CoLLM: A Large Language Model for Composed Image Retrieval

Supplementary Material

7. Additional Method Details

7.1. Modification Text Synthesis Templates

As described in Sec. 3.2 and illustrated in Fig. 3 (right), the synthesis of modification text plays a vital role in the initial pre-training stage. During this stage, we generate modification text w_i^* by randomly choosing one of the templates provided below:

1. "show w_i instead of w_j"
2. "w_i instead of w_j"
3. "show w_i rather than w_j"
4. "w_i rather than w_j"
5. "rather than w_j, show w_i"
6. "rather than w_j, w_i"
7. "instead of w_j, w_i"
8. "w_j, changed to w_i"
9. "not w_j, but w_i"
10. "show w_i, not w_j"
11. "w_j is missing, w_i"
12. "w_i, and w_j is missing"
13. "remove w_j, add w_i"
14. "add w_i, remove w_j"
15. "w_j become w_i"

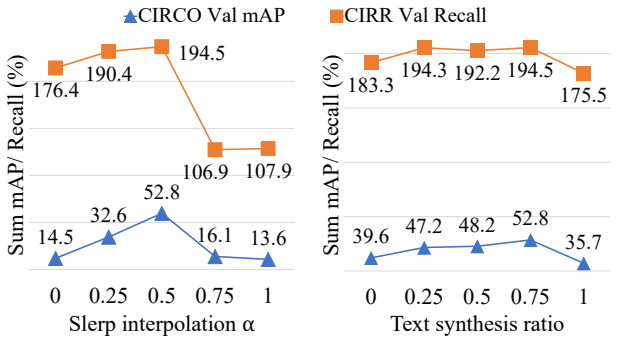
The templates are designed based on our analysis of the real modification texts from the CIRCO and CIRR datasets, aiming to integrate information from both the reference and target images. While the fully synthesized modification texts may not be grammatically or semantically correct, the language encoder is pre-trained to handle such noise robustly.

7.2. LLM Instruction Template

As stated in Eq. (3)-(5), the input to the LLM must adhere to a specific template. We adopt the LLEM (LLM specialized for text retrieval) instruction format to structure our input instruction as:

```
Instruct: Find the image that matches
          the query.
Query:
Image: [IMAGE]
Text: [TEXT]
```

where `[IMAGE]` corresponds to $g(\mathbf{h}_\mathbf{i}^*)$ or $g(\mathbf{h}_\mathbf{i})$, and `[TEXT]` corresponds to w_i^* or w_i when training with image-caption pairs or triplets, respectively. If either `[IMAGE]` or `[TEXT]` is missing, the line `Image: [IMAGE]` or `Text: [TEXT]` is removed from the query accordingly.



(a) Different Slerp α values (b) Different Text synthesis ratios

Figure 7. Performance of our model under varying Slerp α values and text synthesis ratios. Text synthesis = 75% in (a) and Slerp $\alpha = 0.5$ in (b). The optimal configuration is achieved with $\alpha = 0.5$, where text synthesis is applied to 75% of the samples.

Table 8. Performance of our CoLLM BLIP-L/16 (384×384) fine-tuned on COCO after training on MTCIR and test with different instructions on CIRR Val and Fashion-IQ.

	CIRR Val			FIQ	
	@1	@10	@50	@10	@50
Mean	47.0	85.7	96.0	38.7	60.6
Std	0.10	0.04	0.04	1.02	0.48

7.3. Additional Ablation Studies

We investigate the impact of synthesis strength hyperparameters for the reference image embedding $\mathbf{h}_\mathbf{i}^*$ and modification text w_i^* in Fig. 7. The same training setup as described in Sec. 5.4 is used. As explained in Sec. 3.2, the Slerp α value represents the interpolation distance of the reference image embedding relative to the original $\mathbf{h}_i'$. A larger α value indicates a greater difference between $\mathbf{h}_\mathbf{i}^*$ and $\mathbf{h}_\mathbf{j}^*$. For modification text synthesis, it is applied partially to the training samples. When synthesis does not occur, $w_i^* = w_i$, the caption of the target image. From the figures, the model achieves optimal performance with Slerp $\alpha = 0.5$ and text synthesis applied to 75% of the training samples. Performance drops significantly with higher α values.

To assess the robustness of our model across different instructions, we generate nine additional instruction variants using Claude Sonnet, as described in Sec. 7.2:

1. "Identify the image corresponding to the given query."
2. "Locate the image that aligns with the provided query."
3. "Search for the image that fits the query."
4. "Retrieve the image that matches the query."
5. "Determine the image that corresponds to the query."

6. “Select the image that best matches the query.”
7. “Find the image associated with the query.”
8. “Choose the image that matches the given query.”
9. “Match the query to its corresponding image.”

As shown in Table 8, when tested with ten different instructions, our model demonstrates robustness, exhibiting negligible performance variation across instruction variants.

8. Dataset Construction Details

8.1. MTCIR

Image Pairing. The process follows CIRR [37] with some modifications. Specifically, we use CLIP-L-14/336 [45] to extract image features instead of ResNet-152 [16] pretrained on ImageNet [10]. This updated network provides more robust features compared to the previous one. Groups of six similar images are formed, where each image is added to the group with a similarity score between 0.5 and 0.95 relative to the first image, using an interval of 0.03. Groups with fewer than six members are discarded. Pairs are then constructed between consecutive images and between the first image and all other images within each group.

Modification Text Categories. We define six categories as outlined in Table 9, drawing inspiration from previous works, CIRR [37] and CIRCO [4]. The largest category, Attribute Changed, comprises approximately half of the dataset's text. Object Added and Object Removed have similar proportions, each accounting for around 20% of the dataset. The remaining three categories collectively represent less than 10% of the dataset.

Prompt. The input to Claude Sonnet 3 is detailed in Table 22. It begins with a system prompt that provides an overview of the task to the model. Next, the detailed image captions (`[CAPTION]`) generated by LLaVA-Next-34B [35] are included, followed by the definitions of categories outlined in Table 9.

For each category, real captions and modification texts from CIRR [4] (with some corrections) are provided as examples to enable in-context learning. Both forward examples (`[FORWARD]`: describing changes from image 1 to image 2) and backward examples (`[BACKWARD]`: describing changes from image 2 to image 1) are included to ensure the model accurately understands the task.

Additionally, during the initial iterations, a set of bad examples (`[BAD EXAMPLES]`), which fail to describe the changes correctly, is collected and incorporated into the prompt to refine the model's understanding. Finally, the JSON output requirement is specified at the end for straightforward parsing.

This prompt structure allows for the generation of multiple modification texts in both forward and backward directions for a single image pair, reducing costs while ensuring all detailed differences are captured.

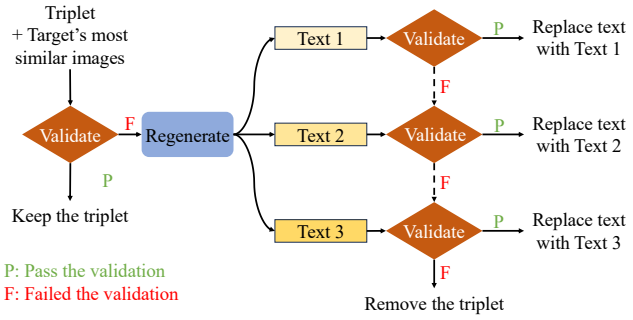


Figure 8. **Pipeline for regenerating text in CIR benchmarks.** Starting with a triplet and dataset-similar images, we assess text ambiguity by evaluating the model's ability to select the correct target image. If the model fails, three new texts with varying levels of specificity are generated and re-tested. The process concludes either when ambiguity is resolved by any of the texts or when the triplet is removed if ambiguity remains unresolved.

We include 20 MTCIR samples in `mtcir_samples.html` of the supplementary material, showcasing various modification texts and categories for each. Our pipeline effectively captures the differences between pairs without producing lengthy sentences.

Diversity and Quality. We evaluate the diversity of the MTCIR dataset by analyzing variability in both image content and textual descriptions. For image diversity, we utilize RAM [66] to tag images in our dataset and those in previously published benchmarks. For textual diversity, we employ spaCy [17] to process modification texts. As presented in Table 10, our dataset achieves the highest count of unique visual tags and textual tokens, indicating superior diversity.

To assess dataset quality, we employ state-of-the-art LLMs to evaluate the generated modification texts. Specifically, we use GPT-4o [19] and DeepSeek-V3 [33], two leading-performing models, to validate the accuracy of modifications. Each model is provided with captions of the reference and target images alongside our generated modification text and tasked to identify any incorrect transformations described by the modification text. The evaluation is conducted on 1000 randomly selected samples from the MTCIR dataset. Our dataset achieves good sample ratios of 83.4% and 85.2%, as rated by GPT-4o and DeepSeek-V3, respectively.

8.2. Refined Benchmarks

The refinement pipeline, as detailed in Sec. 4.2, is illustrated in Fig. 8. It consists of three steps to ensure that only “good” samples remain in the benchmarks.

In the first and final steps, sample validation is conducted using the prompt outlined in Table 24. The reference image is included as `[REFERENCE IMAGE]`, while the target image and all hard negative samples (the top-3 similar images to the target image) are concatenated hor-

Table 9. Modification text categories define six types of changes that can occur between two images. These categories capture the variety of transformations described in the dataset.

Category ID	Name	No. Samples	Definition
attribute_change	Attribute Changed	8,139,415 (45.95%)	The same object is present in both images, but the attributes of the object have changed, not including the quantity or number.
added_object	Object Added	3,856,642 (21.77%)	An object or objects is present in the second image that is not present in the first image.
removed_object	Object Removed	3,695,121 (20.86%)	An object or objects is present in the first image that is not present in the second image.
relationship_change	Relationship Changed	1,122,834 (6.34%)	If the objects in the images are the same, but the relationship between the objects has changed.
viewpoint_change	Viewpoint Changed	650,735 (3.67%)	The viewpoint from which the image is taken has changed between the two images.
number_change	Number Changed	249,098 (1.41%)	The same object is present in both images, but the number of the object has changed.

Table 10. MTCIR is more diverse than previous datasets in both visual and textual information.

	CIRR [37]	LaSCo [28]	CC-CoIR [58]	MTCIR
# Unique Visual Tags	2,787	3,421	4,072	**4,198**
# Unique Text Tokens	5,838	16,270	18,031	**164,914**

izontally in random order as [CANDIDATE IMAGES]. Given the modification text as [MODIFICATION TEXT], the Claude Sonnet model is tasked with selecting the correct target image. Each sample is evaluated three times with different orders of [CANDIDATE IMAGES]. Samples that pass in at least two evaluations are considered "good." Occasionally, the model refuses to answer, providing responses beginning with "I apologize...", a behavior triggered by its harmful content detection mechanism. Such samples are excluded from the benchmarks.

In the second step, modification texts for ambiguous samples are regenerated. Claude Sonnet 3 is used to create new modification texts, guided by the prompt described in Table 23. The original triplet is retained as input with [REFERENCE IMAGE], [TARGET IMAGE], and [MODIFICATION TEXT]. Additionally, some randomly selected "good" samples from the first step are included as [GOOD SAMPLES] to align the model's output with human expectations. The prompt instructs the model to generate three modification texts, ranging from coarse to fine, to minimize inference costs.

We present some "good" samples classified by our pipeline from both the CIRR and Fashion-IQ validation sets in Fig. 10. The modification texts in these samples are sufficiently detailed to distinguish the target image from hard-negative samples, which are visually similar to the target. Examples of Text 1-3 are shown in Fig. 11 along with new chosen text. In these examples, our pipeline prioritizes using coarse modification text to replace ambiguous ones. At each level, an additional detail is introduced to further differentiate the correct target image from the other hard-negative samples.

9. Additional Experimental Details

9.1. Pre-training on Image-Caption Pairs.

CIRR Recall on Subset Metric. During our evaluation on the CIRR validation set, we observed some contradictions between Recall on the whole index set and Recall on the subset. These inconsistencies raise concerns about the reliability of the recall on the subset metric. We evaluate BLIP-L baselines with the vision encoder BLIP-L/16 fine-tuned on COCO, using different settings and the synthetic CIR dataset, as shown in Table 11. While our proposed MTCIR achieves the best results, some interesting observations emerge regarding Recall Subset.

Firstly, the initial model already outperforms the fine-tuned models trained on previous synthetic datasets. Notably, this model uses only modification text in the query and achieves the second-best performance, with a small gap to the best-performing model. Additionally, the model fine-tuned on WebCoVR shows slight degradation in performance when both image and text are used in the query. These results suggest that the reference image does not play a significant role in the Recall Subset, indicating that this metric is unreliable for evaluating CIR methods.

Image-Caption Datasets. We provide additional details about the pre-training dataset mentioned in the main paper. Our dataset, comprising 5 million image pairs, follows the Slerp-TAT [21] protocol: nearly 3 million samples are sourced from CC3M [51], 2 million random samples are selected from LAION-115M [50], and 558K samples are obtained from LLaVA-558K [34]. All captions are synthetically generated using the BLIP [29] model.

Implementation Details. We utilize SFR-Embedding-2 [47] as our LLM backbone. For the vision encoder, we employ pre-trained OpenAI CLIP-B/32 and CLIP-L/14 [45]. We adopt BLIP-L/16 pre-trained weights from the official repository [29]. To ensure a fair comparison, all

Table 11. Unreliability of Recall Subset metric on CIRR validation. The BLIP-L/16 384 ft. COCO model is trained on various CIR datasets and evaluated under different query settings. Notably, even without fine-tuning, the initial model achieves the second-best performance, surpassing all previous datasets while not using the reference image in the query. **Bold** and underline are used to highlight the best and second-best scores, respectively.

Fine-tuned Dataset	Query		Recall Index ↑			Recall SubSet ↑		
	Image	Text	@1	@10	@50	@1	@2	@3
None		✓	38.5	75.1	89.3	75.5	88.4	94.2
	✓	✓	20.6	54.7	76.2	67.0	84.5	91.7
LaSCo [28]		✓	23.4	59.2	80.0	65.1	82.9	91.0
	✓	✓	40.4	80.9	94.9	68.2	84.0	91.5
WebCoVR [59]		✓	34.3	73.0	88.7	73.3	87.5	93.7
	✓	✓	40.6	81.5	94.5	72.7	87.4	93.5
MTCIR		✓	22.2	58.1	79.4	67.3	84.9	92.9
	✓	✓	**43.9**	**84.1**	**95.4**	**75.6**	**89.3**	**95.4**

Table 12. Performance of different BLIP-L vision encoders after pre-training with image-caption pairs. The model demonstrates a significant improvement when utilizing a more advanced image encoder.

BLIP-L Variants	CIRCO (mAP↑)			CIRR (Rec.↑)			FIQ (Rec.↑)	
	@5	@10	@50	@1	@10	@50	@10	@50
Base	19.4	20.4	23.3	35.1	78.6	94.2	34.6	56.0
Fine-tuned COCO	26.0	26.7	29.9	41.8	81.9	95.3	37.0	57.4

pre-training experiments use 224×224 pixel images.

LoRA [18] tuning is applied with a rank of 64 for large models (BLIP-L and CLIP-L) and 32 for the CLIP-B model, using a dropout rate of 0.1. The BLIP-L vision encoder is frozen during training, while other model variants are tuned on both the LLM and vision encoder parts.

Pre-training is conducted over one epoch using a constant learning rate of $1e^{-4}$ and a batch size of 1024. All experiments are performed on 8 NVIDIA A100 40GB GPUs. The training script is based on the LLaVA [34] codebase, while the evaluation script is adopted from WebCoVR [59].

Different BLIP Vision Encoders. We note that two BLIP-L vision encoders are used and compared to other baselines. During the pre-training stage, our model is compared with the BLIP-L base, which processes images at a size of 224×224 pixels. In the fine-tuning stage, since other approaches use an enhanced BLIP-L, we also train another CoLLM variant using the BLIP-L fine-tuned on COCO [32] captions. This variant utilizes a larger image size of 384×384 pixels.

The performance differences between these variants are presented in Table 12. A significant gap can be observed between the two variants, particularly in the CIRCO metrics.

Additional Quantitative Results. In Table 13, we provide detailed results of models evaluated on the Fashion-IQ Validation set without training on CIR triplet datasets. This table extends Table 2. Our models achieve the best results on most metrics when using CLIP-L and BLIP-L vision encoders. For CLIP-B, our CoLLM ranks second in the dress and shirt categories.

Table 13. Full results of Fashion-IQ validation, extension of Table 2. **Bold** and underlined values indicate the best and second-best scores within each vision encoder group. Models that incorporate LLMs in their architectures are marked with ⋆, and results reproduced by our team are denoted with ‡.

Model	Dress		Shirt		Toptee	
	@10	@50	@10	@50	@10	@50
OpenAI CLIP-B/32						
PALAVRA [8]	17.3	35.9	21.5	37.1	20.6	38.8
SEARLE [4]	18.5	39.5	24.4	41.6	25.7	46.5
Slerp-TAT [21]	19.2	42.1	23.1	42.0	26.6	47.8
CIReVL⋆ [26]	**25.3**	**46.4**	**28.4**	**47.8**	**31.2**	**53.9**
CoLLM⋆	22.9	43.8	24.9	45.1	26.4	46.8
OpenAI CLIP-L/14						
Pic2World [49]	20.0	40.2	26.2	43.6	27.9	47.4
SEARLE [4]	20.5	43.1	26.9	45.6	29.3	50.0
LinCIR‡ [15]	20.9	41.9	29.2	47.4	29.2	50.5
ContextI2W [55]	23.1	45.3	29.7	48.6	30.6	52.9
Slerp-TAT [21]	23.4	45.1	29.6	46.5	32.0	51.2
CIReVL⋆ [26]	**24.8**	44.8	29.5	47.4	31.4	**53.7**
CoLLM⋆	24.6	**46.5**	**33.4**	**50.5**	**32.4**	51.6
BLIP-L/16						
Slerp-TAT [21]	29.2	50.6	32.1	51.6	37.0	57.7
CoLLM⋆	**30.8**	**53.8**	**34.2**	**53.9**	**38.7**	**60.2**
BLIP-L/16 384×384; fine-tuned COCO						
CoLLM⋆	32.7	54.1	38.1	57.5	40.3	61.0

We also examine the effect of LoRA-tuning on different vision encoders during pre-training, as shown in Table 14. While CLIP models show significant improvement with vision encoder tuning, BLIP-L exhibits a performance drop in both CIRCO and Fashion-IQ. This issue may stem from BLIP's synthetic captions. CLIP models, trained on noisier captions, benefit from further tuning. In contrast, BLIP, as a more advanced model, is already well-trained, and additional vision encoder tuning on a smaller dataset might lead to overfitting.

9.2. Fine-tuning

Implementation Details. To ensure a fair comparison across models and datasets, we implement several adjustments in our training process. We reduce the number of trainable parameters by setting the LoRA rank and alpha to 16. At this stage, only the LLM is fine-tuned, as the vision encoder features are already aligned during the pre-training phase. Other settings remain consistent with the pre-training stage. For the BLIP-L vision encoder, we use BLIP-L/16 fine-tuned on COCO captions and increase the

Table 14. Performance of CoLLM with different vision encoder and LoRA tuning applied to Vision Encoder (ViT). CLIP models require ViT tuning to achieve optimal performance, whereas BLIP-L performs better with a frozen ViT. **Bold** denotes the best score for each vision encoder.

Vision Encoder	LoRA ViT	CIRCO (mAP↑)			CIRR (Recall↑)			Fashion-IQ (Recall↑)							
								Dress		Shirt		Toptee		Average	
		@5	@10	@50	@1	@10	@50	@10	@50	@10	@50	@10	@50	@10	@50
OpenAI CLIP-B/32		11.7	12.0	13.7	23.2	67.4	91.1	20.3	40.2	23.8	42.1	24.7	42.6	22.9	41.6
OpenAI CLIP-B/32	✓	**12.9**	**13.2**	**15.0**	**28.6**	**71.8**	**92.7**	**22.9**	**43.8**	**24.9**	**45.1**	**26.4**	**46.8**	**24.8**	**45.2**
OpenAI CLIP-L/14		16.1	16.9	19.1	24.5	69.2	90.9	23.5	42.4	32.7	49.1	29.8	48.9	28.7	46.8
OpenAI CLIP-L/14	✓	**20.3**	**20.8**	**23.4**	**29.7**	**72.8**	**91.5**	**24.6**	**46.5**	**33.4**	**50.5**	**32.4**	**51.6**	**30.1**	**49.5**
BLIP-L/16		**19.4**	**20.4**	**23.3**	35.1	78.6	94.2	**30.8**	**53.8**	34.2	53.9	**38.7**	**60.2**	**34.6**	**56.0**
BLIP-L/16	✓	18.6	19.4	22.1	**37.7**	**79.2**	**94.6**	30.6	53.4	**34.4**	**54.1**	37.3	59.7	34.1	55.7

Table 15. BLIP-L/16 (384 × 384) fine-tuned on COCO exhibits rapid overfitting on the LaSCo dataset after the first training epoch. Performance is measured by Recall Sum on CIRR validation set (@1,10,50) and Fashion-IQ (@10,50).

Epoch	1	2	3	4	5
CIRR Val	**216.2**	214.3	214.8	212.7	213.4
Fashion-IQ	**68.9**	63.9	62.7	62.5	62.6

image input size to 384 × 384 pixels, aligning with prior methodologies.

For experiments involving LaSCo and our MTCIR, both BLIP-L and CoLLM models are trained for one epoch. We utilize the publicly available BLIP-L weights pretrained on WebCoVR. Despite an imbalance in sample size between WebCoVR and LaSCo, extending the training beyond one epoch for LaSCo is impractical, as the model rapidly overfits after the initial epoch (see Table 15).

Additional Quantitative Results. We provide details of models fine-tuned on synthetic CIR datasets in Table 16. Our CoLLM with the BLIP-L vision encoder achieves the best overall performance across most metrics, even surpassing models equipped with larger vision encoders. Using CLIP-L vision encoders, our model achieves the best scores in half of the metrics compared to other methods.

Fashion-IQ detailed results from Table 4 are also presented in Table 18. Our MTCIR consistently enhances the performance of both models across all sub-category metrics of Fashion-IQ. For completeness, we also report the models' performance on the CIRCO benchmark in Table 17. However, we note that CIRCO is not an ideal benchmark for these models due to data leakage concerns. Despite this, our models achieve strong performance, even though other works may have been trained on a subset of the CIRCO images.

Table 19 illustrates the performance drop when the BLIP-L/16 model with resolution 384 × 384, initially fine-tuned on COCO, is directly trained on the MTCIR dataset without further pretraining. While the model trained solely on MTCIR still surpasses previous works shown in Table 3, incorporating a pretraining stage results in substantial improvements in performance metrics.

Table 16. Full results of Fashion-IQ validation, extension of Table 3. **Bold** is used to highlight the best overall scores, while underline marks the best metrics within the same vision encoder group.

Model	Dress		Shirt		Toptee	
	@10	@50	@10	@50	@10	@50
CoCA-L/18 288 × 288						
MagicLens [64]	32.3	52.7	40.5	59.2	41.4	63.0
EVA-CLIP ViT-G/14 364 × 364						
CoVR2 [58]	34.3	56.2	**41.2**	59.3	39.0	59.8
OpenAI CLIP-L/14 224 × 224						
CompoDiff [14]	32.2	46.3	37.7	49.1	38.1	50.6
MagicLens [64]	25.5	46.1	32.7	53.8	34.0	57.7
CoLLM	28.1	51.6	36.3	55.8	34.4	55.1
BLIP-L/16 384 × 384; fine-tuned on COCO						
Omkar et al. [56]	24.6	40.9	33.1	48.4	33.2	50.2
CoLLM	**35.8**	**58.9**	39.6	**59.5**	**42.0**	**63.8**

Table 17. Performance of models training on synthetic datasets on CIRCO benchmark.

Method	Dataset	CIRCO (mAP↑)		
		@5	@10	@50
CoCa-L/18 288 × 288				
MagicLens [64]	MagicLens [64]	**34.1**	**35.4**	**39.2**
EVA-CLIP ViT-G/14 364 × 364				
CoVR2 [58]	WV-CC-CoVIR [58]	28.3	29.6	33.3
OpenAI CLIP-L/14 224 × 224				
CompoDiff [14]	SynTrip18M [14]	12.6	13.4	16.4
MagicLens [64]	MagicLens [64]	29.6	30.8	34.4
CoLLM	MTCIR (ours)	24.4	25.2	28.2
BLIP-L/16 384 × 384; fine-tuned on COCO				
CoVR [59]	WebCoVR [59]	21.4	22.3	25.5
CoLLM	MTCIR (ours)	29.0	29.8	33.4

Qualitative Results. Fig. 12 and Fig. 13 present a per-

Table 18. Full results of Fashion-IQ validation, extension of Table 4. **Bold** values indicate the best score within each method group.

Dataset	Dress		Shirt		Toptee	
	@10	@50	@10	@50	@10	@50
BLIP-L [29]						
LaSCo [28]	20.2	38.5	26.3	43.3	28.0	50.3
WebCoVR [59]	22.0	39.1	30.5	46.1	27.7	44.7
MTCIR (ours)	**32.3**	**55.3**	**40.6**	**58.8**	**40.9**	**63.4**
CoLLM						
LaSCo [28]	34.9	58.2	38.8	58.8	41.8	63.4
MTCIR (ours)	**35.8**	**58.9**	**39.6**	**59.5**	**42.0**	**63.8**

Table 19. Performance of CoLLM (with BLIP-L/16 384×384 finetuned on COCO) is superior when pre-training on 5M image-caption pairs.

Pre-train	CIRR Test			FIQ		Ref. CIRR			Ref. FIQ	
	@1	@10	@50	@10	@50	@1	@10	@50	@10	@50
Yes	45.8	84.7	95.9	39.1	60.7	60.7	92.7	98.2	57.2	76.4
No	42.0	81.8	95.6	34.7	56.3	55.4	90.7	97.8	52.1	73.4

formance comparison of CoLLM after the pre-training stage, BLIP-L, and our CoLLM fine-tuned on the respective datasets. All models use the BLIP-L/16-384 vision encoder fine-tuned on COCO.

The pre-trained model already demonstrates reasonable performance, while the fine-tuned version retrieves a higher number of correct images. Although BLIP-L achieves good results, it struggles with capturing precise image details in some cases (e.g., the second samples in Fig. 12 and Fig. 13).

9.3. Refined benchmarks

Human Studies on Quality. As detailed in Sec. 4.2, we have improved the CIRR [37] and Fashion-IQ [61] validation benchmarks. To evaluate the quality of the newly generated texts in the refined benchmarks, we conducted human studies on random samples from the Regenerated group. The results are summarized in Fig. 9:

1. *Refined CIRR Evaluation:* We used the same strategy as the validation step (Step 1) during the benchmark refinement process. Seven random regenerated samples, along with their original texts, were selected. Participants were asked to identify the target image using the reference image and either the regenerated or original modification text. Alongside the target image, two of its most similar images were included as options. Participants could refuse to answer if they believed there was no or more than one correct answer. From 130 responses, the new refined CIRR benchmark reduced ambiguity, increasing the correct answers by 4%.
2. *Refined Fashion-IQ Evaluation:* A similar process was used for the Fashion-IQ dataset, with 12 questions (4 per category). From 130 collected responses, the refined benchmark significantly addressed the issues in the original dataset, increasing the proportion of correct answers by approximately 17%.
3. *LLM-generated Text Quality Evaluation:* Participants were tasked with identifying which text was more likely generated by LLMs from nine pairs of old and new modification texts across both CIRR and Fashion-IQ datasets. Participants could also refuse to answer. From 125 responses, over half either selected incorrect answers or were unable to distinguish LLM-generated texts, as shown in the last row of Fig. 9.

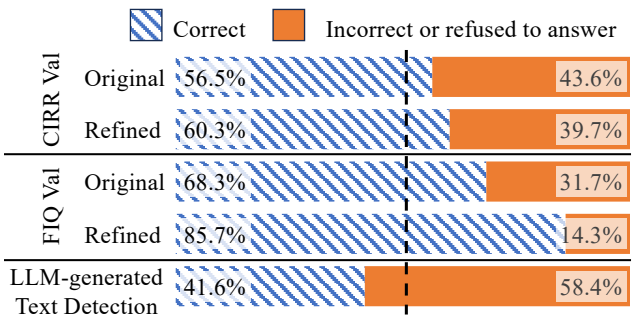


Figure 9. Human studies in evaluating refined benchmarks. Our new refined benchmarks increase the number of correct answers while human finds difficulty in detect AI-generated text.

These surveys validate our assumptions in improving benchmarks. Firstly, the existing evaluation sets contain ambiguities that even humans struggle to resolve. Secondly, regenerating texts significantly reduces these ambiguities, as evidenced by the improvement in human accuracy. Lastly, the newly generated texts align closely with human language, as demonstrated by participants' difficulty in identifying AI-generated texts. This highlights the effectiveness of our pipeline in creating refined and natural benchmarks.

Benchmark Ambiguity. Table 20 presents the performance discrepancy across different evaluation queries on both the original and refined benchmarks, following the analysis in [28]. The BLIP-L/16 (384×384) model, fine-tuned on COCO, is evaluated after training on the MTCIR dataset. Notably, using only the modification text in the query yields high performance in both benchmarks. One possible explanation is that paired images share fewer common features, making the text a crucial factor in retrieval. This observation highlights a potential research direction for improving benchmark design.

Additional Quantitative Results. Table 21 presents the recall metrics for all Fashion-IQ categories, extending Table 5 from the main paper. Our MTCIR continues to enhance model performance, achieving the best results across most metrics. Notably, CoLLM fine-tuned on MTCIR achieves

Table 20. Performance of different query types on the original and refined benchmarks of BLIP-L/16 384×384 fine-tuned on COCO.

Query	CIRR Val			FIQ		Ref. CIRR			Ref. FIQ	
	@1	@10	@50	@10	@50	@1	@10	@50	@10	@50
Composed	43.8	84.1	95.4	37.9	59.2	58.0	91.6	97.9	56.8	76.6
Text	38.5	75.1	89.3	28.4	48.5	52.8	86.7	95.0	49.9	69.9

Table 21. Performance of models on all categories of refined Fashion-IQ validation set. This is an extension of Table 5. **Bold** indicates the highest score, while underlined values represent the best metric within the same vision encoder group.

Method	Dataset	Dress		Shirt		Toptee	
		@10	@50	@10	@50	@10	@50
EVA-CLIP ViT-G/14 364×364							
CoVR2 [58]	WV-CC-VIR [58]	48.6	69.8	**58.5**	74.7	55.4	74.2
OpenAI CLIP-L/16 224×224							
MagicLens [64]	MagicLens [64]	38.0	62.6	49.1	69.9	49.5	**71.9**
CoLLM	MTCIR (ours)	40.9	64.4	53.2	71.1	50.8	70.3
BLIP-L/16 384×384; fine-tuned on COCO							
BLIP-L [29]	MTCIR (ours)	48.1	70.6	58.4	75.6	57.8	76.7
CoLLM	LaSCo [28]	52.2	72.9	57.6	75.1	**60.9**	**79.9**
CoLLM	MTCIR (ours)	**52.5**	**73.4**	58.2	**76.3**	**60.9**	79.4

the best overall results, outperforming both CoVR2 and MagicLens, despite utilizing a smaller fine-tuned dataset.

Qualitative results. The performance of CoLLM after fine-tuning with our MTCIR on the Refined CIRR and Fashion-IQ benchmarks is presented in Fig. 14 and Fig. 15. The original modification texts are often ambiguous, lacking specific details needed to identify the correct target image. With refined modification texts, our model achieves superior results on both datasets. The new texts remain concise but provide more useful information, enabling the model to perform better.

Table 22. Prompt structure to generate modification texts in MTCIR.

System	You are a language assistant that helps to generate the modification text between two image captions.
Prompt	Generate the modified text for the following pair of image captions: Caption 1: [CAPTION 1] Caption 2: [CAPTION 2] <instruction> You need to answer in both forward, changes from image 1 to image 2, and backward, changes from image 2 to image 1, directions. The definition of each category and examples are as follows: 1. [CATEGORY ID 1]: [CATEGORY DEFINITION 1] <example> Caption 1: [CAPTION EXAMPLE 1] Caption 2: [CAPTION EXAMPLE 2] Forward: [FORWARD EXAMPLE] Backward: [BACKWARD EXAMPLE] </example> ... 6. [CATEGORY ID 6]: [CATEGORY DEFINITION 6] ... The text needs to be concise and details as you can see the images, not as you are reading the text. You should not add words "details, specific, description" to the text. Here are some bad examples: <example> [BAD EXAMPLES] </example> </instruction> One category can has multiple changes. For each change, you need to write one short sentence less than 20 words to describe the change. You need to answer all changes in the json format. Here is an example of the correct format: {"forward": [{"category": "number_change", "text": "modified text"},...],"backward": [{"category": "number_change","text": "modified text"},...]}

Table 23. Prompt regenerating new modification texts for ambiguous samples in CIRR and Fashion-IQ.

System	You are the vision language bot that helps to generate the modification text given the reference image and the target image.
Prompt	[REFERENCE IMAGE][TARGET IMAGE] You are given the reference image and the target image. The original modification text: "[MODIFICATION TEXT]" is bad and does not have enough details to find the target image. These are some examples of the modification text: <example> [GOOD SAMPLES] </example> Generate three new modification texts following the instruction below: <instruction> 1. Understand the image content of the reference image (the first image). 2. Understand the image content of the target image (the second image). 3. text1: generate a short modification based on the original modification text with more specific details about the main information in the target image. It can be objects added or removed, colors, shapes or any other details. 4. text2: add one more detail to the text1 without removing any information. It can be the information about the relationship between the objects in the target image, the background information. 5. text3: add one more detail to the text2 without removing any information. It can be the view different from the reference image, any other details that is not in the first two texts. 7. Answer in json format {"text1": "new text 1", "text2": "new text 2", "text3": "new text 3"}. </instruction> Again, note that the modification text should be short and concise.

Table 24. Prompt validate sample ambiguity in CIRR and Fashion-IQ.

System	`You are the vision language bot that helps to find the target image given reference image and modification text.`

Prompt	`[REFERENCE IMAGE][CANDIDATE IMAGES]`
	`You are given the reference image and the candidate images. From the reference image and the modified text "[MODIFICATION TEXT]", find the best matched target image following the instruction below:`
	`<instruction>`
	`1. Understand the image content and the modification text.`
	`2. For each image in the candidate images, understand the image content.`
	`3. Find the best matched target image that matches the modification text.`
	`4. If there are two or more target images that are equally matched, answer -1.`
	`5. If the target image is not in the candidate images, answer -1.`
	`6. If the target image is in the candidate images, answer the index of the target image in the candidate images from left to right from 0 to 3.`
	`7. Answer in json format {"answer": target image index, "explain": give the reason for each unmatched image}.`
	`</instruction>`

Figure 10. "Good" samples kept in the Refined CIRR (left) and Fashion-IQ (right). The original modification text correctly highlights the different between target and most similar images.

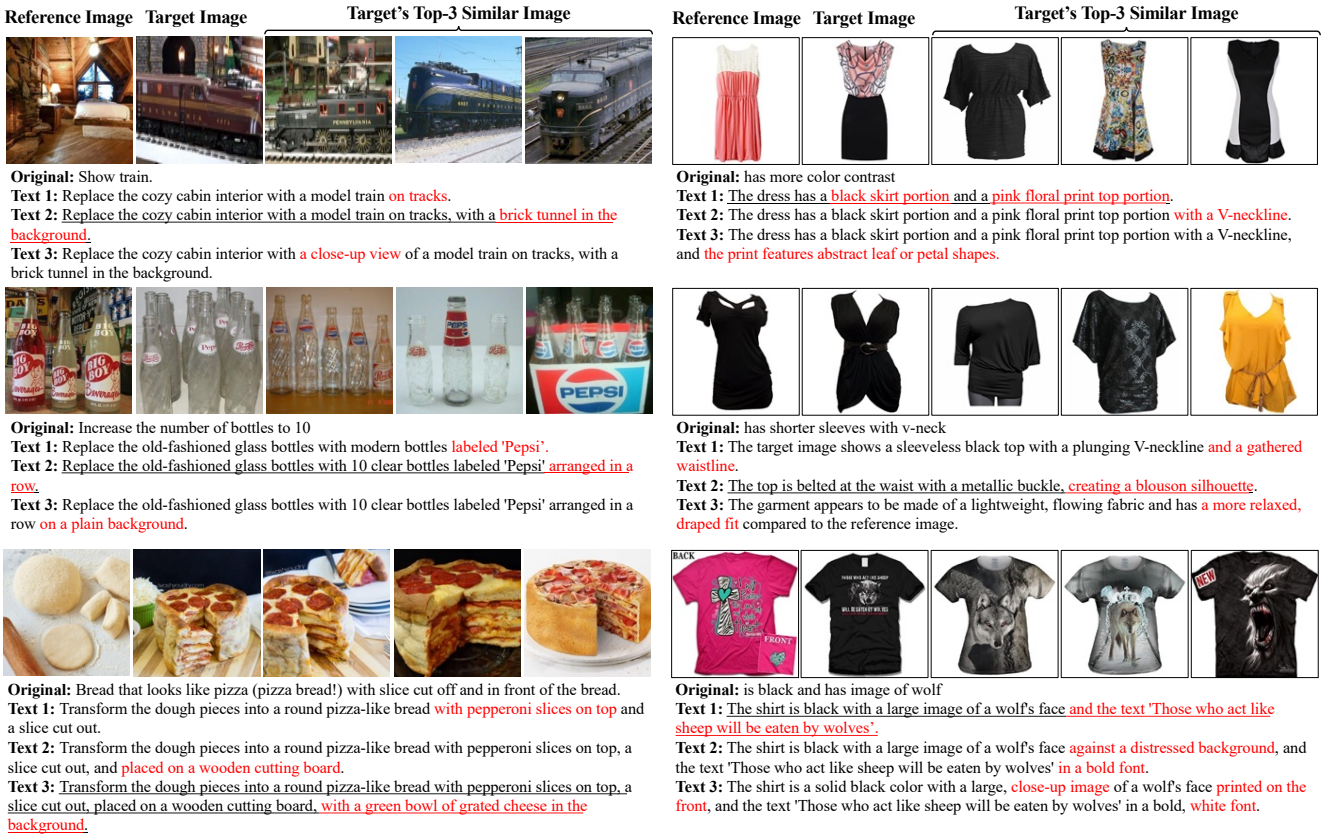


Figure 11. "Bad" samples with re-generated text in the Refined CIRR (left) and Fashion-IQ (right). The underlined is the selected modification text to replace the original. Red highlights the adding detail from the original to finest Text.

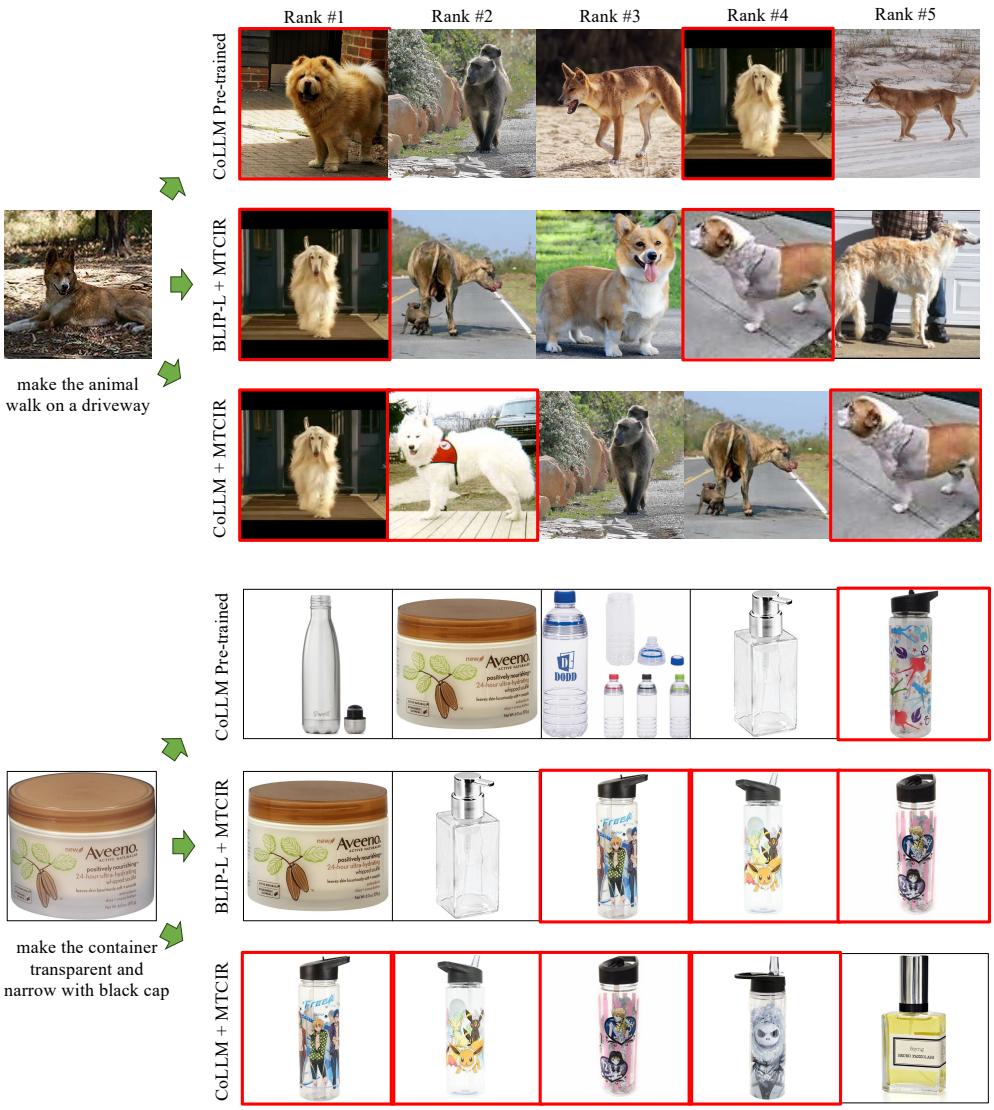


Figure 12. Retrieval results of Pre-trained CoLLM,BLIP-L fine-tuned on MTCIR (BLIP-L + MTCIR) and CoLLM fine-tuned on MTCIR (CoLLM + MTCIR) on CIRR Test set. Red highlights potential correct results (since we do not have the ground-truth on that test set).

Figure 13. Retrieval results of Pre-trained CoLLM, BLIP-L fine-tuned on MTCIR (BLIP-L + MTCIR) and CoLLM fine-tuned on MTCIR (CoLLM + MTCIR) on Fashion-IQ Validation set. Red highlights the ground-truth.

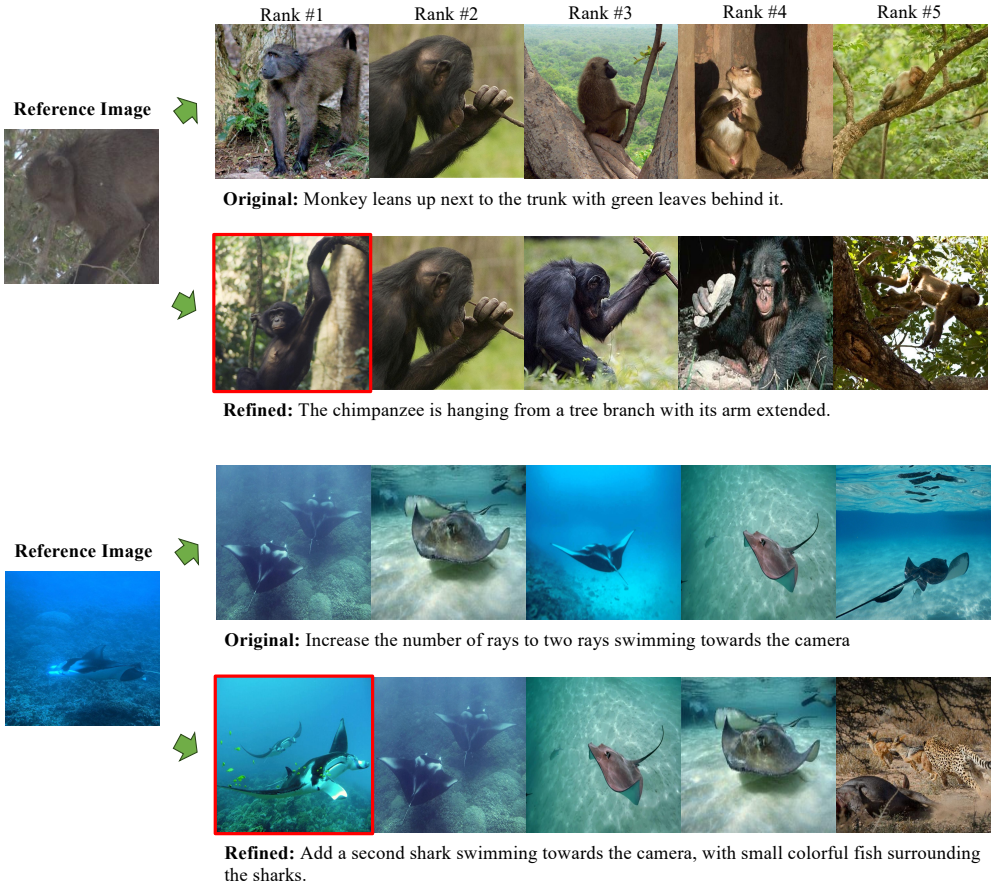


Figure 14. Retrieval results of CoLLM fine-tuned on MTCIR on original and Refined CIRR validation set. Red highlights the ground-truth. The new modification text helps the model to find the correct target images.

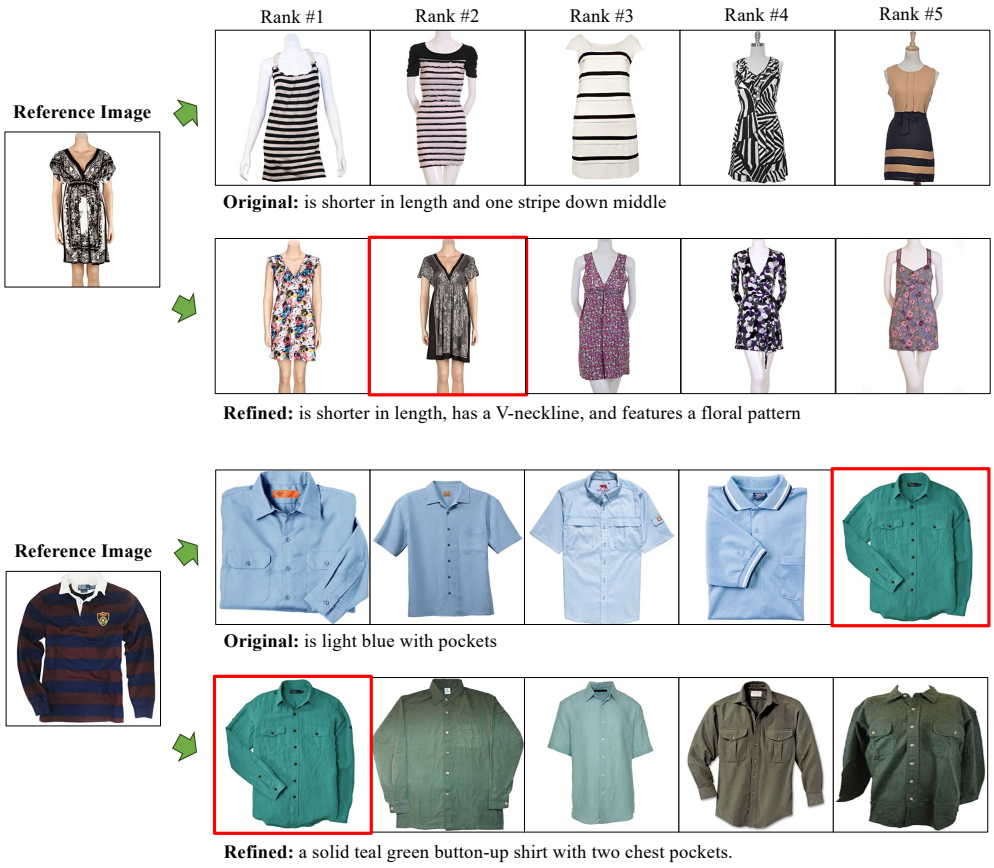


Figure 15. Retrieval results of CoLLM fine-tuned on MTCIR on original and Refined Fashion-IQ validation set. Red highlights the ground-truth. The new modification text with more details helps the model to find the correct target images.