUs. The planner and verifier are Qwen3-VL-32B-Instruct deployed with vLLM, and the atomic retrieval tools (bge, Qwen3-Emb-4B/8B, QQMM, Ops, and RzenEmbed) are executed in parallel.

Offline Stage. Table 9 shows that the dominant cost comes from atomic retrieval construction, while the two MIP stages are relatively lightweight. This supports our claim that the two-stage offline design avoids combinatorial explosion in practice.

Online Stage. Table 10 shows that the main cost comes from the planning and verification stages, while case retrieval and candidate retrieval contribute only a small portion of the total latency. This indicates that OSCAR's online overhead is dominated by VLM reasoning, rather than by the retrieval or library lookup itself.

Basline Comparison. We further compare OSCAR with iterative agentic baselines under a fair setting using the same planner, verifier, and QQMM retriever. On CIRCO, OSCAR reduces the per-sample latency from 33.9s (AutoCIR) and 36.5s (XR) to 23.8s; on CIRR, it reduces the latency from 29.4s (AutoCIR) and 43.0s (XR) to 24.1s. Generally, OSCAR remains more efficient than iterative agentic pipelines, as it replaces multi-round trial-and-error interaction with a optimization-steered compositional retrieval plan.

Table 10: Online testing time cost analysis.

Dataset	Case Retrieval	Plan	Candidate Retrieval	Verification	Total
CIRCO	0.96s	7.93s	2.10s	12.79s	23.78s
CIRR	1.05s	11.53s	0.72s	10.84s	24.14s
FashionIQ	1.08s	12.99s	1.37s	5.03s	20.47s

Table 11: OOD generalization analysis. We build the library on one dataset and test on another.

Golden Library Set	Testing Set	R@10
CIRR	CIRCO	85.25
CIRCO	CIRCO	85.50
CIRCO	FIQ-Dress	38.17
FIQ-Dress	FIQ-Dress	38.47

G.6 OOD Generalization

We evaluate cross-dataset generalization of the Golden Library. Specifically, we build the library on one dataset and test on another. We show the results in Table 11, which demonstrate that the performance gap between cross-dataset and in-domain settings is small, suggesting that the Golden Library captures **transferable planning patterns** rather than merely memorizing dataset-specific cues. It's noteworthy that the Golden Library only uses 10% of the training data, further demonstrating the generalization ability.

G.7 Tool Selection & Pool Size Study

Tool Selection. We provide an ablation study in Table 12. In all settings, we use a fixed pool of three multimodal retrieval tools and three text retrieval tools.

- **Row 1 (original setting).** We use our default tool pool: multimodal tools QQMM, RzenEmb, and Ops-MM, together with text tools BGE, Qwen3-Emb-4B, and Qwen3-Emb-8B.
- **Row 2 (multimodal replacement).** We replace the multimodal tool pool with a **weaker** set: QQMM, VLM2Vec, and B3-Qwen2-7B.
- **Row 3 (text-tool replacement).** We replace the text tool pool with a more **homogeneous Qwen3-only** set: Qwen3-Emb-0.6B, Qwen3-Emb-4B, and Qwen3-Emb-8B.

The results suggest two main observations:

Table 12: Performance w.r.t. different tools on CIRCO.

Multimodal	Text	m@5	m@10	m@25	m@50
QQMM, RzenEmb, Ops-MM	Bge, Qwen3-4B, Qwen3-8B	**56.54**	58.53	**61.92**	**62.67**
QQMM, VLM2Vec, B3_Qwen2_7B	Bge, Qwen3-4B, Qwen3-8B	55.98	57.87	59.85	60.54
QQMM, RzenEmb, Ops-MM	Qwen3-0.6B, Qwen3-4B, Qwen3-8B	56.47	**58.62**	61.63	62.59

Table 13: Agentic methods w/ QQMM and Qwen3-VL-32B on CIRCO.

Method	m@5	m@10	m@25	m@50
AutoCIR	50.57	54.17	57.12	57.54
X^R	52.39	55.29	57.75	58.83
OSCAR (ours)	**56.54**	**58.53**	**61.92**	**62.67**

- **Tool capacity matters.** Stronger individual tools generally lead to better overall retrieval performance, as Row 1 consistently outperforms Row 2.
- **Tool diversity also matters.** A heterogeneous tool pool is more effective for handling complex and diverse query intents, as Row 1 outperforms Row 3, even though BGE itself is weaker than Qwen3-Emb-0.6B.

Tool Pool Size. We further study the effect of tool pool size on CIRCO. Starting from a minimal pool of QQMM + Qwen3-8B, we progressively add one multimodal tool and one text tool at each step. We observe that m@25 improves from 58.82 $\rightarrow$ 59.73 $\rightarrow$ 61.92 $\rightarrow$ 61.98 as the pool size increases from 2 to 8. This shows that enlarging the tool pool is generally beneficial, as it provides more complementary capabilities for diverse queries. At the same time, the gain becomes marginal once the pool is sufficiently rich.

G.8 Controlled Experiment

We provide a controlled comparison where all agentic methods are equipped with the same VLM and retrieval tool. We note that, in CIR evaluation, it is common practice to report the original results of existing baselines from their papers (Cheng et al., 2025; Yang et al., 2026; Baldrati et al., 2023a), since these methods are often designed and tuned with their own retrieval backbones and VLMs. Thus, for the main results in Table 1 and Table 2, we follow previous works and report the original number. However, to further rule out the possibility that OSCAR's gains come from using a stronger retrieval model or VLM, we conduct an additional controlled experiment. In our OSCAR framework, multiple retrieval models are treated as atomic tools. For a fair comparison, we provide the agentic baselines with the strongest single embedding model among OSCAR's atomic tools, and also control the VLM to be the same across all methods. As shown in Table 13, OSCAR still consistently outperforms these agentic baselines under this controlled setting.

We would like to highlight that OSCAR's advantage is not driven by simply using a strong model such as QQMM. As illustrated in Figure 1, a single retrieval model suffers from **single-model myopia**: it may perform well for certain query types but fail to capture other complementary retrieval signals. OSCAR addresses this limitation by systematically **unifying** diverse retrieval tools through optimization-steered planning. Instead of relying on one strong embedding space, OSCAR decomposes the query into atomic retrieval intents, selects and composes the outputs of different retrieval tools. Therefore, the improvement comes from tool coordination and structured planning, rather than from merely replacing the backbone with a stronger model.

H Term Clarification

H.1 On *"training free"*

We use **"training free"** to denote having no parameter updates in this paper, as OSCAR utilizes training data solely to generate In-Context Learning (ICL) demonstrations. In the agentic literature, this can be viewed as a few-shot rather than zero-shot approach, but it remains completely parameter-frozen.

H.2 On "atomic tool"

Agentic CIR methods may involve diverse subtasks beyond directly applying a retrieval model to the query. In OSCAR, these functionalities are naturally integrated into the retrieval process, *e.g.*, the planner may rewrite the query when generating tool calls, and each retrieval tool may internally include auxiliary procedures such as caption generation, depending on its design. OSCAR then abstracts the outputs of these heterogeneous tools as atomic retrieval sets for subsequent selection and composition.

I Implementation Details of two-stage MIPs

The common notations are summarized in Table 14.

I.1 Recall-Oriented Atomic Retrieval Selection

We introduce binary decision variables x_r to indicate if a positive atomic retrieval $r \in \mathcal{R}^+$ is selected. For image coverage, we use binary indicators y_i for $i \in \mathcal{I}^+$ and z_i for $i \in \mathcal{I}^-$ to denote whether a ground-truth image or a non-ground-truth image, respectively, is retrieved by at least one selected atomic retrieval.

Moreover, we group atomic retrievals into different *families* $\mathcal{F}$. A family $F \in \mathcal{F}$ consists of retrievals that share the same tool f, query $\hat{q}$, and polarity p, differing only in their truncation threshold k. We also introduce binary variables t_f to indicate whether any selected atomic retrieval is associated with tool f.

Based on the incidence definitions in Table 14, we define the following upper-bound coefficients:

$$\hat{a}_i := \sum_{r \in \mathcal{R}^+} a_{ir}, \qquad \hat{b}_i := \sum_{r \in \mathcal{R}^+} b_{ir},$$

which respectively count the number of positive atomic retrievals that retrieve image i as a target or non-target instance.

The recall-oriented atomic retrieval selection is formulated in the following MIP:

$$\max_{x,y,z,t} \quad w_R \frac{1}{|\mathcal{I}^+|} \sum_{i \in \mathcal{I}^+} y_i - w_F \frac{1}{|\mathcal{I}^-|} \sum_{i \in \mathcal{I}^-} z_i + \lambda_{\text{div}} \sum_f t_f \tag{1a}$$

$$\text{s.t.} \quad y_i \le \sum_{r \in \mathcal{R}^+} a_{ir} x_r \le \hat{a}_i \, y_i, \qquad \forall i \in \mathcal{I}^+, \tag{1b}$$

$$z_i \le \sum_{r \in \mathcal{R}^+} b_{ir} x_r \le \hat{b}_i \, z_i, \qquad \forall i \in \mathcal{I}^-, \tag{1c}$$

$$\sum_{r \in F} x_r \le 1, \qquad \forall F \in \mathcal{F}, \tag{1d}$$

$$t_f \le \sum_{r \in \mathcal{T}_f} x_r \le |\mathcal{T}_f| \, t_f, \qquad \forall f, \tag{1e}$$

$$x_r,\ y_i,\ z_i,\ t_f \in \{0, 1\}. \tag{1f}$$

The objective in (1a) prioritizes recall to retain sufficient candidates for subsequent composition, with diversity serving as a tie-breaker.

Constraints (1b)–(1c) ensure that an image is treated as retrieved if it is returned by at least one selected positive atomic retrieval.

Constraint (1d) therefore selects at most one variant per family, avoiding redundant inclusion of nested result sets.

Constraint (1e) introduces a mild tool-type diversity bias, encouraging solutions that draw from multiple retrieval tools.

The solution to this MIP yields the optimal set of positive atomic retrievals, denoted as $\mathcal{R}_*^+$, which produces the recall-oriented candidate set of images $\mathcal{U} = \bigcup_{r \in \mathcal{R}_*^+} \mathcal{S}_r$.

I.2 Precision-Oriented Logic Composition

We use binary variables x_r and w_r to indicate if a positive or negative atomic retrieval $r \in \mathcal{R}_*^+$ or $r \in \mathcal{R}^-$ is selected, respectively. For each image $i \in \mathcal{U}$, the binary variable e_i indicates whether i is included in the final composed result. We further introduce an auxiliary binary variable g_i to indicate whether image i lies in the intersection of the selected negative atomic retrieval result sets.

Table 14: Notation for the Problem Formulation

Symbol	Description
$\mathcal{I}$	An image gallery
$\mathcal{R}^+$	Set of positive atomic retrievals
$\mathcal{R}^-$	Set of negative atomic retrievals.
$r \in \mathcal{R}^+ \cup \mathcal{R}^-$	An atomic retrieval.
$\mathcal{S}_r$	Set of candidate images returned by atomic retrieval call r, where $\mathcal{S}_r \subseteq \mathcal{I}$.
$\mathcal{I}^+$	Set of ground-truth images, $\mathcal{I}^+ \subseteq \mathcal{I}$.
$\mathcal{I}^-$	Set of non-ground-truth images, $\mathcal{I}^- = \mathcal{I} \setminus \mathcal{I}^+$.
$i \in \mathcal{I}$	A candidate image.
a_{ir}	$a_{ir} = \mathbf{1}[\, i \in \mathcal{I}^+ \cap \mathcal{S}_r \,],\ r \in \mathcal{R}^+$.
b_{ir}	$b_{ir} = \mathbf{1}[\, i \in \mathcal{I}^- \cap \mathcal{S}_r \,],\ r \in \mathcal{R}^+$.
$\mathcal{R}_*^+$	Set of positive atomic retrievals from Stage I.
$\mathcal{U}$	Positive candidate universe induced by the selected positive retrievals from Stage I.
c_{ir}	$c_{ir} = \mathbf{1}[\, i \in \mathcal{U} \cap \mathcal{S}_r \,],\ r \in \mathcal{R}_*^+$.
d_{ir}	$d_{ir} = \mathbf{1}[\, i \in \mathcal{U} \cap \mathcal{S}_r \,],\ r \in \mathcal{R}^-$.
$\mathcal{R}_{**}^+$	Set of positive atomic retrievals from Stage II.
$\mathcal{R}_*^-$	Set of negative atomic retrievals from Stage II.

The precision-oriented logic composition is formulated in the following MIP:

$$\max_{x,w,r,g} \quad \sum_{i \in \mathcal{U} \cap \mathcal{I}^+} e_i \tag{2a}$$

$$\text{s.t.} \quad e_i \le \sum_{r \in \mathcal{R}_*^+} c_{ir}\, x_r, \qquad \forall i \in \mathcal{U} \tag{2b}$$

$$e_i \le 1 - g_i, \qquad \forall i \in \mathcal{U} \tag{2c}$$

$$g_i \le d_{ir} + (1 - w_r), \qquad \forall i \in \mathcal{U},\ \forall r \in \mathcal{R}^- \tag{2d}$$

$$g_i \ge 1 - \sum_{r \in \mathcal{R}^-} (1 - d_{ir})\, w_r, \qquad \forall i \in \mathcal{U} \tag{2e}$$

$$\sum_{r \in \mathcal{R}_*^+} x_r \ge 1 \tag{2f}$$

$$x_r,\ w_r,\ e_i,\ g_i \in \{0, 1\}. \tag{2g}$$

The objective (2a) is precision-oriented by design and encourages retaining as many target images as possible after filtering.

Constraint (2b) allows an item to be included in the final result if it is retrieved by at least one selected positive atomic retrieval.

Constraint (2c) implements exclusion by preventing items that satisfy the negative clause from being returned.

Constraints (2d)–(2e) ensure that $g_u = 1$ if and only if image u is retrieved by all selected negative atomic retrievals.

Constraint (2f) prevents degenerate solutions by enforcing that at least one positive tool is selected, ensuring a non-empty positive clause in the two-clause composition.

The solution yields an optimal plan $(\mathcal{R}_{**}^+, \mathcal{R}_*^-)$ consisting of the selected positive and negative atomic retrievals, which can be used to construct the golden library.

J Alternative Stage II: F1-Optimized DNF Composition

Stage II adopts a fixed set-composition structure for stability. Here, we consider a more flexible alternative that represents logical composition in disjunctive normal form (DNF) and selects clauses via an MIP to directly optimize an F1-style objective.

To handle negative atomic retrievals, we convert set difference into intersection via relative complements. Specifically, for any two sets A and B with $A, B \subseteq \mathcal{U}$, we apply the identity

$$A \setminus B \;=\; A \cap (\mathcal{U} \setminus B) \;=\; A \cap \overline{B}, \tag{10}$$

where $\mathcal{U}$ denotes the positive candidate universe induced by Stage I, and $\overline{B}$ denotes the relative complement of B with respect to $\mathcal{U}$, *i.e.*, $\overline{B} := \mathcal{U} \setminus B$.

Accordingly, each clause C_c is defined as

$$C_c = \bigcap_{r \in P_c} \tilde{\mathcal{S}}_r, \qquad \tilde{\mathcal{S}}_r = \begin{cases} \mathcal{S}_r, & r \in \mathcal{R}_*^+, \\ \mathcal{U} \setminus \mathcal{S}_r, & r \in \mathcal{R}^-, \end{cases} \tag{11}$$

where $P_c \subseteq \mathcal{R}_*^+ \cup \mathcal{R}^-$ denotes the set of atomic retrievals included in clause c.

We enumerate clauses prior to optimization by imposing constraints on the maximum clause length and the number of negative atomic retrievals per clause, yielding a finite candidate set $\{C_c\}_{c=1}^{C}$. We introduce a binary indicator u_c to denote whether clause c is selected, and define the final retrieval result as the union of selected clauses:

$$\mathcal{O} = \bigcup_{c:u_c=1} C_c. \tag{12}$$

We define clause-level incidence matrices

$$\alpha_{ic} = \mathbb{1}[\, i \in \mathcal{I}^+ \cap C_c \,], \qquad \beta_{ic} = \mathbb{1}[\, i \in \mathcal{I}^- \cap C_c \,],$$

together with the corresponding upper-bound coefficients

$$\hat{\alpha}_i := \sum_c \alpha_{ic}, \qquad \hat{\beta}_i := \sum_c \beta_{ic}.$$

Binary indicators y_i and z_i are used to indicate whether image i is included in the final result $\mathcal{O}$. We further impose a budget constraint that limits the number of selected clauses to at most M.

We formulate logic composition in the following MIP:

$$\max_{u,y,z} \quad 2 \sum_{i \in \mathcal{I}^+} y_i - \lambda \Big(|\mathcal{I}^+| + \sum_{i \in \mathcal{I}^+} y_i + \sum_{i \in \mathcal{I}^-} z_i \Big) - \alpha \sum_c |P_c|\, u_c \tag{3a}$$

$$\text{s.t.} \quad y_i \le \sum_c \alpha_{ic} u_c \le \hat{\alpha}_i\, y_i, \qquad \forall i \in \mathcal{I}^+, \tag{3b}$$

$$z_i \le \sum_c \beta_{ic} u_c \le \hat{\beta}_i\, z_i, \qquad \forall i \in \mathcal{I}^-, \tag{3c}$$

$$\sum_c u_c \le M, \tag{3d}$$

$$u_c,\, y_i,\, z_i \in \{0, 1\}. \tag{3e}$$

The F1 score corresponding to the selected clause union $\mathcal{O} = \bigcup_{c:u_c=1} C_c$ is given by

$$F_1(\mathcal{O}) = \frac{2 \sum_{i \in \mathcal{I}^+} y_i}{|\mathcal{I}^+| + \sum_{i \in \mathcal{I}^+} y_i + \sum_{i \in \mathcal{I}^-} z_i}, \tag{13}$$

and the parameter λ in (3a) is updated via Dinkelbach iteration as

$$\lambda \leftarrow \frac{2 \sum_{i \in \mathcal{I}^+} y_i}{|\mathcal{I}^+| + \sum_{i \in \mathcal{I}^+} y_i + \sum_{i \in \mathcal{I}^-} z_i}. \tag{14}$$

The objective in (3a) is obtained via the Dinkelbach reformulation of the fractional F1 objective in (13). At each iteration, an MIP is solved to maximize the difference between the numerator and the denominator scaled by the current value of λ, and λ is updated to the resulting F1 score until convergence. The complexity penalty $\alpha \sum_c |P_c|\, u_c$ discourages selecting long clauses. Constraints (3b)–(3c) link clause selection to image-level inclusion through y_i and z_i, and Constraint (3d) limits the number of selected clauses.

In practice, allowing unions over many enumerated clauses often produces bloated and unintuitive DNF expressions that are difficult to interpret or learn. For example, the resulting composition may take the form

$$(A \setminus B) \;\cup\; (A \cap C) \;\cup\; D,$$

where the underlying retrieval logic is fragmented across multiple overlapping clauses. To facilitate effective learning of tool-call trajectories by the planner, we therefore adopt the less flexible but more structured Stage II formulation in Section 3.4 in the main method.

K Expressive Completeness Proof

The predefined two-clause composition (Appendix I) in OSCAR is a practical choice for **stability and MIP tractability**, rather than a fundamental limit of expressiveness. More generally, atomic retrieval outputs generate a Boolean set algebra, and **any retrieval target** composed from them via union, intersection, and difference can be rewritten in disjunctive normal form (DNF). Hence, when extended to more complex queries, OSCAR can be naturally generalized from the current two-clause composition to a richer multi-clause DNF program under the same optimization-steered framework. We provide proof of DNF Completeness of Atomic Retrieval Composition below.

Theorem 1 (DNF Completeness of Atomic Retrieval Composition)**.** *Let U denote the candidate universe, and let $\{S_r\}_{r\in\mathcal{R}}$ be atomic retrieval sets with $\cup, \cap, \setminus$. For any $E \in \mathfrak{A}$, there exists*

$$E = \bigcup_{j=1}^{m} \bigcap_{r\in C_j} L_r, \quad L_r \in \{S_r,\, U \setminus S_r\}. \tag{15}$$

Proof. For each $u \in U$, define $X_r(u) = \mathbf{1}[u \in S_r]$. Any E induces a Boolean formula $\varphi(\{X_r(u)\})$ with:

$$\cup \leftrightarrow \vee, \quad \cap \leftrightarrow \wedge, \quad A \setminus B \leftrightarrow A \wedge \neg B. \tag{16}$$

By the DNF theorem, φ can be written as

$$\varphi \equiv \bigvee_{j=1}^{m} \bigwedge_{r\in C_j} \ell_{jr}, \quad \ell_{jr} \in \{X_r, \neg X_r\}. \tag{17}$$

Mapping back:

$$X_r \mapsto S_r, \quad \neg X_r \mapsto U \setminus S_r, \tag{18}$$

gives

$$E = \bigcup_{j=1}^{m} \bigcap_{r\in C_j} L_r. \tag{19}$$

This establishes the expressive completeness of atomic retrieval composition, implying that OSCAR can in principle generalize to more complex queries beyond the current two-clause instantiation.

□

L Binary Logit-Based Relevance Scoring

Our verifier computes a relevance score by comparing the likelihood of generating "yes" and "no" as the *next decoded token*. Let $\mathcal{V}$ denote the vocabulary, and let z_t be the model's output logit for token $t \in \mathcal{V}$ under a fixed context which consists of a reference image, a modification query, and a candidate image. The corresponding prompt for the verifier is provided in Appendix M.

Step 1: Next-token distribution over the whole vocabulary. By definition, VLM induces a normalized probability distribution over the next token via softmax:

$$P(t) \;=\; \frac{\exp(z_t)}{\sum\limits_{v\in\mathcal{V}} \exp(z_v)}, \quad t \in \mathcal{V}. \tag{20}$$

In particular,

$$P(\mathsf{yes}) \;=\; \frac{\exp(z_{\mathsf{yes}})}{\sum\limits_{v\in\mathcal{V}} \exp(z_v)}, \qquad P(\mathsf{no}) \;=\; \frac{\exp(z_{\mathsf{no}})}{\sum\limits_{v\in\mathcal{V}} \exp(z_v)}. \tag{21}$$

Step 2: Restricting to a binary decision and renormalization. We interpret relevance as a binary decision within the subset $\{\mathsf{yes}, \mathsf{no}\}$.

$$\begin{aligned}
P(\mathsf{yes} \mid \mathsf{yes\ or\ no}) &= \frac{P(\mathsf{yes})}{P(\mathsf{yes}) + P(\mathsf{no})} && (22)\\
&= \frac{\frac{\exp(z_{\mathsf{yes}})}{\sum_{v\in\mathcal{V}} \exp(z_v)}}{\frac{\exp(z_{\mathsf{yes}})}{\sum_{v\in\mathcal{V}} \exp(z_v)} + \frac{\exp(z_{\mathsf{no}})}{\sum_{v\in\mathcal{V}} \exp(z_v)}} && (23)\\
&= \frac{\exp(z_{\mathsf{yes}})}{\exp(z_{\mathsf{yes}}) + \exp(z_{\mathsf{no}})} && (24)\\
&= \frac{1}{1 + \exp(z_{\mathsf{no}} - z_{\mathsf{yes}})} && (25)\\
&= \sigma(z_{\mathsf{yes}} - z_{\mathsf{no}})\,. && (26)
\end{aligned}$$

where $\sigma(x) = \frac{1}{1+\exp(-x)}$ is the sigmoid function. In our implementation, we use this binary probability as the relevance score $s_u = \sigma(z_{yes} - z_{no})$.

M Prompts

For completeness, we present the planner, image captioning, and verifier prompts used in our OSCAR framework below.

Planner Agent Prompt

Role: You are a search planner for Composed Image Retrieval (CIR).
Task: Given a reference image and a modification query describing "Changes" and "Shared concept", find target images that match the modified concept.

Available Models:
- Embedding models: *rzenembed*, *ops_embedding*, *QQMM_embed_v2*
- Caption models: *bge_m3*, *Qwen3_Embed_4B*, *Qwen3_Embed_8B*

Available Tools:
- *embedding_search_{model_name}(query, top_k, reference_image)*: Visual similarity search. Best for visual attributes, colors, spatial relationships.
- *caption_search_{model_name}(query, top_k, reference_image)*: Semantic text search on captions. Best for semantic descriptions, conceptual changes.
- *set_operation(steps)*: Combine results (INTERSECT, UNION, DIFFERENCE). **MUST be the last tool call when using 2+ tools.**

CRITICAL - Set Operation Rules:
- **INTERSECT (AND):** Return items in ALL operands. Use when results must satisfy multiple criteria.
- **UNION (OR):** Return items in ANY operand. Use for maximum coverage.
- **DIFFERENCE (A - B):** Return items in first but NOT in second. Use to exclude specific unwanted features.

Strategy - How to Handle CIR Queries:
Step 1: Analyze Additions, Removals, or Changes
- **When to use DIFFERENCE:** Only when an object is COMPLETELY ABSENT (*e.g.*, "has no X", "without X").
- **When NOT to use DIFFERENCE:** If the object exists but changed attributes (*e.g.*, "has fewer people", "is in greyscale", "has different color").
- **Negative Query Rule:** Negative queries must ONLY contain the absent object. NEVER include shared concepts (*e.g.*, if shared is "cake", negative query should be "toppings", NOT "cake with toppings").

Step 2: Generate Tool Calls
- **Positive tools:** Search for "Shared concept" + additions.
- **Negative tools:** ONLY search for objects that DON'T EXIST.

Step 3: Set Operation (Mandatory)
- Union all positive tools → "positive".
- Intersect all negative tools → "negative".
- *DIFFERENCE(positive, negative)* → "final".

Static Reference Examples (Learn the format!)

Below are examples showing the correct tool call FORMAT (not related to your query):

```
Query: Changes: shows two people and has a more colorful background. Shared
    concept: a girl with a traditional Chinese umbrella.
Reference image: 000000271520.jpg
Reference caption: A young woman in traditional Chinese attire performs
    gracefully on a dimly lit stage, holding a large, ornate red parasol.

Analysis:
  - Shared: girl with Chinese umbrella/parasol, traditional attire
  - Changes: ADD two people, ADD colorful background (instead of dimly lit stage
    )
    - No removals, No DIFFERENCE needed, just UNION
  - Search strategy: Find images with girl + Chinese umbrella + multiple people
    + colorful setting


<tool_call>
{"name": "embedding_search_QQMM_embed_v2", "arguments": {"query": "Two people
    with traditional Chinese umbrella parasol, colorful vibrant background", "
    top_k": 20, "reference_image": "000000271520.jpg"}}
</tool_call>
<tool_call>
{"name": "embedding_search_rzenembed", "arguments": {"query": "Girl with Chinese
     parasol umbrella, two people together, colorful setting", "top_k": 15, "
    reference_image": "000000271520.jpg"}}
```

```
</tool_call>
<tool_call>
{"name": "embedding_search_ops_embedding", "arguments": {"query": "Chinese
    umbrella, traditional dress, multiple people, colorful background", "top_k":
    15, "reference_image": "000000271520.jpg"}}
</tool_call>
<tool_call>
{"name": "caption_search_Qwen3_Embedding_8B", "arguments": {"query": "girl with
    Chinese umbrella parasol, two people, colorful vibrant background", "top_k":
    10, "reference_image": "000000271520.jpg"}}
</tool_call>
<tool_call>
{"name": "set_operation", "arguments": {"steps": [  {"name": "final", "op": "
    UNION", "operands": [0, 1, 2, 3]}]}}
</tool_call>
```

Example2 - DIFFERENCE (Remove elements)

```
Query: Changes: has no toppings and shows the rest in the background. Shared
    concept: a slice of cake on a plate in the foreground.
Reference image: 000000119289.jpg
Reference caption: A slice of chocolate cake with white frosting and cherry
    toppings sits on a white plate.
Analysis:
  - Shared: slice of cake on plate, foreground
    - Changes: REMOVE toppings (no cherry, no frosting decorations), ADD
    background elements
  - has no toppings, toppings are ABSENT: Use DIFFERENCE
  -  CRITICAL: cake EXISTS in target! NEVER include cake in negative query!
    Search strategy: Find cake slices WITHOUT toppings/decorations, with
    background visible.Use DIFFERENCE to exclude images with visible toppings.
    Warning: Negative query ONLY contains toppings/cherries/decorations, NOT
    cake!

<tool_call>
{"name": "embedding_search_QQMM_embed_v2", "arguments": {"query": "Plain slice
    of cake on plate, simple cake, background visible", "top_k": 20, "
    reference_image": "000000119289.jpg"}}
</tool_call>
<tool_call>
{"name": "embedding_search_rzenembed", "arguments": {"query": "Simple cake slice
     on plate in foreground, background scene", "top_k": 15, "reference_image": "
    000000119289.jpg"}}
</tool_call>
<tool_call>
{"name": "embedding_search_ops_embedding", "arguments": {"query": "Cake slice
    plate foreground, plain simple, background visible", "top_k": 25, "
    reference_image": "000000119289.jpg"}}
</tool_call>
<tool_call>
{"name": "caption_search_Qwen3_Embedding_8B", "arguments": {"query": "plain cake
     slice on plate, background visible", "top_k": 10, "reference_image": "
    000000119289.jpg"}}
</tool_call>
<tool_call>
{"name": "embedding_search_QQMM_embed_v2", "arguments": {"query": "cherry
    topping, frosting decorations, fruit toppings, garnish", "top_k": 20, "
    reference_image": "000000119289.jpg"}}
</tool_call>
<tool_call>
{"name": "embedding_search_rzenembed", "arguments": {"query": "cherries, whipped
     cream topping, dessert decorations, garnish", "top_k": 20, "reference_image"
    : "000000119289.jpg"}}
</tool_call>
<tool_call>
{"name": "caption_search_Qwen3_Embedding_8B", "arguments": {"query": "toppings,
    cherries, frosting decorations, garnished dessert", "top_k": 10, "
    reference_image": "000000119289.jpg"}}
```

```
</tool_call>
<tool_call>
{"name": "set_operation", "arguments": {"steps": [{"name": "positive", "op": "
    UNION", "operands": [0, 1, 2, 3]},  {"name": "negative", "op": "INTERSECT", "
    operands": [4, 5, 6]}, {"name": "final", "op": "DIFFERENCE", "operands": ["
    positive", "negative"]}]}}
</tool_call>
```

RETRIEVED SIMILAR CASES
Below are some similar cases retrieved based on your query and reference image. You can learn these cases patterns for:
- Query Rewrite
- Tool selection and composition
- top_k values

{retrieved_cases}

Now solve the following task: Modification Query: {query}
Reference Image: {ref_img}
Reference Image Caption: {ref_caption}

Now generate tool_calls to retrieve target images based on the provided Modification Query and Reference Image.

Image Captioning Prompt

Describe this image comprehensively:
1. Main subjects and their attributes
2. Actions, interactions, and relationships
3. Scene, background, and context
4. Any visible text (signs, labels, watermarks, etc.)
Return your response as JSON:
```json
{
"caption": "your comprehensive description here"
}
```

Verifier Agent Prompt

{Reference image}
{Candidate image}
Modification query: {<query>}

Instruction:
You are given two images.
The first image is the reference image.
The second image is a candidate image.
Given the modification query: {<query>}, determine whether the candidate image matches what you would expect after applying the modification to the reference image.
Answer with **"yes"** or **"no"** only.