

# Composed Object Retrieval: Object-level Retrieval via Composed Expressions

Tong Wang<sup>1,2</sup>, Guanyu Yang<sup>†1</sup>, Nian Liu<sup>†2,3</sup>, Zongyan Han<sup>2</sup>, Jinxing Zhou<sup>2</sup>, Salman Khan<sup>2</sup>, Fahad Shahbaz Khan<sup>2</sup>

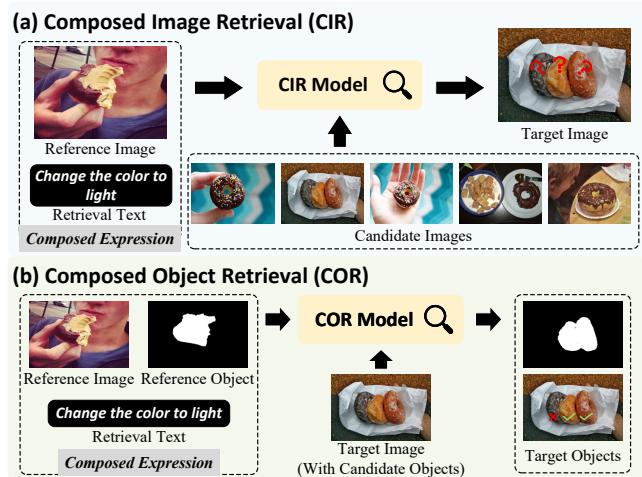
<sup>1</sup>Southeast University    <sup>2</sup>Mohamed bin Zayed University of Artificial Intelligence    <sup>3</sup>Northwestern Polytechnical University  
<sup>†</sup>Corresponding author.

**Abstract:** Retrieving fine-grained visual content based on user intent remains a challenge in multi-modal systems. Although current Composed Image Retrieval (CIR) methods combine reference images with retrieval texts, they are constrained to image-level matching and cannot localize specific objects. To this end, we propose **Composed Object Retrieval (COR)**, a brand-new task that goes beyond image-level retrieval to achieve object-level precision, allowing the retrieval and segmentation of target objects based on composed expressions combining reference objects and retrieval texts. COR presents significant challenges in retrieval flexibility, which requires systems to identify arbitrary objects satisfying composed expressions while avoiding semantically similar but irrelevant negative objects within the same scene. We construct **COR127K**, the first large-scale COR benchmark that contains 127,166 retrieval triplets with various semantic transformations in 408 categories. We also present **CORE**, a unified end-to-end model that integrates reference region encoding, adaptive visual-textual interaction, and region-level contrastive learning. Extensive experiments demonstrate that CORE significantly outperforms existing models in both base and novel categories, establishing a simple and effective baseline for this challenging task while opening new directions for fine-grained multi-modal retrieval research. Code will be released publicly at <https://github.com/wangtong627/COR>.

## 1 Introduction

Image retrieval aims to accurately match user queries with relevant content. However, traditional single-modal methods [1, 2] struggle with subtle semantics and personalized, fine-grained content needs, particularly as multi-modal data grows in scale and complexity. Recently, Composed Image Retrieval (CIR) [3, 4, 5] has emerged as a prominent multi-modal retrieval paradigm that combines reference images with retrieval text to retrieve semantically aligned target images. The reference image provides visual details, while the retrieval text specifies the desired modifications (*e.g.*, *change the color*, *remove the logo*). This approach leverages complementary visual and textual strengths, showing significant potential in multimedia analysis, social media, and e-commerce applications. However, CIR operates at the image level, which hinders the understanding of fine-grained objects and the precise localization. As shown in Fig. 1(a), CIR retrieves whole images that often include both matching (*i.e.*, light-colored doughnut) and non-matching objects (*i.e.*, dark-colored doughnut). Consequently, the ambiguity in object-text alignment often requires manual filtering, which increases post-processing time and reduces the scalability of CIR in the real world.

To facilitate object-level retrieval within complex scenes, we propose the **Composed Object Retrieval (COR)** task. As illustrated in Fig. 1(b), COR takes a target image containing various candidate objects, a reference image, a reference object mask, and a retrieval text as input. Given these components, COR retrieves and segments the most relevant object



**Figure 1** Illustration of COR, which retrieves arbitrary target objects from a target image containing candidate objects using composed expressions. It enables precise object-level retrieval, distinguishing target objects (*i.e.*, light-colored doughnuts) from negative ones (*i.e.*, dark-colored doughnut). The retrieval text (*i.e.*, *change the color to light*) specifies attribute changes, allowing flexible retrieval based solely on the reference object and text, without requiring explicit object names (*i.e.*, doughnut), thus supporting effective retrieval even when object categories are difficult to describe.

from the candidate objects that matches the composed query (*i.e.*, retrieval text and reference object) while ignoring unrelated ones. Specifically, each input component plays a distinct role. The reference mask specifies the reference object with-

out relying on explicit class names, enabling flexible retrieval even when object categories are ambiguous or hard to describe. Meanwhile, the retrieval text describes the desired attribute modifications that distinguish the target object from the reference object. This enables COR to process complex queries that integrate visual and textual information. Unlike traditional CIR that retrieves full images, COR enables precise object-level retrieval and segmentation.

COR is more challenging than CIR, as it requires retrieving precise objects that match complex composed expressions, while carefully excluding similar but incorrect ones in the same scene. Specifically, the task involves three main challenges: **1) Compositional matching.** The model needs to understand both the reference object and the retrieval text together to capture subtle changes in attributes such as color or shape. **2) Negative Object Filtering.** The model needs to distinguish the correct target objects from other visually similar candidates in the same image that do not meet the retrieval information. **3) Multi-object retrieval.** The model must locate and segment one or more instances in the target image that match the composed expression.

To advance COR research, we construct **COR127K**, a large-scale benchmark built automatically using public image resources [6, 7] and large multi-modal models [8]. Spanning 408 categories, it includes 127,166 retrieval triplets (*i.e.*, target object, reference object, retrieval text) across 28,183 images and 35,630 objects, divided into Train, Test-Base, and Test-Novel subsets, with Test-Novel featuring 78 novel categories to evaluate generalization. The annotation pipeline is publicly released to facilitate the construction.

We further present **CORE** (Composed Object REtrieval), a unified end-to-end baseline tailored for COR that incorporates: 1) a Reference Region Embedding (RRE) module extracting region-level features from reference objects; 2) an Adaptive Visual-Textual Interaction (AVTI) module constructing composed representations through dynamic fusion; and 3) a COR-oriented contrastive loss that enhances discriminative power between target and negative objects.

The experimental results demonstrate that CORE establishes a robust benchmark in COR127K, outperforming seven state-of-the-art methods. It achieves a 35% improvement in the Dice score and 37% in the IoU on Test-Base, with improvements 21% and 19% on Test-Novel, respectively, demonstrating superior accuracy and generalization.

In summary, the main contributions are as follows.

- We propose a brand-new Composed Object Retrieval (COR) task, enabling fine-grained object-level retrieval via composed expressions.
- We construct COR127K, a large-scale benchmark containing 127,166 retrieval triplets across 408 categories.
- We present CORE, an end-to-end baseline that integrates reference region modeling, adaptive visual-textual fusion, and contrastive learning.

- Our model achieves state-of-the-art results in both base and novel categories with significant improvements in accuracy, flexibility, and generalization, supported by a comprehensive visualization analysis.

## 2 Related Works

### 2.1 Composed Image Retrieval

Composed Image Retrieval (CIR) enhances retrieval flexibility by combining reference images with text, leveraging visual specifics like color and shape from images while obtaining precise attribute and contextual details from text. Recent CIR works [4, 9, 10, 11] typically involve feature extraction, multi-modal fusion, and alignment with target representations. Early methods [12, 13, 14] used separate encoders for visual and textual modalities, while recent approaches [4, 9, 10] leverage pre-trained vision-language models like CLIP [15] and BLIP [16] as unified backbones. Feature extraction uses either global representations [4, 17, 9] or global-local feature combinations [18]. Fusion strategies include target-guided composition [17], progressive learning [19], and bidirectional dual encoder training [9, 10]. Alignment commonly adopts triplet or contrastive loss with in-batch negatives [4, 9], with recent innovations including diffusion-based augmentation [20] and scaled contrastive learning [11]. However, existing CIR methods are limited to image-level retrieval and lack the precision to locate specific objects or distinguish multiple instances in complex scenes. Additionally, current CIR datasets such as FashionIQ and CIRR do not support object-level retrieval because of the absence of region annotations and pixel-level labels. To this end, we propose the task COR, which extends multi-modal retrieval to the object level through a unified framework that integrates region encoding, vision-language interaction, and contrastive learning.

### 2.2 Vision-Language Pre-training Models

Vision-Language Pretraining Models (VLMs) learn multi-modal representations from large-scale image-text pairs and have demonstrated strong zero-shot transfer across various tasks [15, 21, 16]. Early models like CLIP [15] adopt contrastive learning between image and text embeddings, while BLIP [16] introduces a bidirectional encoder-decoder architecture for enhanced cross-modal understanding. Subsequent works such as ALBEF [21] and BLIP-2 [22] further improve vision-language alignment through momentum distillation and vision-to-language bridging, respectively. Despite these advances, existing VLMs operate primarily at the image level, limiting their ability to perform precise object-level localization and segmentation. To overcome this, we propose a unified framework that combines SigLIP [23] for robust vision-language alignment with SAM [24] for accurate object segmentation. This integration enables end-to-end learning that bridges image-level semantics with pixel-level precision, supporting fine-grained performance in the COR task.

### 3 Composed Object Retrieval

#### 3.1 Task Definition

This paper introduces a novel task called Composed Object Retrieval (COR), which aims to localize one or more specific target objects  $O_{tar}$  in a target image  $I_{tar}$ , based on a composed expression formed by a reference object  $O_{ref}$  in a reference image  $I_{ref}$  and a retrieval text  $T_{ret}$ . The task takes four inputs: a reference image  $I_{ref}$ , a binary mask  $M_{ref}$  that specifies the reference object  $O_{ref}$  within  $I_{ref}$ , a target image  $I_{tar}$ , and a retrieval text  $T_{ret}$  that describes the attribute-level transformation from  $O_{ref}$  to the desired target object  $O_{tar}$ . The output is a binary mask  $M_{tar}$  that accurately localizes  $O_{tar}$  in  $I_{tar}$ . The retrieval text  $T_{ret}$  describes only the changes at the attribute level (*e.g.*, shape, color, spatial relations, *etc.*) from  $O_{ref}$  to  $O_{tar}$ , without explicitly naming the object. This noun-free formulation enables the model to generalize to novel or ambiguous categories beyond a fixed vocabulary, enhancing its flexibility in real-world applications. In practice, we provide the reference mask  $M_{ref}$  to specify the reference object  $O_{ref}$ , which is available through simple user interactions like points, bounding boxes, or click-based segmentation tools. This design simplifies the application of COR in the real world.

##### 3.1.1 Comparison with Other Tasks

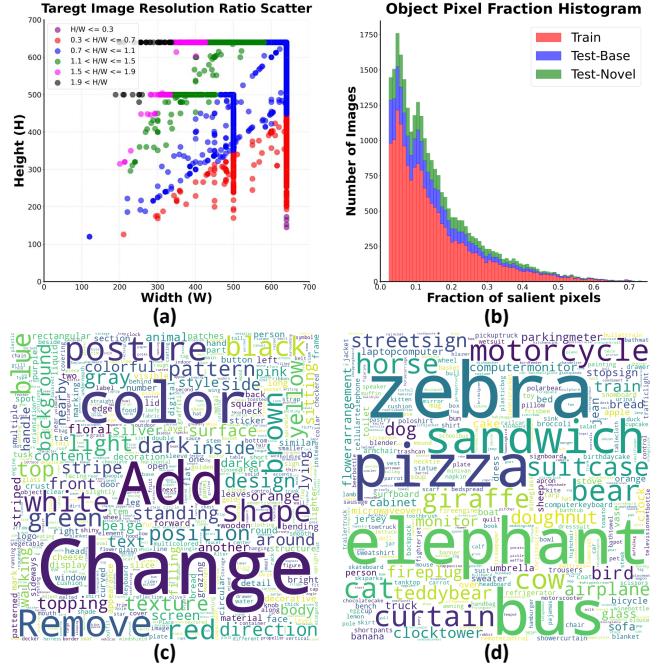
COR fundamentally differs from existing tasks by enabling pixel-accurate object-level localization guided jointly by visual and textual inputs. 1) Compared to CIR methods that focus on image-level matching, COR produces segmentation masks, allowing for fine-grained localization of the specific object of interest within the image. 2) Unlike text-only retrieval methods such as refer segmentation that rely solely on language, COR uses reference images and masks to visually ground target objects, reducing textual ambiguity and enabling more accurate retrieval for cases involving ambiguous categories or subtle visual differences.

#### 3.2 COR127K Dataset

We present COR127K, a large-scale dataset for the COR, built from COCO2017 [6] and LVIS [7] images and annotations. Through a ten-step automated pipeline, we generate 127,166 high-quality retrieval triplets (each consisting of a target object  $O_{tar}$ , a reference object  $O_{ref}$ , and a retrieval text  $T_{ret}$  describing their attribute-level transformation). Each object is precisely localized by a binary mask:  $M_{ref}$  defines  $O_{ref}$  in the reference image  $I_{ref}$ , and  $M_{tar}$  defines  $O_{tar}$  in the target image  $I_{tar}$ . Thus, each pair contains five elements:  $(I_{ref}, M_{ref}, T_{ret}, I_{tar}, M_{tar})$ . Retrieval texts are generated using Qwen-VL 72B [8], ensuring fine-grained and compositional descriptions. In general, COR127K includes 28,183 images, 35,630 objects, and 408 categories, forming a diverse and challenging benchmark for object-level retrieval. We

Metric	All	Train	Test-Base	Test-Novel
total pairs	127,166	85,928	23,337	17,901
total categories	408	330	284	78
target images $I_{tar}$	21,434	14,861	4,371	3,735
target objects $O_{tar}$	26,576	17,783	4,921	4,158
reference images $I_{ref}$	16,533	12,851	6,982	3,308
reference objects $O_{ref}$	18,406	14,031	7,278	3,378
all images $I_{all}$	28,183	20,689	11,010	5,125
all objects $O_{all}$	35,630	24,975	11,949	5,637

**Table 1** Statistics for the COR127K dataset and its subsets.



**Figure 2** Statistics of the COR127K dataset: (a) image-resolution distribution; (b) object-to-image area ratio; (c) retrieval text word-cloud; (d) category word-cloud.

divide COR127K into Train, Test-Base (330 base categories) and Test-Novel (78 novel categories) to evaluate the generalization. The dataset supports flexible retrieval in both single- and multi-object scenarios and introduces visually similar distractors to increase difficulty and encourage fine-grained discrimination.

##### 3.2.1 COR127K Dataset Details.

Tab. 1 summarizes the pair statistics of COR127K and its three subsets: Train, Test-Base, and Test-Novel. Fig. 2(a)–(d) provide further insights into the dataset, including the distribution of target-to-image resolution ratios, object-to-image area ratios, and word clouds for both retrieval texts and object categories. In Fig. 2(a), different colors represent resolution-ratio intervals, highlighting the scale diversity. In (b), colors indicate subsets for direct comparison of area-ratio distributions. In (c) and (d), word sizes reflect frequency, illustrating

common text expressions and dominant categories. Detailed class distributions across subsets are included in Appendix A. The Train set comprises 85,928 triplets across 330 base categories, with 17,783 unique target objects from 14,861 images and 14,031 reference objects from 12,851 images. The Test Base set includes 23,337 triplets over 284 base categories, involving 4,921 target objects from 4,371 images and 7,278 references from 6,982 images. The Test-Novel set contains 17,901 triplets spanning 78 novel categories, with 4,158 target objects from 3,735 images and 3,378 references from 3,308 images.

### 3.2.2 Automated Annotation Generation.

To construct the large-scale dataset COR127K, we developed a fully automated pipeline that uses COCO2017, LVIS annotations, and the Qwen2.5-VL model. The process covers image filtering, object pairing, text generation, and strict quality control, resulting in 127,166 diverse and semantically precise retrieval triplets. The pipeline is structured into four stages and ten steps: **Stage 1: Raw Data Preprocessing**, which involves *Step 1: candidate object selection* and *Step 2: low-quality object removal*; **Stage 2: Data Split**, where categories are divided into *Step 3: base/novel sets* and then split into *Step 4: training/testing sets*; **Stage 3: Retrieval Triplet Building**, encompassing *Step 5: reference selection*, *Step 6: target selection*, *Step 7: pair construction*, and *Step 8: retrieval text generation*; and finally, **Stage 4: Triplet Validation**, which includes *Step 9: retrieval verification* and *Step 10: false match rejection*. More detailed data annotation information, including the specific content of each step and the prompts used for data generation, is provided in Appendix B.

## 4 Approach

### 4.1 Limitations of Existing CIR Methods

Current CIR approaches are inadequate for the COR due to two main limitations: 1) they can only retrieve entire images rather than directly localizing specific objects within images; 2) they extract features at the global image level rather than focusing on the specified reference object, leading to suboptimal representations with irrelevant information.

Although a multistage pipeline combining CIR with an object detection and segmentation model could perform COR, this solution has three major drawbacks: 1) it is not end-to-end trainable, 2) it requires high computational costs, and 3) it lacks support for multi-object retrieval.

### 4.2 Overall Architecture

We propose CORE (Composed Object REtrieval), an end-to-end baseline model for COR, as illustrated in Fig. 3. CORE integrates three core designs: 1) a Reference Region Embedding (RRE) module that extracts object-level features from reference images; 2) an Adaptive Vision-Text Interaction

(AVTI) module that dynamically fuses reference visual features with retrieval text to produce discriminative composed representations; and 3) a COR oriented contrastive loss  $\mathcal{L}_{cor}$  that aligns target objects with their references while suppressing background and distractor.

The framework processes target image  $I_{tar}$ , reference image  $I_{ref}$ , reference object mask  $O_{ref}$ , and retrieval text  $T_{ret}$  through the SAM image encoder, VLM vision encoder, mask encoder, and VLM text encoder, respectively, generating features  $F_{tar}$ ,  $F_{ref}$ ,  $F_{mask}$ , and  $F_{txt}$ . The RRE module combines  $F_{ref}$  and  $F_{mask}$  to produce the representation  $F_{rre}$ . The AVTI module then fuses  $F_{rre}$  with  $F_{txt}$  to generate the composed representation  $F_{avti}$ , which guides the SAM mask decoder to process  $F_{tar}$  for the final prediction.

### 4.3 Reference Region Embedding (RRE)

We utilize a mask to highlight the object of interest. This strategy does not rely on class names, which makes it effective even when certain categories are difficult to define or require expert level knowledge. Instead of the commonly used mask cropping [25, 26] or mask pooling [27] strategies, our RRE module learns to fuse mask features with image features, thereby preserving semantic context and avoiding disruptions to the image distribution. Inspired by [28], the RRE module employs a semantic activation-based strategy that computes activation maps from the features of the reference image and the masks of objects, highlighting regions relevant to identity while preserving contextual information. Concretely, we utilize a Mask Encoder consisting of two convolutional layers to extract the mask feature  $F_{mask}$  from the reference mask  $M_{ref}$ :

$$F_{mask} = \text{MaskEncoder}(M_{ref}). \quad (1)$$

The RRE module takes the reference mask feature  $F_{mask}$  and reference image features  $F_{ref}$  as inputs. The RRE module combines  $F_{mask}$  with reference image features  $F_{ref}$ , processing their sum through three stacked Semantic Feature Enhancing (SFE) blocks to enrich semantics:

$$\text{SFE}(x) = x + \text{PWC}(\text{GeLU}(\text{PWC}(\text{LN}(\text{DWC}(x))))), \quad (2)$$

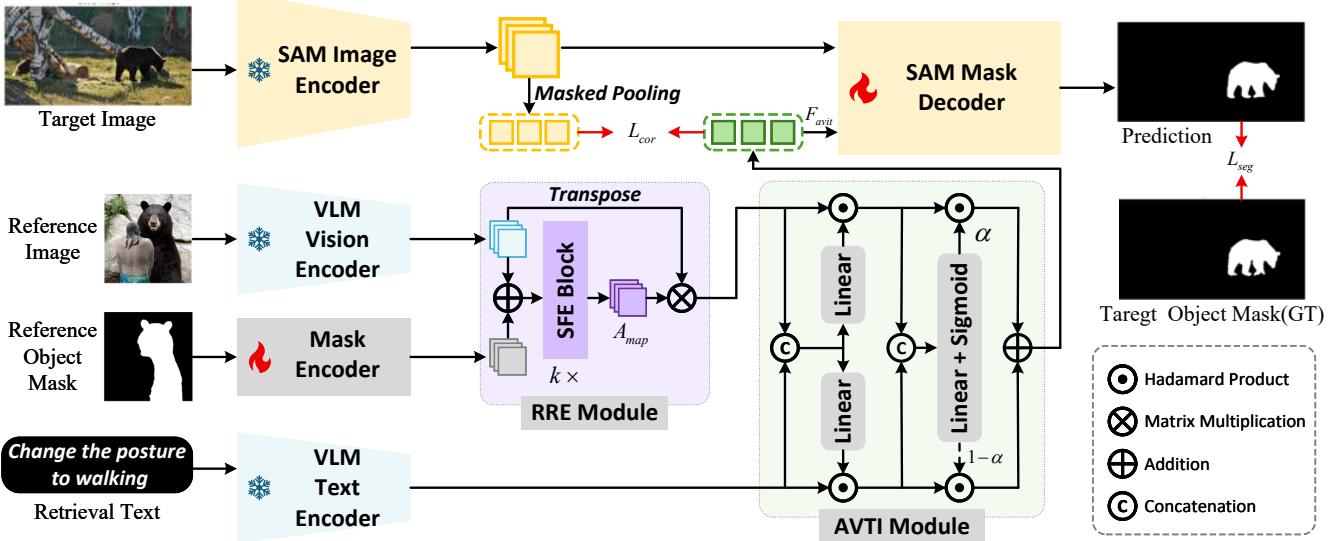
where  $x = F_{mask} + F_{ref}$ , DWC is a  $7 \times 7$  depth-wise convolution, PWC is a  $1 \times 1$  point-wise convolution, LN is layer normalization. The semantic activation map  $A_{map} \in \mathbb{R}^{h \times w \times k}$  is computed as:

$$A_{map} = \text{Conv}_{1 \times 1}(\text{SFE}_3(F_{ref} + F_{mask})), \quad (3)$$

where semantic activation maps  $A_{map} \in \mathbb{R}^{h \times w \times k}$ , corresponding to  $k$  semantic subspace.

The  $F_{rre}$  is obtained by aggregating semantic-aware features via batch matrix multiplication between spatially flattened  $F_{ref}$  and normalized activation maps  $\bar{A}_{map}$ :

$$F_{rre} = \frac{1}{K} \sum_{k=1}^K \bar{A}_{map}^k \cdot F_{ref}^T, \quad (4)$$



**Figure 3** Architecture of our proposed model (CORE), which comprises three key components: the Reference Region Embedding (RRE) module, the Adaptive Vision-Text Interaction (AVTI) module, and a COR-oriented contrastive loss  $\mathcal{L}_{cor}$ .

where  $\bar{A}_{map}^k$  is the  $k$ -th normalized activation map. Averaging across  $K$  subspaces yields  $F_{rre}$ , capturing various semantic cues for robust reference object representation.

#### 4.4 Adaptive Vision-Text Interaction (AVTI)

The AVTI module, inspired by [4, 5], enhances the COR task by adaptively fusing reference object features with retrieval text to produce semantically rich embeddings. Given reference object features  $F_{rre} \in \mathbb{R}^d$  and retrieval text features  $F_{txt} \in \mathbb{R}^d$ , the module concatenates them into  $F_{comb} = [F_{rre}, F_{txt}] \in \mathbb{R}^{2d}$ . Modality-specific attention weights are computed via:

$$attn_V = \sigma(\text{Linear}_{v2}(\text{ReLU}(\text{Linear}_{v1}(F_{comb})))), \quad (5)$$

$$attn_T = \sigma(\text{Linear}_{t2}(\text{ReLU}(\text{Linear}_{t1}(F_{comb})))), \quad (6)$$

where  $\sigma(\cdot)$  is the sigmoid function. The attended features  $attn_V \cdot F_{rre}$  and  $attn_T \cdot F_{txt}$  are concatenated and processed for a scalar weight  $\alpha \in [0, 1]$ :

$$\alpha = \sigma(\text{Linear}(\text{ReLU}(\text{Linear}([attn_V \cdot F_{rre}, attn_T \cdot F_{txt}])))). \quad (7)$$

The composed feature  $F_{avit}$  is then generated as:

$$F_{avit} = \alpha \cdot attn_V \cdot F_{rre} + (1 - \alpha) \cdot attn_T \cdot F_{txt}, \quad (8)$$

which serves as the sparse prompt for the SAM Mask Decoder, guiding the decoding of  $F_{tar}$ . The decoder leverages  $F_{avit}$  to focus on the target object, producing the final segmentation prediction  $Pred = \text{MaskDecoder}(F_{tar}, F_{avit})$ , which accurately delineates the target object while suppressing background and distractor interference.

#### 4.5 Loss Function

To enhance target object distinction from background clutter, we propose a COR-oriented contrastive loss  $\mathcal{L}_{cor}$ , combining

foreground alignment and background repulsion to align the target region and suppress non-target areas. The foreground alignment term ensures semantic consistency:

$$\mathcal{L}_{fg} = 1 - \frac{1}{V} \sum_{i=1}^V \text{CosSim}(F_{tar}^{fg}, F_{avit}), \quad (9)$$

where  $F_{tar}^{fg} = \text{MaskedPooling}(F_{tar}, gt)$  is the masked average-pooled foreground feature, and  $\text{CosSim}(a, b) = \frac{a \cdot b}{\|a\| \|b\|}$  denotes cosine similarity. The background repulsion term minimizes similarity with distractors:

$$\mathcal{L}_{bg} = 1 + \frac{1}{V} \sum_{i=1}^V \text{CosSim}(F_{tar}^{bg}, F_{avit}), \quad (10)$$

where  $F_{tar}^{bg} = \text{MaskedPooling}(F_{tar}, 1 - gt)$  represents the background feature, reducing distractor influence. The COR-oriented contrastive loss is  $\mathcal{L}_{cor} = \mathcal{L}_{fg} + \mathcal{L}_{bg}$ .

The final training objective integrates the contrastive and segmentation losses is  $\mathcal{L}_{total} = \mathcal{L}_{seg} + \mathcal{L}_{cor}$ , with  $\mathcal{L}_{wbe}$  and  $\mathcal{L}_{wiou}$  denoting edge-aware weighted binary cross-entropy and IoU losses, respectively.

## 5 Experiments

### 5.1 Experimental Setup

*Implementation Details.* We evaluate CORE on COR127K, tailored for the COR, comprising three splits: Train, Test-Base, and Test-Novel. We use standard segmentation metrics: Dice, IoU, mDice, mIoU, and MAE. Higher Dice and IoU scores indicate superior performance, while lower MAE reflects precise prediction. Our model builds on SAM-Base [24] for segmentation and SigLIP-Base [23] as the VLM’s vision

Method	COR127K-Test-Base					COR127K-Test-Novel				
	Dice $\uparrow$	IoU $\uparrow$	MAE $\downarrow$	mDice $\uparrow$	mIoU $\uparrow$	Dice $\uparrow$	IoU $\uparrow$	MAE $\downarrow$	mDice $\uparrow$	mIoU $\uparrow$
CLIP4CIR	0.5333	0.4771	0.1166	0.7292	0.6759	0.5420	0.4903	0.1149	0.7347	0.6842
BLIP4CIR	0.5146	0.4570	0.1251	0.7174	0.6618	0.5032	0.4462	0.1306	0.7107	0.6545
BLIP24CIR	0.5157	0.4585	0.1180	0.7203	0.6660	0.5097	0.4546	0.1179	0.7184	0.6649
Bi-BLIP4CIR	0.5308	0.4729	0.1231	0.7258	0.6706	0.5490	0.4916	0.1207	0.7364	0.6818
CLIP4CIR-SPN	0.5474	0.4895	0.1137	0.7371	0.6835	0.5545	0.5004	0.1120	0.7419	0.6905
BLIP4CIR-SPN	0.5277	0.4692	0.1232	0.7243	0.6687	0.5222	0.4644	0.1270	0.7211	0.6652
BLIP24CIR-SPN	0.5709	0.5094	0.1098	0.7501	0.6952	0.5883	0.5285	0.1044	0.7612	0.7081
<b>CORE (Ours)</b>	<b>0.7703</b> <sub>+35%</sub>	<b>0.6955</b> <sub>+37%</sub>	<b>0.0741</b> <sub>-33%</sub>	<b>0.8603</b> <sub>+15%</sub>	<b>0.8044</b> <sub>+16%</sub>	<b>0.7102</b> <sub>+21%</sub>	<b>0.6290</b> <sub>+19%</sub>	<b>0.0858</b> <sub>-18%</sub>	<b>0.8276</b> <sub>+9%</sub>	<b>0.7652</b> <sub>+8%</sub>

**Table 2** Quantitative results. Bold: best, underline: second-best. Percentage: improvement over second-best.

Method	COR127K-Test-Base							COR127K-Test-Novel						
	All	1p0n	1p1n	1p2n	2p0n	2p1n	3p0n	All	1p0n	1p1n	1p2n	2p0n	2p1n	3p0n
CLIP4CIR	0.5333	0.6637	0.4828	0.5209	0.4428	0.3035	0.3802	0.5420	0.6447	0.4595	0.4990	0.3371	0.2494	0.2889
BLIP4CIR	0.5146	0.6359	0.4732	0.5001	0.4324	0.2908	0.3511	0.5032	0.5825	0.4505	0.4455	0.3441	0.2577	0.2810
BLIP24CIR	0.5157	0.6351	0.4770	0.4978	0.4230	0.3014	0.3824	0.5097	0.6162	0.4029	0.4382	0.3320	0.2383	0.2301
Bi-BLIP4CIR	0.5308	0.6622	0.4855	0.5023	0.4361	0.3061	0.3666	0.5490	0.6364	0.4796	0.5377	0.3688	0.2937	0.3026
CLIP4CIR-SPN	0.5474	0.6738	0.5034	0.5439	0.4514	0.3223	0.3898	0.5545	0.6525	0.4817	0.5224	0.3522	0.2603	0.2913
BLIP4CIR-SPN	0.5277	0.6547	0.4828	0.5271	0.4354	0.2958	0.3603	0.5222	0.6047	0.4670	0.4735	0.3529	0.2671	0.2909
BLIP24CIR-SPN	0.5709	0.6983	0.5454	0.5531	0.4601	0.3295	0.3961	0.5883	0.6969	0.4846	0.5443	0.3968	0.2847	0.2758
<b>CORE (Ours)</b>	<b>0.7703</b>	<b>0.8644</b>	<b>0.6335</b>	<b>0.7165</b>	<b>0.8465</b>	<b>0.6210</b>	<b>0.7744</b>	<b>0.7102</b>	<b>0.8120</b>	<b>0.5317</b>	<b>0.6840</b>	<b>0.6075</b>	<b>0.4977</b>	<b>0.6170</b>

**Table 3** Comparison across different retrieval settings on the Test dataset. The  $xpyn$  configuration denotes setups where  $x$  represents the number of positive objects and  $y$  represents the number of negative objects. Bold: best, underline: second-best.

and text encoders for visual-textual representation learning, with both backbones frozen during training. All other parameters are trainable. The model is fine-tuned for 15 epochs using the AdamW optimizer with a learning rate of  $1 \times 10^{-4}$ , using input resolutions of  $1024 \times 1024$  for SAM and  $384 \times 384$  for SigLIP. Training on 4 RTX 4090 GPUs with BF16 precision and a per-GPU batch size of 6.

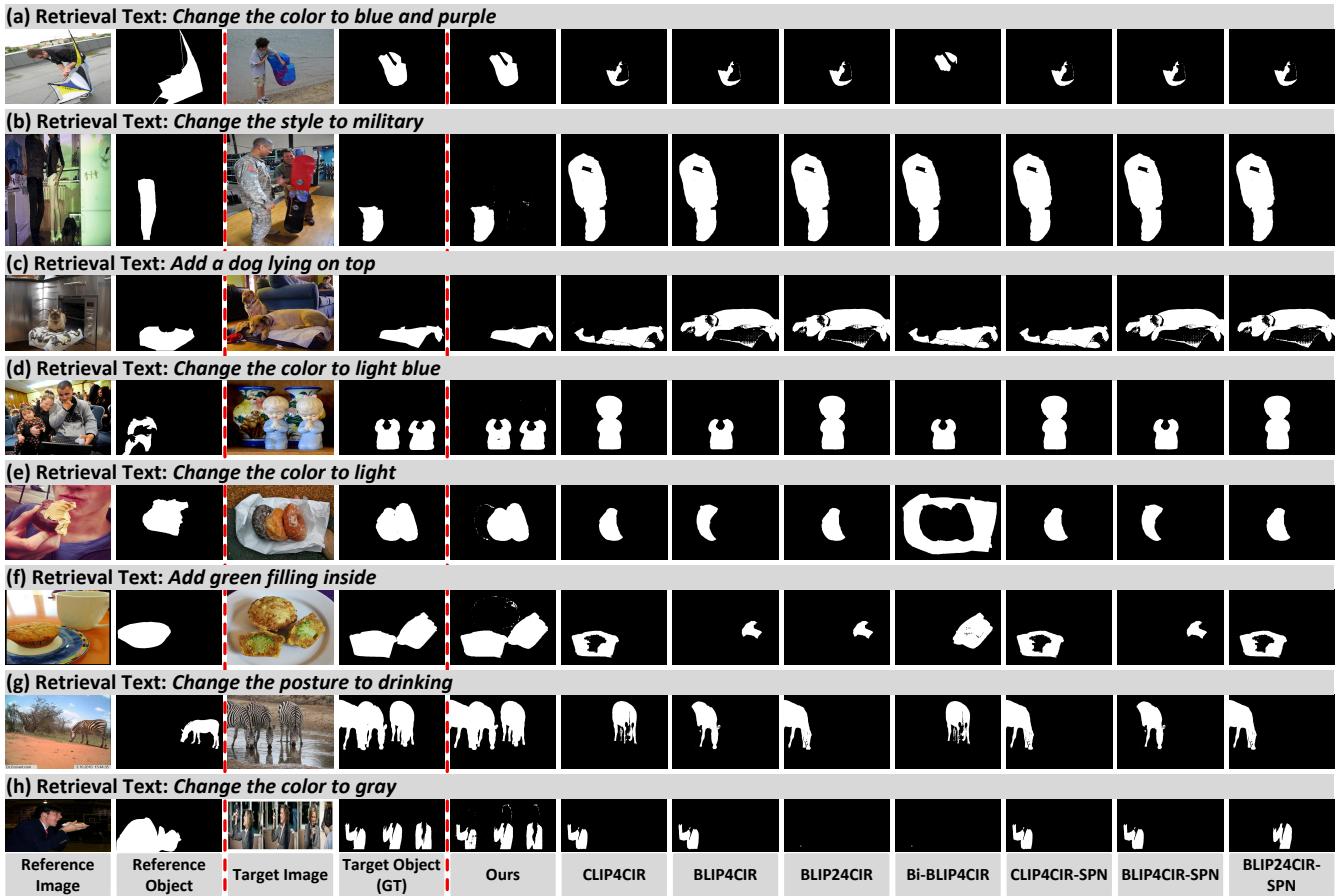
**Comparable Methods.** We construct comparable methods by implementing a modular paradigm for COR, composed of **detection model + CIR model + segmentation model**. Specifically, we adopt Detic [29] as the detection model, which is pre-trained on LVIS dataset [7]. For segmentation model, we employ SAM [24], which is capable of generating precise masks from bounding boxes. As the retrieval model, we integrate existing CIR models that compute feature similarity between a reference input and candidate regions. The baseline pipeline proceeds as follows: 1) Detic identifies up to 30 candidate objects in the target image (confidence  $> 0.3$ , NMS threshold  $< 0.8$ ); 2) Candidate regions are extracted as cropped patches using detected bounding boxes; 3) CIR models compute feature similarity between the reference object and each candidate, selecting the most similar region; 4) The selected bounding box and target image are fed into SAM to produce the final object mask. This modular pipeline provides a competitive baseline for evaluating CORE on the COR task.

## 5.2 Quantitative Results

We evaluate CORE on the COR127K dataset with seven strong CIR baselines for comparison: CLIP4CIR [4], BLIP4CIR [9], BLIP24CIR [10], CLIP4CIR-SPN [11], BLIP4CIR-SPN [11], BLIP24CIR-SPN [11], and Bi-BLIP4CIR-Sum [30]. All baselines use open-source pre-trained weights from the CIRR dataset [3], enabling direct evaluation. Results are summarized in Tab. 2.

On Test-Base, CORE surpasses the strongest baseline, BLIP24CIR-SPN, achieving substantial improvements of 35.0% in Dice, 36.5% in IoU, 14.7% in mDice, and 15.7% in mIoU. On Test-Novel, it records gains of 20.7%, 19.0%, 8.7%, and 8.1%, respectively. These improvements stem from CORE’s unified end-to-end design, which seamlessly integrates reference feature encoding, adaptive vision-language interaction, and region-level contrastive learning.

We further evaluate performance across retrieval settings in Tab. 3. The  $xpyn$  setting denotes retrieval with  $x$  positive objects and  $y$  negative objects. On Test-Base, CORE achieves significant Dice improvements: 1p0n (+23.8%), 1p1n (+16.2%), 1p2n (+29.5%), 2p0n (+83.9%), 2p1n (+88.5%), and 3p0n (+95.5%). On Test-Novel, gains are consistent: 1p0n (+16.5%), 1p1n (+9.7%), 1p2n (+25.7%), 2p0n (+53.1%), 2p1n (+74.8%), and 3p0n (+123.7%). Notably, negative object interference (*e.g.*, 1p1n, 1p2n, 2p1n) increases retrieval complexity, degrading performance across all models. Settings with three or more positive objects (*e.g.*, 3p0n) are particularly challenging due to heightened semantic



**Figure 4** Qualitative comparisons of our CORE with other state-of-the-art methods. Zoom in for detailed information.

ambiguity, yet CORE maintains robust performance, demonstrating its strength in multi-object retrieval and negative object discrimination.

### 5.3 Qualitative Results

We compare CORE with state-of-the-art CIR methods across eight examples (a)–(h) in Fig. 4, highlighting three key advantages. First, in examples **a**, **b**, and **c**, CORE effectively retrieves objects that are difficult to describe or belong to ambiguous categories, demonstrating strong semantic understanding beyond explicit labels. Second, in **d**, **e**, **f**, **g**, and **h**, it accurately retrieves multiple target objects, showcasing reliable multi-instance retrieval. Third, in **b**, **c**, **e**, and **f**, CORE successfully filters out negative objects that are visually similar but semantically incorrect, indicating strong robustness against distractors. We have supplemented more results in the Appendix C.1. These collectively demonstrate CORE’s superior performance in complex retrieval scenarios.

### 5.4 Ablation Studies

We perform ablation studies on three aspects of CORE: network modules, loss functions, and pre-trained model scaling. The experiments follow the settings in Sec. 5, with results

summarized in Tab. 4. We have also supplemented the ablation experiments in Appendix C.2, where the reference information is from different constituents.

*Network Modules and Loss Functions.* Removing any of the RRE, AVTI modules or the contrastive loss  $\mathcal{L}_{cor}$  leads to performance drops. Specifically, excluding RRE (replaced with masked pooling) causes a Dice drop of 0.84% on Test-Base and 2.66% on Test-Novel. Excluding the AVTI (using a simple sum operation for feature fusion) leads to a 5.85% Dice drop on Test-Novel. Omitting  $\mathcal{L}_{cor}$  causes the largest decline, with a 7.05% Dice reduction on Test-Novel. These results show that RRE strengthens the representation of the reference object by separating it from the background; the AVTI enhances vision-language alignment, enabling more reliable composed expressions; and  $\mathcal{L}_{cor}$  improves the alignment between the reference and target object representations while suppressing negative objects.

*Pre-trained Model Scaling.* We evaluate scaling the SigLIP and SAM by replacing with larger counterparts, keeping encoders frozen during 15-epoch training at a learning rate of  $1 \times 10^{-4}$ . Using SAM-Large with SigLIP-Base improves

Setting	RRE	AVTI	$\mathcal{L}_{cor}$	COR127K-Test-Base					COR127K-Test-Novel				
				Dice $\uparrow$	IoU $\uparrow$	MAE $\downarrow$	mDice $\uparrow$	mIoU $\uparrow$	Dice $\uparrow$	IoU $\uparrow$	MAE $\downarrow$	mDice $\uparrow$	mIoU $\uparrow$
w/o RRE	$\times$	$\checkmark$	$\checkmark$	0.7638	0.6848	0.0766	0.8561	0.7972	0.6913	0.6067	0.0976	0.8140	0.7477
w/o AVTI	$\checkmark$	$\times$	$\checkmark$	0.7541	0.6762	0.0808	0.8499	0.7909	0.6686	0.5877	0.1055	0.8005	0.7346
w/o $\mathcal{L}_{cor}$	$\checkmark$	$\checkmark$	$\times$	0.7533	0.6768	0.0784	0.8504	0.7925	0.6601	0.5791	0.1041	0.7967	0.7311
<b>CORE (Ours)</b>	$\checkmark$	$\checkmark$	$\checkmark$	<b>0.7703</b>	<b>0.6955</b>	<b>0.0741</b>	<b>0.8603</b>	<b>0.8044</b>	<b>0.7102</b>	<b>0.6290</b>	<b>0.0858</b>	<b>0.8276</b>	<b>0.7652</b>
SigLIP (B) + SAM (L)	$\checkmark$	$\checkmark$	$\checkmark$	0.7784	0.7118	0.0692	0.8660	0.8158	0.7008	0.6338	0.0882	0.8223	0.7677
SigLIP (L) + SAM (B)	$\checkmark$	$\checkmark$	$\checkmark$	0.7741	0.6977	0.0729	0.8625	0.8060	0.6828	0.5990	0.0989	0.8097	0.7436
SigLIP (L) + SAM (L)	$\checkmark$	$\checkmark$	$\checkmark$	0.7793	0.7127	0.0682	0.8670	0.8169	0.6938	0.6261	0.0917	0.8175	0.7620

**Table 4** Ablation study on the key components and model scaling.  $\checkmark$ : enabled;  $\times$ : disabled.

Dice by 1.05% on Test-Base but reduces it by 1.32% on Test-Novel, with similar trends for SigLIP-Large. Larger pre-trained models enhance in-domain performance but risk overfitting, hindering cross-domain generalization. The SigLIP-Base + SAM-Base configuration optimally balances accuracy, generalization, and efficiency.

## 6 Conclusion

We introduce Composed Object Retrieval (COR), a novel task that extends multi-modal retrieval from image-level to object-level with composed expressions. We present COR127K, a large-scale dataset with 127,166 retrieval triplets in 408 categories. We also present CORE, an end-to-end framework that integrates reference region encoding, adaptive vision-language interaction, and region-level contrastive learning. Experiments demonstrate that CORE surpasses existing methods in both base and novel categories, excelling in fine-grained object retrieval. COR enables applications in fine-grained visual search, content analysis, and advanced image understanding, paving the way for sophisticated object-level multi-modal retrieval systems.

## References

- [1] Sanghyuk Chun, Seong Joon Oh, Rafael Sampaio De Rezende, Yannis Kalantidis, and Diane Larlus. Probabilistic embeddings for cross-modal retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8415–8424, 2021. [1](#)
- [2] Shiv Ram Dubey. A decade survey of content based image retrieval using deep learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(5):2687–2704, 2021. [1](#)
- [3] Zheyuan Liu, Cristian Rodriguez-Opazo, Damien Teney, and Stephen Gould. Image retrieval on real-life images with pre-trained vision-and-language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2125–2134, 2021. [1, 6](#)
- [4] Alberto Baldrati, Marco Bertini, Tiberio Uricchio, and Alberto Del Bimbo. Composed image retrieval using contrastive learning and task-oriented clip-based features. *ACM Transactions on Multimedia Computing, Communications and Applications*, 20(3):1–24, 2023. [1, 2, 5, 6](#)
- [5] Fuxiang Huang, Lei Zhang, Xiaowei Fu, and Suqi Song. Dynamic weighted combiner for mixed-modal image retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 2303–2311, 2024. [1, 5](#)
- [6] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. [2, 3](#)
- [7] Agrim Gupta, Piotr Dollar, and Ross Girshick. LVIS: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019. [2, 3, 6](#)
- [8] 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, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025. [2, 3](#)
- [9] Zheyuan Liu, Weixuan Sun, Damien Teney, and Stephen Gould. Candidate set re-ranking for composed image retrieval with dual multi-modal encoder. *arXiv preprint arXiv:2305.16304*, 2023. [2, 6](#)
- [10] Xinxing Xu, Yong Liu, Salman Khan, Fahad Khan, Wangmeng Zuo, Rick Siow Mong Goh, Chun-Mei Feng, et al. Sentence-level prompts benefit composed image retrieval. In *The International Conference on Learning Representations*, 2024. [2, 6](#)
- [11] Zhangchi Feng, Richong Zhang, and Zhijie Nie. Improving composed image retrieval via contrastive learning with scaling positives and negatives. In *Proceedings of the ACM International Conference on Multimedia*, pages 1632–1641, 2024. [2, 6](#)
- [12] Eric Dodds, Jack Culpepper, Simao Herdade, Yang Zhang, and Kofi Boakye. Modality-agnostic attention fusion for visual search with text feedback. *arXiv preprint arXiv:2007.00145*, 2020. [2](#)
- [13] Jongseok Kim, Youngjae Yu, Hoeseong Kim, and Gunhee Kim. Dual compositional learning in interactive image retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1771–1779, 2021. [2](#)

- [14] Nam Vo, Lu Jiang, Chen Sun, Kevin Murphy, Li-Jia Li, Li Fei-Fei, and James Hays. 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, 2019. 2
- [15] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 2
- [16] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022. 2
- [17] Haokun Wen, Xian Zhang, Xuemeng Song, Yinwei Wei, and Liqiang Nie. Target-guided composed image retrieval. In *Proceedings of the ACM International Conference on Multimedia*, pages 915–923, 2023. 2
- [18] Matan Levy, Rami Ben-Ari, Nir Darshan, and Dani Lischinski. Data roaming and quality assessment for composed image retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 2991–2999, 2024. 2
- [19] Yida Zhao, Yuqing Song, and Qin Jin. Progressive learning for image retrieval with hybrid-modality queries. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1012–1021, 2022. 2
- [20] Geonmo Gu, Sanghyuk Chun, Wonjae Kim, HeeJae Jun, Yoohoon Kang, and Sangdoo Yun. Compodiff: Versatile composed image retrieval with latent diffusion. *arXiv preprint arXiv:2303.11916*, 2023. 2
- [21] Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021. 2
- [22] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 2
- [23] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023. 2, 5
- [24] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023. 2, 5, 6
- [25] Jian Ding, Nan Xue, Gui-Song Xia, and Dengxin Dai. Decoupling zero-shot semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11583–11592, 2022. 4
- [26] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7061–7070, 2023. 4
- [27] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *European conference on computer vision*, pages 540–557. Springer, 2022. 4
- [28] Yongkang Li, Tianheng Cheng, Bin Feng, Wenyu Liu, and Xinggang Wang. Mask-adapter: The devil is in the masks for open-vocabulary segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14998–15008, 2025. 4
- [29] Xingyi Zhou, Rohit Girdhar, Armand Joulin, Philipp Krähenbühl, and Ishan Misra. Detecting twenty-thousand classes using image-level supervision. In *European conference on computer vision*, 2022. 6
- [30] Zheyuan Liu, Weixuan Sun, Yicong Hong, Damien Teney, and Stephen Gould. Bi-directional training for composed image retrieval via text prompt learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5753–5762, 2024. 6

## Supplementary Material

### A Category Distribution of COR127K

The COR127K dataset comprises 127,166 retrieval triplets spanning 408 diverse object categories, systematically partitioned into three distinct subsets: Train, Test-Base, and Test-Novel. Each triplet represents a fundamental unit of our retrieval task, consisting of a target object  $O_{tar}$ , a reference object  $O_{ref}$ , and a retrieval text  $T_{ret}$  that precisely describes the attribute-level transformation between them. This comprehensive structure enables the evaluation of fine-grained object retrieval capabilities across various semantic relationships and attribute modifications.

The category-wise distribution of retrieval triplet counts across all 408 categories is illustrated in Fig. 5, where the horizontal axis represents category names and the vertical axis indicates the number of retrieval triplets for each corresponding category. In this visualization, red, blue, and green bars represent the Train, Test-Base, and Test-Novel sets, respectively, providing a clear overview of data distribution.

### B Automated Annotation Pipeline for COR127K

#### B.1 Overall Pipeline

To construct COR127K, a large-scale and high-quality composed object retrieval dataset specifically designed for fine-grained object-level retrieval tasks, we developed a comprehensive and fully automated pipeline that leverages state-of-the-art resources including COCO2017 images, LVIS annotations, and the advanced QWen2.5-VL multimodal language model. This sophisticated pipeline systematically integrates multiple critical components: intelligent image filtering to select suitable candidate images, strategic sample pairing to establish meaningful object relationships, automated text generation to create descriptive retrieval queries, and rigorous multi-stage data quality control to ensure semantic precision and consistency.

The entire pipeline is strategically organized into **four stages** that logically aggregate **ten steps**, creating a coherent workflow from raw data preprocessing to final dataset validation, as comprehensively illustrated in Fig. 6 (left). Through this meticulous process, we successfully generated 127,166 diverse and semantically precise retrieval triplets spanning 408 object categories, establishing a robust foundation for advancing composed object retrieval research.

**Stage 1 (Raw Data Preprocessing)** serves as the foundation of our pipeline, incorporating *Step 1 (Candidate Object Selection)* and *Step 2 (Low-Quality Object Removal)* to establish a comprehensive and clean data foundation for all subsequent processing stages, ensuring that only high-quality objects are retained for dataset construction.

Building upon this preprocessed data, **Stage 2 (Data Split)**

systematically partitions the cleaned dataset through *Step 3 (Base/Novel Category Split)* and *Step 4 (Train/Test Split)* to establish rigorous evaluation protocols that allow for proper evaluation of model generalization capabilities across both seen and unseen categories.

With the data properly organized, **Stage 3 (Triplet Building)** forms the core component of our pipeline by meticulously implementing *Step 5 (Reference Object Selection)*, *Step 6 (Target Object Selection)*, *Step 7 (Pair Construction)*, and *Step 8 (Retrieval Text Generation)* to create semantically meaningful object relationships and generate comprehensive retrieval data that capture diverse attribute transformations.

Finally, **Stage 4 (Triplets Validation)** ensures the overall quality and reliability of the constructed triplets through a comprehensive two-stage quality control process, implementing *Step 9 (Retrieval Verification)* and *Step 10 (False Match Rejection)* to systematically identify and eliminate unqualified data entries, thus guaranteeing the semantic accuracy and consistency of the final dataset.

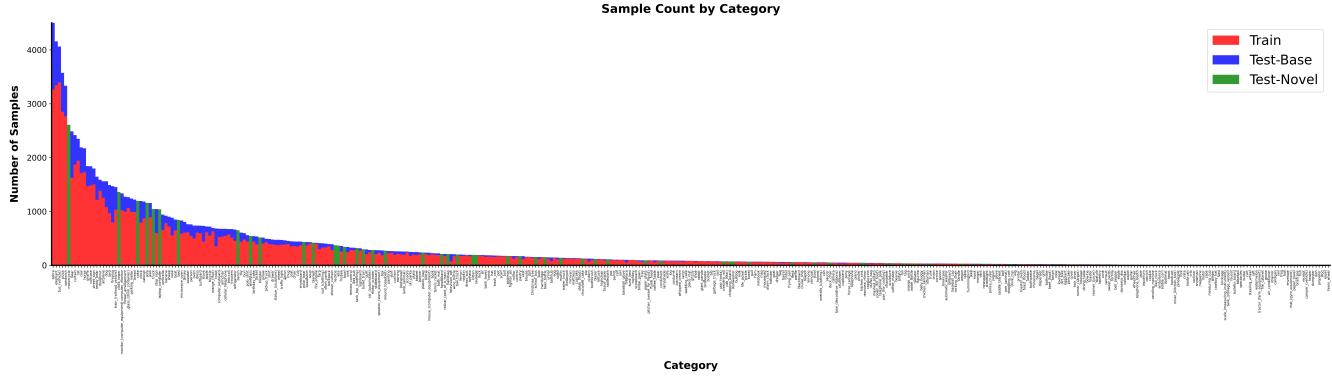
#### B.2 Details of Annotation Generation

The detailed implementation process is as follows.

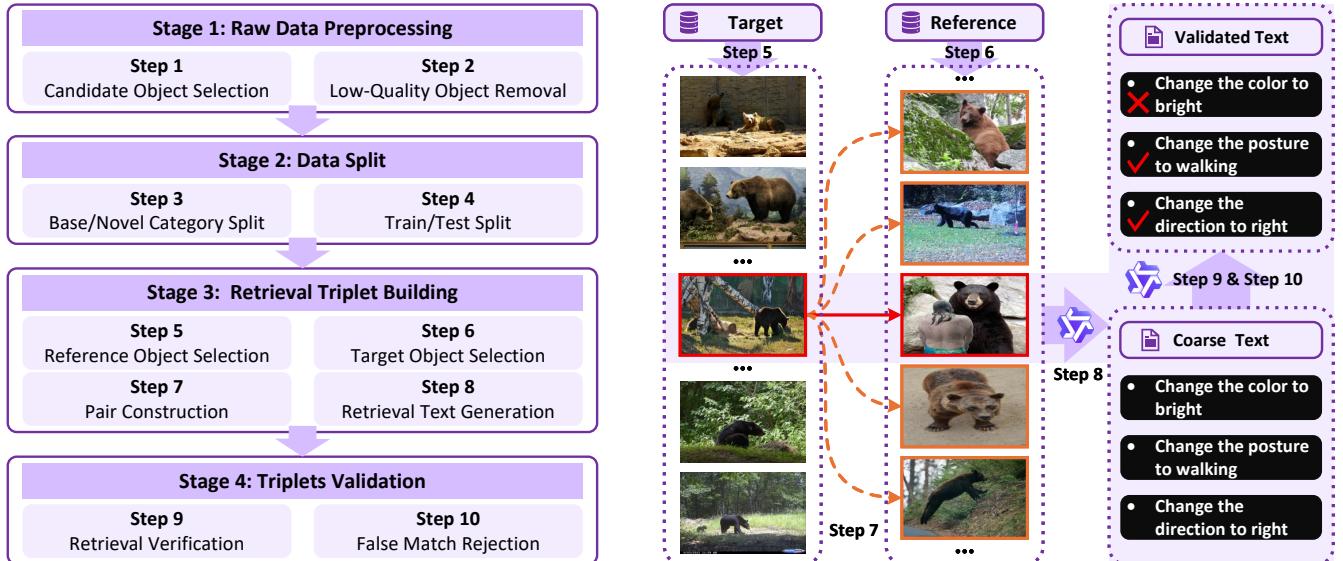
##### B.2.1 Stage 1: Raw Data Preprocessing.

This stage prepares the raw COCO2017 dataset by filtering images and objects to ensure high-quality, diverse candidate objects suitable for the composed object retrieval task. This stage consists of the following:

- **Step 1: Possible Candidate Object Selection.** This step filters COCO2017 images using LVIS annotations to select diverse, semantically relevant objects. The strict criteria exclude objects that are too small ( $< 3\%$  of the image area) or too large ( $> 80\%$ ), retaining those with mask areas that exceed 20% of their bounding box. Categories with fewer than two distinct images or more than three objects in the same category are discarded. For categories exceeding 300 samples, the size is capped at 300, prioritizing images from varied sources to enhance diversity and ensure high-quality candidate objects.
- **Step 2: Low-Quality Target Object Removal.** To ensure data quality, we employ QWen2.5-VL with a carefully designed prompt to systematically remove low-quality target objects that exhibit occlusion, blurriness, or incomplete shapes. Each candidate object is highlighted with a red bounding box based on LVIS annotations and subsequently evaluated for visual clarity and structural completeness. Through this automated assessment process, only objects that are clearly identifiable and free from visual artifacts are retained for further processing. The prompt is as follows:



**Figure 5** Category distribution in the proposed COR127K dataset. Different colors indicate different subsets (red: Training Set, blue: Test-Base Set, green: Test-Novel Set). Zoom in for detailed information.



**Figure 6** (Left): Fully automated 4-stage, 10-step annotation pipeline for COR127K. (Right): Illustration of single sample-pair sampling and retrieval-text generation.

### QWen-VL Prompt for Step 2

You are a data quality control expert. Please evaluate whether the input [IMAGE] meets the following:

- 1) Contains at least  $\{ins\_len\}$  identifiable instances of the  $\{cat\_name\}$  category.
- 2) The  $\{cat\_name\}$  objects are complete, clear, and not occluded or blurry.

Return 1 if all the conditions are met; otherwise, return 0. Only return 1 or 0, with no other content.

where [IMAGE] is the image token, {cat\_name} is the category of the object, and {ins\_len} is the number of objects contained in the image. These can be obtained from the annotations of LVIS.

**Stage 1** ensures a robust foundation by curating a diverse set

of high-quality objects, setting the stage for effective data splitting and the construction of a retrieval triplet.

### B.2.2 Stage 2: Data Split.

This stage organizes the data into balanced subsets for training and testing, ensuring a clear separation between the seen and unseen categories to prevent data leakage. This stage consists of the following:

- **Step 3: Base/Novel Category Split.** Categories are divided into 330 base classes and 78 novel classes using a ratio of 4:1. This split ensures a balanced distribution, enabling evaluation on both familiar and novel categories, which forms the basis for train/test set construction.
- **Step 4: Train/Test Set Partitioning.** All novel category samples are assigned to the Test-Novel set, while base

category samples are divided into Train-Base and Test-Base in a 3:1 ratio. This creates three disjoint subsets: Train-Base, Test-Base, and Test-Novel, ensuring robust training and evaluation without overlap.

By carefully partitioning the data, **Stage 2** establishes a structured framework for training and testing, enabling a reliable evaluation of retrieval performance.

### B.2.3 Stage 3: Retrieval Triplet Building.

This stage constructs reference-target pairs and generates retrieval texts, ensuring semantic relevance and visual clarity for effective retrieval. This stage consists of the following:

- Step 5: Reference Object Selection.** Reference objects are selected based on LVIS annotations with constraints: 1) the image contains only one instance of the given category, and 2) the object occupies at least 5% of the image area for sufficient visibility. This ensures clear and discriminative reference objects for retrieval tasks.
- Step 6: Target Object Selection.** Target candidate objects are categorized into different configurations (*e.g.*, 1p0n, 1p1n, 1p2n, 2p0n, 2p1n, 3p0n) to ensure semantic separability. For cases involving negative objects (*i.e.*, 1p1n, 1p2n and 2p1n) a two-step filtering strategy is implemented. 1) DINOv2 extracts object-level features, and pairs with a cosine similarity exceeding 0.8 are discarded. 2) using QWen2.5-VL and a specific prompt, we verify that positive objects (marked with red boxes) and negative objects (marked with blue boxes) exhibit clear differences in attributes (*e.g.*, color, shape, pose), thereby ensuring reliable retrieval. The prompt is as follows:

#### QWen-VL Prompt for Step 6

*You are a data quality control expert. Please evaluate whether the input [IMAGE] meets the following:*

*The {cat\_name} object marked with a red bounding box is clearly distinguishable from the {cat\_name} object marked with a blue bounding box in terms of the attributes (e.g., color, shape, pose, action, or appearance).*

*Return 1 if the condition is met; otherwise, return 0. Only return 1 or 0, with no other content.*

- Step 7: Target-Reference Pair Construction and Quality Assurance.** Target-reference pairs are constructed, with each target object paired with up to five reference objects from different images. We use QWen2.5-VL and given prompt to verify: 1) reference and target objects are distinguishable in attributes (*e.g.*, color, shape, action); 2) both objects are complete, clear, and free from blur or occlusion; and 3) the target object is

distinguishable from same-category distractors in the target image. This ensures pairs meet retrieval task requirements. The prompt is as follows:

#### QWen-VL Prompt for Step 7

*You are a data quality control expert. The first [IMAGE] (reference image) contains a {cat\_name} object marked with a red bounding box as the reference object. The second [IMAGE] (target image) contains a {cat\_name} object marked with a red bounding box as the target object. Verify the following:*

- 1) The reference and target objects are distinguishable in attributes such as color, shape, or action.*
  - 2) Both objects are complete, clear, and free from motion blur or occlusion.*
  - 3) The target object in the target image is distinguishable from same-category interfering objects.*
- Return 1 if all the conditions are met; otherwise, return 0. Only return 1 or 0, with no other content.*

- Step 8: Retrieval Text Generation.** We use QWen2.5-VL and a specific prompt generates concise, semantically accurate retrieval texts for quality-verified pairs. For each pair (objects marked in red boxes), the model describes attribute differences in a structured format: [ (change1), (change2), (change3) ], with each phrase limited to 10 words. Texts avoid category names, terms like “reference” or “target,” and use simple vocabulary. For animals, dynamic attributes (*e.g.*, pose, action, color) are emphasized; for non-living objects, static attributes (*e.g.*, shape, color, layout) are used, ensuring discriminative and unambiguous descriptions. The prompt is as follows:

#### QWen-VL Prompt for Step 8

*You are a data annotation expert. The first [IMAGE] shows a {cat\_name} object (reference), and the second [IMAGE] shows a {cat\_name} object (target), both marked with red boxes. Other same-category objects may appear in the target image as distractors.*

*Describe the changes from the reference to the target object so the target can be identified.*

- 1) For animals: use dynamic attributes (e.g., pose, action, appearance, color, pattern, direction, style).*
- 2) For inanimate objects: use static attributes (e.g., shape, position, color, direction, style).*

*Avoid using the words “reference,” “target,” {cat\_name}, or uncommon terms. Use simple language and follow the format: [ (change1), (change2), (change3) ], with each change no longer than 10 words.*

**Stage 3** creates high-quality reference-target pairs and descriptive texts that form the core of the COR127K dataset and allow precise object retrieval.

#### B.2.4 Stage 4: Triplets Validation.

This final stage validates the retrieval triplets to ensure semantic correctness and specificity, eliminate ambiguities, and improve reliability.

- **Step 9: Positive Retrieval Verification.** QWen2.5-VL verifies that the reference object and the retrieval text accurately identify the target object. Given a reference object (first image, red-box marked) and retrieval text, the model checks if the described attribute changes match the target object (second image, red-box marked), returning 1 for valid matches and 0 otherwise, ensuring semantic alignment. The prompt is as follows:

##### QWen-VL Prompt for Step 9

You are a data quality control expert. The first [IMAGE] contains a {target\_cat} object marked with a red bounding box.  
Based on this object and the attribute description {retrieval\_text}, determine whether the red-boxed object in the second image matches the description.  
Return 1 if all the conditions are met; otherwise, return 0. Only return 1 or 0, with no other content.

- **Step 10: False Match Rejection.** To ensure specificity, using QWen-VL and a specific prompt, we verify that the retrieval text does not match non-target objects. The target object in the target image is masked out and the model evaluates whether any remaining object incorrectly matches the retrieval text when paired with the reference object (red-box marked). A return value of 1 indicates that there are no false matches, improving the reliability of the dataset by reducing ambiguity.

##### QWen-VL Prompt for Step 10

You are a data quality control expert. The first [IMAGE] contains a {target\_cat} object marked with a red bounding box. The second [IMAGE] is the target image with the target object masked out. The attribute change description is: {retrieval\_text}.  
Verify that no object in the second image matches the description when combined with the marked object in the first image.  
Return 1 if all the conditions are met; otherwise, return 0. Only return 1 or 0, with no other content.

Stage 4 guarantees the accuracy and specificity of the triplets,

ensuring that the COR127K dataset is robust and suitable for advanced retrieval tasks.

### B.3 Pipeline Illustration Using a Sample

We illustrate the construction process of a retrieval triplet (*i.e.*, target object, reference object, retrieval text) using a sample from Fig. 6 (right). In Stage 1, the candidate objects have been filtered; in Stage 2, the training set and the testing set have been determined. The specific steps for constructing the retrieval triplet are as follows:

First, in Step 5, a specific sample is selected from the eligible target objects, *i.e.*, a bear object marked with a red box. Next, in Step 6, five candidate reference objects meeting similarity criteria are randomly sampled from the reference set, *i.e.*, bear objects marked with yellow boxes. For each candidate reference object, we use a multi-modal large model in Step 7 to verify whether it is suitable to form a target-reference pair with the target object.

If the pair passes the verification in Step 7, we proceed to Step 8. The target image and the reference image are fed into the multi-modal large model, with the corresponding objects marked by red bounding boxes. Based on a given prompt, three candidate retrieval texts are generated (*e.g.*, *change the color to bright*, *change the posture to walking*, *change the direction to right*), forming the initial retrieval triplet (target object, reference object, retrieval text).

In Stage 4, we perform quality validation on the generated coarse retrieval triplets. In Step 9, the target object and reference object are cropped based on their bounding boxes and input into the multi-modal large model. Using the retrieval text from Step 8, we evaluate: 1) whether the target object can be identified using the retrieval text and reference object; and 2) whether the retrieval text accurately describes the attribute changes. If the triplet passes this initial check, we remove the target object's bounding box, leaving only the background, and reevaluate using the multi-modal large model to determine whether the target object can still be identified with the retrieval text and the reference object. If it can, this indicates potential noise in the background, and the sample is filtered out. For instance, since the target contains two black bears, the retrieval text “*change the color to bright*” fails to accurately describe the attribute change, leading to its rejection during quality control.

Through these steps, we ensure the quality of the retrieval triplets, providing reliable data for the COR task.

## C Additional Experiments

### C.1 Additional Qualitative Results

To further demonstrate the effectiveness of our method (CORE), we present additional qualitative results in Fig. 7. These visualizations highlight the robustness and versatility of our approach in various retrieval scenarios.

Specifically:

1. In examples **a** and **b**, our model successfully retrieves objects that are challenging to describe textually, showcasing its strong semantic understanding.
2. In examples **c** and **e**, the model accurately distinguishes negative objects within the same category, demonstrating precise discriminative capabilities.
3. In examples **d**, **e**, and **f**, it effectively retrieves multiple target objects within a single scene, highlighting its ability to handle complex multi-object scenarios.

## C.2 Additional Ablation Studies

In the setting of COR, we use the reference image  $I_{ref}$ , the reference object mask  $M_{ref}$ , and the retrieval text  $T_{ret}$  as composed expressions, which are then treated as the prompt to retrieve target objects  $M_{tar}$  in the target image  $I_{tar}$ .

To evaluate the effectiveness of the COR and determine whether the composed expressions constructed as described enhance retrieval performance, we conducted ablation experiments on their constituent components. Specifically, we investigated the impact of different components of the composed expressions on retrieval performance. The settings for the ablation study are as follows:

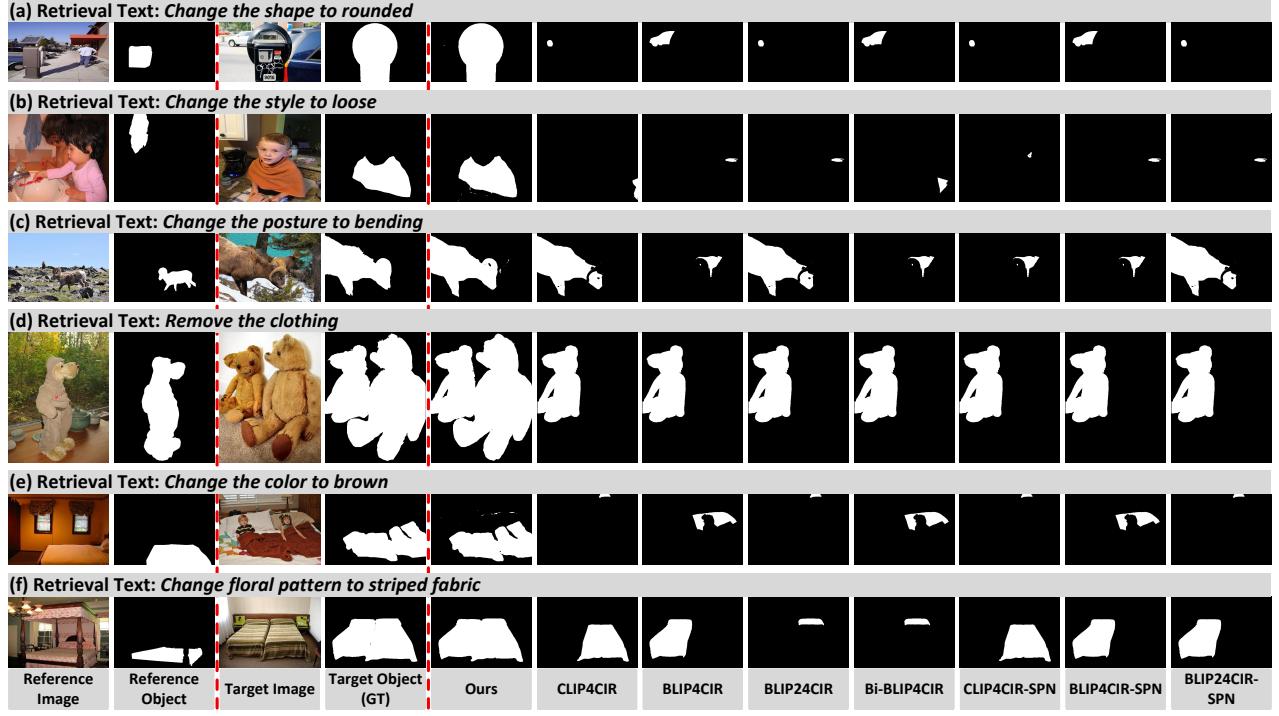
1. Only remove the retrieval text while retaining the reference image and reference object mask (*i.e.*,  $I_{ref} + M_{ref}$ ), in which case the AVTI module will be discarded, and only the RRE module will be used;
2. Only remove the reference object mask while retaining the reference image and retrieval text (*i.e.*,  $I_{ref} + T_{ret}$ ), in which case the RRE module will be discarded, and only the AVTI module will be used;
3. Remove both the reference object mask and the retrieval text while retaining only the reference image (*i.e.*,  $I_{ref}$ ), in which case both the RRE and AVTI modules will be discarded;
4. Remove both the reference image and reference object mask while retaining only the retrieval text (*i.e.*,  $T_{ret}$ ), in which case both the RRE and AVTI modules will be discarded;

The experimental results are presented in Tab. 5. The results demonstrate that retrieval using composed expressions achieves the best performance (*i.e.*, Ours). The detailed findings are as follows.

1. **Removing Retrieval Text ( $T_{ret}$ )**: When only the retrieval text is excluded from the composed expression, the Dice score on Test-base decreases from 0.7703 to 0.7411 (a 3.79% drop), and on Test-novel, it falls from 0.7102 to 0.6664 (a 6.17% drop).

2. **Removing Reference Object Mask ( $M_{ref}$ )**: When only the reference object mask is removed, the Dice score on Test-base drops from 0.7703 to 0.7319 (a 4.98% decrease), and on Test-novel, it decreases from 0.7102 to 0.6712 (a 5.49% decrease).
3. **Removing Both Retrieval Text ( $T_{ret}$ ) and Reference Object Mask ( $M_{ref}$ )**: When both the retrieval text and reference object mask are excluded (retaining only the reference image  $I_{ref}$ ), performance degrades significantly. The Dice score on Test-base falls from 0.7703 to 0.6770 (a 12.11% drop), and on Test-novel, it drops from 0.7102 to 0.6137 (a 13.59% drop).
4. **Removing Both Reference Image ( $I_{ref}$ ) and Reference Object Mask ( $M_{ref}$ )**: When both the reference image and reference object mask are removed (retaining only the retrieval text  $T_{ret}$ ), the Dice score on Test-base decreases to 0.6767 (a 12.15% drop), and on Test-novel, it falls to 0.6144 (a 13.49% drop).

In summary, the use of complete composed expressions, integrating the retrieval text ( $T_{ret}$ ), the reference object mask ( $M_{ref}$ ), and the reference image ( $I_{ref}$ ), consistently delivers the highest retrieval performance. These results underscore the necessity of incorporating all components of composed expressions, as their synergistic integration significantly outperforms single-modal retrieval approaches, ensuring robust and accurate retrieval outcomes.



**Figure 7** Additional qualitative results. From left to right: 1) reference image  $I_{ref}$ ; 2) reference object  $O_{ref}$ ; 3) target image  $I_{tar}$ ; 4) target object  $O_{tar}$ ; 5) our result; 6) CLIP4CIR; 7) BLIP4CIR; 8) BLIP24CIR; 9) Bi-BLIP4CIR; 10) CLIP4CIR-SPN; 11) BLIP4CIR-SPN; 12) BLIP24CIR-SPN.

Setting	$I_{ref}$	$M_{ref}$	$T_{ret}$	COR127K-Test-Base					COR127K-Test-Novel				
				Dice $\uparrow$	IoU $\uparrow$	MAE $\downarrow$	mDice $\uparrow$	mIoU $\uparrow$	Dice $\uparrow$	IoU $\uparrow$	MAE $\downarrow$	mDice $\uparrow$	mIoU $\uparrow$
$I_{ref} + M_{ref}$	✓	✓	✗	0.7411	0.6601	0.0833	0.8427	0.7813	0.6664	0.5813	0.1054	0.7994	0.7313
$I_{ref} + T_{ret}$	✓	✗	✓	0.7101	0.6302	0.0943	0.8238	0.7609	0.6460	0.5644	0.1122	0.7871	0.7194
$I_{ref}$	✓	✗	✗	0.6770	0.5974	0.1057	0.8036	0.7389	0.6137	0.5359	0.1213	0.7682	0.7007
$T_{ret}$	✗	✗	✓	0.6767	0.6002	0.1036	0.8043	0.7415	0.6144	0.5363	0.1211	0.7686	0.7012
<b>CORE (Ours)</b>	✓	✓	✓	<b>0.7703</b>	<b>0.6955</b>	<b>0.0741</b>	<b>0.8603</b>	<b>0.8044</b>	<b>0.7102</b>	<b>0.6290</b>	<b>0.0858</b>	<b>0.8276</b>	<b>0.7652</b>

**Table 5** Ablation study on the information used in the composed expressions. ✓: enabled; ✗: disabled.