ST-Bridge: Bridging Sketch and Text with Large Language Models for Coarse-to-Fine Image Retrieval

Haoxiang Hu[1,2,*] **Cangjun Gao**[1,2,*]
Zhengming Zhang[4,*] **Yaxian Shan**[1,2] **Zeyuan Huang**[1,2] **Ran Zuo**[5]
Qiang He[1,2] **Qingkun Li**[1] **Xiaoming Deng**[1,2,3,†] **Cuixia Ma**[1,2,3,†]
Yu-Kun Lai[6] **Yong-Jin Liu**[7] **Hongan Wang**[1,2]

[1] Institute of Software, Chinese Academy of Sciences, Beijing, China
[2] University of Chinese Academy of Sciences, Beijing, China
[3] Key Laboratory of System Software (Chinese Academy of Sciences)
[4] China Mobile Jiutian Artificial Intelligence Technology (Beijing) Co., Ltd
[5] Beijing Jiaotong University, Beijing, China
[6] Cardiff University, Cardiff, UK [7] Tsinghua University, Beijing, China

Abstract

Sketch-Text image retrieval aims to find natural images using both free-hand sketches and text descriptions. Although these two modalities provide complementary information, the task remains challenging because of their large representational differences and the difficulty of matching fine details in complex scenes. Most existing methods rely on simple feature fusion or global alignment, which often fails to preserve important discriminative cues and leads to inaccurate retrieval results. To tackle these issues, we propose ST-Bridge, a high-precision sketch-text image retrieval framework that enables reliable coarse-to-fine matching through an effective semantic bridging mechanism. Specifically, we introduce multimodal LLM as a semantic bridge to generate sketch representations that are closer to the textual semantic space. We further enhance these representations and the original sketch features via gated attention and feature normalization, substantially reducing the sketch–text modality gap. Building upon this, we design a coarse-to-fine alignment strategy that supports accurate sketch–text image retrieval. Extensive experiments on scene-level sketch–text image retrieval benchmarks demonstrate that ST-Bridge significantly outperforms existing methods and achieves state-of-the-art performance, validating the effectiveness of large-model-based semantic bridging for cross-modal retrieval tasks.

1 Introduction

Image retrieval is a critical field of Computer Vision research, aiming to retrieve images that meet specified requirements from vast repositories of image data. Existing research has mainly focused on Text-based Image Retrieval (TBIR) and Sketch-based Image Retrieval (SBIR) [26, 21, 22], where text provides high-level semantic information, while sketches offer structural cues that are more effective for capturing fine-grained shapes and layouts [28].

Recent studies have explored Text+Sketch-based Image Retrieval (TSBIR) by jointly leveraging the complementary information from sketches and textual descriptions, achieving encouraging progress [28, 27, 5, 36]. However, as shown in Figure 1, most existing methods still rely mainly on

* These authors contributed equally. † Corresponding authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

direct feature fusion or global-level feature alignment and ignore substantial differences between sketch and text modalities. Moreover, they generally lack discrimination and emphasis on critical fine-grained information, resulting in imprecise fine-grained feature alignment. Consequently, discriminative and important details are unnecessarily weakened or lost. These issues are particularly pronounced in complex scene-level retrieval tasks, ultimately leading to significant discrepancies between the retrieved results and the query intentions. As a result, fine-grained text–image alignment is often inaccurate. For instance, the word “girl" was wrongly aligned with the “kite" area - this led to the sketch and text failing to achieve precise fine-grained matching, thereby affecting the retrieval accuracy.

To address these issues, we propose **ST-Bridge**, which establishes an effective semantic bridge between sketches and text to explicitly reduce the modality gap, as shown in Figure 1. We propose to distill the scene understanding and semantic reasoning ability of a closed-source LLM into an LLM-based Semantic Sketch Encoder (LSSE), by transferring knowledge from GPT-4o [25] to a fine-tunable Qwen3-VL-8B model [1]. Given a sketch alone, LSSE produces semantic token hidden states that are better aligned with the textual semantic space. We then inject these semantics into ViT sketch tokens via feature normalization and gated attention, reducing the sketch–text modality gap. Built upon improved cross-modal alignment, our framework supports coarse-to-fine retrieval: global alignment captures overall scene semantics for coarse-grained retrieval, while the Token Selector and Fine-grained Feature Fusion modules enable fine-grained retrieval. Experiments on scene-level sketch–text image retrieval benchmarks show that our method achieves state-of-the-art performance, validating ST-Bridge’s effectiveness for cross-modal retrieval. The main contributions of this work are listed as follows:

• We propose **ST-Bridge**, a high-precision sketch–text image retrieval framework that significantly outperforms existing methods on standard benchmarks and achieves state-of-the-art performance.

• We introduce an **LLM-based Semantic Sketch Encoder (LSSE)** for sketch–text semantic bridging. Specifically, we construct a distillation dataset consisting of 17,115 GPT-4o-generated supervision texts and perform distillation learning. We then selectively inject the distilled semantic information into ViT sketch features via **feature normalization** and **gated attention**, reducing the sketch–text modality gap and providing a strong foundation for fine-grained alignment.

• We further design a **coarse-to-fine** alignment and matching scheme, in which global alignment captures coarse-grained semantic correspondence between fused sketch–text queries and images, while local alignment, guided by salient token selection, learns fine-grained correspondences between textual phrases and sketch details, improving retrieval accuracy in complex scenes.

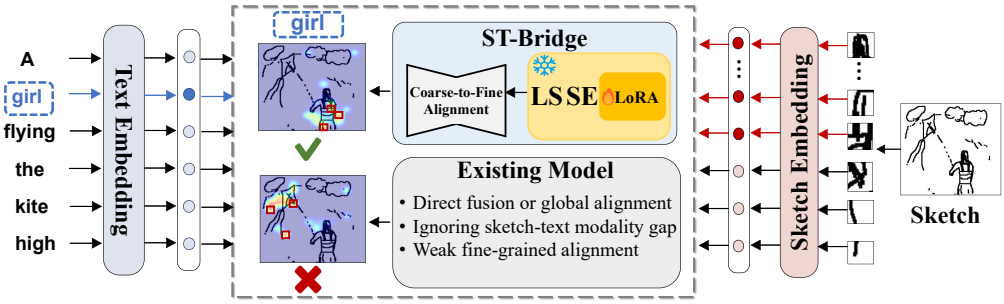

Figure 1: Comparison between existing methods and **ST-Bridge** in sketch–text feature alignment. Taking the word “girl” as an example, we highlight the corresponding sketch patch with a red box and visualize the attended sketch regions via attention maps. Existing methods incorrectly align “girl” to the “kite” region, whereas **ST-Bridge** achieves accurate fine-grained alignment.

2 Related Work

2.1 Sketch- and Text-Based Image Retrieval

Text-based image retrieval (TBIR) aligns linguistic queries with visual content by learning a unified cross-modal embedding space [17, 9, 19, 26], and further enhances fine-grained alignment through

Transformer architectures and large-scale pretraining [24, 20, 33]. Sketch-based image retrieval (SBIR) uses sketches as queries and matches real images via structural cues, extending from object-level to scene-level retrieval [21, 22]. Recent approaches increasingly employ graph-based modeling to capture instance relations and spatial layouts, while also addressing sketch abstraction and open-vocabulary semantics [13, 2, 18]. To mitigate ambiguities arising from a single modality, multimodal retrieval leverages complementary cues, such as compositional queries combining text with reference images [31, 7, 16], or augmenting sketch queries with semantic labels or textual descriptions [30, 6]. Compared to TBIR or SBIR, research on joint text+sketch-based image retrieval (TSBIR) remains relatively limited; its core idea is to integrate the structural constraints of sketches with the semantic expressiveness of text. Existing work explores joint sketch–image–text embeddings and benchmarks [27, 5, 12], as well as generative alignment paradigms that construct shared latent spaces using diffusion models [36].

2.2 LLM-augmented Sketch+Text Image Retrieval

The recent rise of LLMs offers new opportunities for sketch+text retrieval, especially in query understanding, data construction, and interactive search. LLMs can serve as semantic "interpreters" on the text side, rewriting underspecified queries, expanding attributes/interactions that are difficult to sketch, and bridging vocabulary gaps (a motivation aligned with composite sketch+text retrieval settings [12]). LLMs enable scalable supervision for composed retrieval by synthesizing training triplets and richer modification texts from image–caption corpora. CoLLM [15] exemplifies this direction by generating triplets on-the-fly and leveraging LLMs to enhance multimodal fusion and representation learning for composed image retrieval. Instruction-tuned sketch-language models begin to directly connect sketches, images, and natural language instructions. O3SLM [14] constructs large-scale sketch–image–instruction data and demonstrates improved sketch understanding and retrieval-related capabilities through pretraining and instruction tuning. Finally, LLMs naturally support conversational retrieval interfaces: sketch-based conversational image search systems [3] integrate an LLM to manage dialog state and refinement, while coupling it with powerful generative or vision-language backbones to iteratively narrow down results. Overall, LLMs provide high-level reasoning and language grounding that complements sketch structure, suggesting a promising direction for robust and user-friendly sketch+text image retrieval.

3 Methodology

3.1 Stage I: Semantic Distillation

In Stage I, we aim to distill the scene understanding and semantic reasoning capability of a closed-source LLM into a lightweight vision–language model, yielding an *LLM-based Semantic Sketch Encoder* (LSSE) for Stage II. Given a sketch, LSSE outputs semantic-rich token-level hidden states that serve as sketch features for subsequent retrieval and alignment. We adopt a teacher–student distillation framework, with GPT-4o as the teacher and Qwen3-VL-8B as the student. During training, the teacher takes the sketch S and its paired natural image V to resolve sketch ambiguity, and generates sketch-consistent supervision text Y. The supervision contains (i) scene/instance descriptions strictly grounded in visible sketch content, and (ii) high-level semantics inferred from sketch structure (e.g., attributes, poses, background, and potential events/actions); detailed prompt design is provided in Appendix A. We construct a distillation set of **17,115** GPT-4o-generated supervision texts from SFSD, FSCOCO, and SketchyCOCO for training (see Supplementary Materials).The student is built on Qwen3-VL-8B and fine-tuned via LoRA, with a lightweight Token Conv Adapter inserted into the visual branch while all other parameters remain frozen. The student, in contrast, is conditioned only on the sketch in both training and inference. Training is formulated as conditional text generation. For each sample, we concatenate a prompt sequence X_{prompt} and teacher-generated supervision text Y as input, and optimize the **Understanding-and-Reasoning Aligned Generation Loss (URA Loss)**:

$$L_{\text{URA}} = -\sum_{t=L_p+1}^{L_p+L_y} \log p_\theta\left(y_{t-L_p} \mid X_{\text{prompt}}, S, y_{<t-L_p}\right), \tag{1}$$

where $p_\theta(\cdot)$ denotes the student's conditional generation distribution, and L_p and L_y are the token lengths of X_{prompt} and Y, respectively.

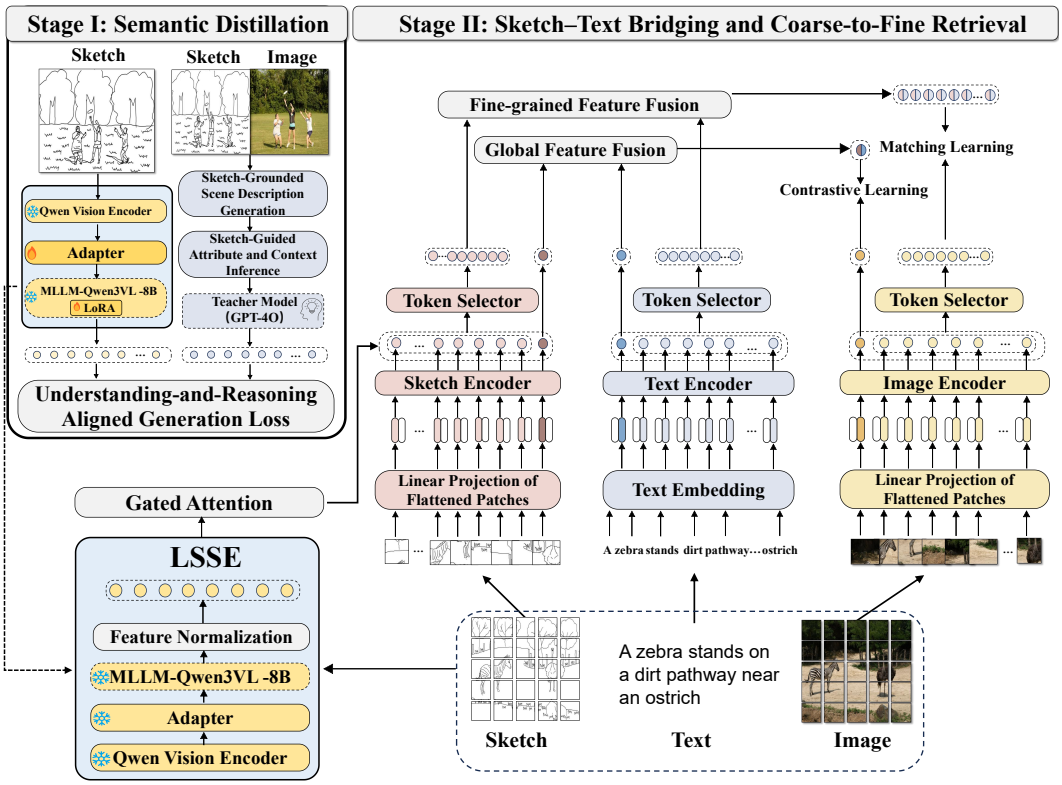

Figure 2: Overview of ST-Bridge. We adopt a two-stage framework for sketch-text image retrieval. **Stage I**: the scene understanding and semantic reasoning ability of GPT-4o is distilled into a fine-tunable Qwen3-VL-8B to obtain an *LLM-based Semantic Sketch Encoder* (LSSE). **Stage II**: LSSE bridges the sketch–text modality gap by first aligning the statistical distributions of semantic and sketch tokens via feature normalization, then selectively injecting semantic information into ViT sketch tokens through gated cross-attention. Building on these enhanced representations, we perform coarse-to-fine feature alignment and conduct image retrieval via contrastive learning with images.

3.2 Stage II: Sketch-Text Bridging and Coarse to Fine Retrieval

3.2.1 LLM-based Semantic Sketch Encoder (LSSE)

After Stage I, we freeze the distilled Qwen3-VL-8B and its Adapter parameters and use it directly as the LSSE in Stage II to generate sketch semantic token hidden states aligned with the textual semantic space. Given a sketch S, the frozen Qwen3-VL-8B produces semantic token hidden states $\{z_j\}_{j=1}^{N_z}$. Meanwhile, a ViT-based sketch encoder outputs patch-level sketch features $\{s_i^*\}_{i=1}^{N_s}$ that mainly preserve spatial structure and local geometric details (see Appendix A for encoder details). Because these representations come from heterogeneous encoders, their feature scales and statistics differ substantially; directly fusing them can destabilize training and weaken semantic injection. To mitigate this, we apply Feature Normalization to align semantic features to the sketch feature space:

$$\hat{z}_j = \frac{z_j - \mu(z)}{\sigma(z)}, \qquad \bar{z}_j = \hat{z}_j \odot \sigma(s^*) + \mu(s^*) \tag{2}$$

where $\mu(\cdot)$ and $\sigma(\cdot)$ denote the mean and standard deviation computed over all tokens and feature dimensions, respectively, and $\odot$ denotes element-wise multiplication. This normalization aligns the scale of semantic tokens with that of sketch patch features before fusion.

After normalization, we introduce a Gated Attention mechanism to selectively inject semantic information into sketch patch features, enhancing patches when semantics are beneficial while avoiding indiscriminate fusion that may overwrite spatial structure. Concretely, we perform cross-attention with each sketch patch token s_i^* as a query and the aligned semantic tokens $\{\bar{z}_j\}$ as keys

and values, producing a patch-specific semantic response:

$$\alpha_{ij} = \mathrm{Softmax}_j\left(\frac{(W_q s_i^*)^\top (W_k \bar{z}_j)}{\sqrt{d}}\right), a_i \quad = \sum_{j=1}^{N_z} \alpha_{ij} W_v \bar{z}_j \tag{3}$$

where W_q, W_k, and W_v are learnable projections and d is the feature dimension. This cross-attention captures fine-grained patch–token correspondences between sketch structure and semantic cues. Rather than conditioning the gate on potentially noisy patch-level attention responses, we compute a global semantic context vector via mean pooling and generate the patch-wise gating signal by combining the local sketch feature with this global context:

$$g_i = \sigma\Big(W_g \Big[\, s_i^*,\ \mathrm{MeanPool}(\{\bar{z}_j\}_{j=1}^{N_z}) \,\Big]\Big), \tag{4}$$

where W_g is a learnable projection and $\sigma(\cdot)$ denotes the Sigmoid function. This design provides a stable semantic prior reflecting the overall scene inferred from the sketch. Finally, we update sketch patch features via gated residual modulation followed by layer normalization:

$$s_i = \mathrm{LN}(s_i^* + g_i \odot a_i) \tag{5}$$

The resulting semantic-enhanced sketch features $\{s_i\}_{i=1}^{N_s}$ preserve the sketch encoder's spatial structure while being selectively strengthened by globally consistent semantic cues from LSSE.

3.2.2 Global Feature Alignment

With semantic-enhanced sketch tokens from the LSSE module, we first perform global alignment to enable coarse-grained retrieval. Given a text query and an image, we extract token-level features using pre-trained encoders (see Appendix A for details). The text encoder outputs $T = \{t_g; t_1, \ldots, t_{N_t}\}$ and the image encoder outputs $V = \{v_g; v_1, \ldots, v_{N_v}\}$, where t_g and v_g are the final-layer `[CLS]` embeddings and the remaining tokens are word-/patch-level features. For the sketch, we obtain semantic-enhanced patch tokens $\{s_i\}_{i=1}^{N_s}$ and summarize them into a global sketch representation s_g via mean pooling. We then fuse the global text and sketch representations by direct addition:

$$s_g = \mathrm{MeanPool}\big(\{s_i\}_{i=1}^{N_s}\big), \qquad f_g = \mathrm{Add}(t_g, s_g), \tag{6}$$

where f_g denotes the global feature of the fused sketch–text query F. Following CLIP [26], we optimize a batch-wise contrastive objective. For a mini-batch of size M, each triplet (T_m, S_m, V_m) forms a positive pair between the fused query F_m and its paired image V_m, while the remaining images (or fused queries) in the batch serve as negatives. We compute cosine similarity between a fused query and an image using their global features:

$$s(F_m, V_n) = (f_g^{(m)})^\top v_g^{(n)}. \tag{7}$$

The fusion-to-image and image-to-fusion matching distributions are defined as

$$y_{m\to n}^{f2v} = \frac{\exp(s(F_m, V_n)/\tau)}{\sum_{k=1}^{M} \exp(s(F_m, V_k)/\tau)}, \qquad y_{n\to m}^{v2f} = \frac{\exp(s(F_m, V_n)/\tau)}{\sum_{k=1}^{M} \exp(s(F_k, V_n)/\tau)}. \tag{8}$$

where τ is a learnable temperature parameter. Let $\hat{y}$ denote the one-hot ground-truth matching distribution (1 for the positive pair and 0 otherwise). The global alignment loss is defined as the symmetric cross-entropy:

$$L_g = \frac{1}{2}\Big(H(\hat{y}^{f2v}, y^{f2v}) + H(\hat{y}^{v2f}, y^{v2f})\Big). \tag{9}$$

where $\hat{y}^{v2f}(V)$ and $\hat{y}^{f2v}(F)$ represent the ground-truth one-hot similarity.

3.2.3 Token Selector

Although global feature alignment captures coarse-grained sketch/text–image semantics, it is insufficient for modeling fine-grained correspondences in complex scenes. Moreover, many local tokens are non-discriminative, introducing noise and unnecessary computation in cross-modal alignment. To address this, we introduce Token Selector, which preserves key information while selecting only salient

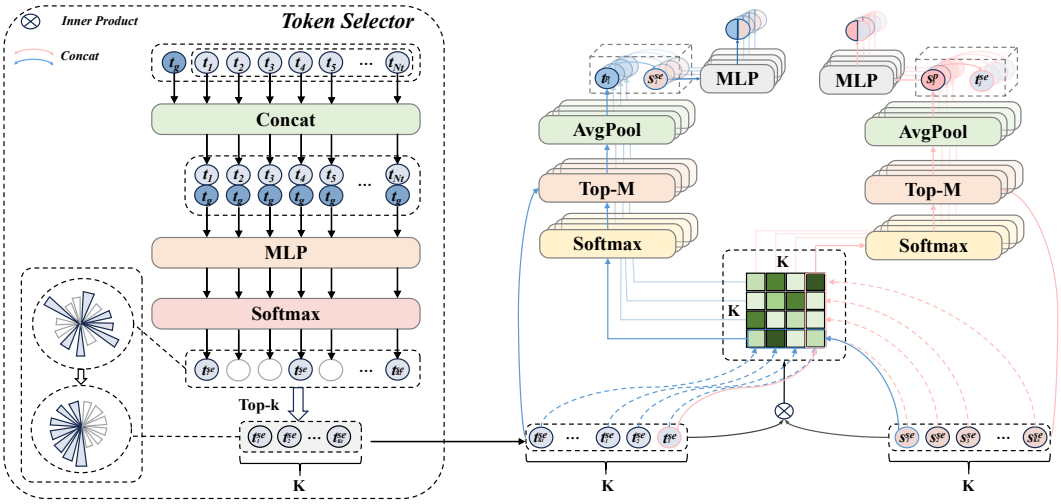

Figure 3: Token Selector and Fine-grained feature fusion.

tokens, improving fine-grained matching accuracy and reducing computational cost. The attention weight matrix encodes the correlation between the global token and local tokens [23, 29, 35, 32]. Consequently, we identify discriminative local tokens by sorting attention weights. As illustrated in Figure 3, we utilize a trainable module similar to the framework described in [23] to compute saliency scores across all local tokens. We then rank these tokens and select the top K based on their saliency scores. We repeat this token selection process across text, sketch, and image modalities. For illustration, we describe this process using text as a reference. The text feature $T = \{t_g; t_1, \ldots, t_{N_t}\}$, encoded by the Transformer, comprises a sequence of tokens. Firstly, we concatenate the global token t_g with all local tokens t_i to create a new token $\hat{t}_i = concat(t_g, t_i), 0 < i \leq N_t$. Subsequently, the concatenated tokens are fed into a multi-layer perceptron (MLP). Finally, a Softmax layer estimates the saliency score for each local token. The specific formulation is as follows:

$$Score = SoftMax(MLP(\hat{t})) \tag{10}$$

Based on the calculated score of each local token, we select the top K_t tokens with the highest scores for subsequent fine-grained feature fusion. The selected token sequence is represented as follows: $T^{se} = \{t_1^{se}, t_2^{se}, \ldots, t_{K_t}^{se}\}$. For the given sketch and image, we select K_s and K_v salient tokens, respectively. The selected token sequences are represented as: $S^{se} = \{s_1^{se}, s_2^{se}, \ldots, s_{K_s}^{se}\}$, $V^{se} = \{v_1^{se}, v_2^{se}, \ldots, v_{K_v}^{se}\}$.

3.2.4 Fine-grained Feature Fusion

To achieve fine-grained alignment between text and sketch, simple one-to-one word–patch matching is often unreliable in complex scenes. Considering that the relationship between textual descriptions and corresponding sketches is inherently many-to-many, we adopt a more practical strategy: for each word, we select the top K_{t2s} most relevant sketch regions, and for each sketch patch, we select the top K_{s2t} most relevant words. For each (T, S) text-sketch pair, we employ token sequences T^{se} and S^{se} selected by the Token Selector. This process, illustrated in Figure 3, aims to achieve fine-grained alignment between text and sketch. First, we compute the cosine similarity $Sim(T, S) = TS^{\top}$ between T^{se} and S^{se}, then obtain similarity scores through a Softmax layer. For text-based fine-grained feature fusion, for each word T_i^{se}, we select the top K_{sp} sketch blocks from the ith row of $Sim(T, S)$ based on their similarity scores. These K_{sp} sketch blocks are subsequently aggregated using an average pooling layer to generate a new sketch block s_i^p, thereby forming a new matching pair (t_i^{se}, s_i^p).

$$s_i^p = AvgPool(TopK(Softmax(Sim(T^{se}, S^{se}))[i, :])) \tag{11}$$

We perform fine-grained feature alignment for each salient local token in the text, yielding K_t word-sketch block matching pairs. For each matching pair $\{(t_1^{se}, s_1^p), (t_2^{se}, s_2^p), \ldots, (t_{K_t}^{se}, s_{K_t}^p)\}$, the features are concatenated and passed through a multi-layer perceptron (MLP) to generate a new

text-based text-sketch fine-grained fusion feature t_i^{ts}, represented as follows:

$$t_i^{ts} = MLP(Concat(t_i^{se}, s_i^p)) \tag{12}$$

Ultimately, the text-based fine-grained feature sequence $T^{ts} = \{t_1^{ts}, t_2^{ts}, \ldots, t_{K_t}^{ts}\}$. Similarly, we obtain the sketch-based fine-grained feature sequence $S^{st} = \{s_1^{st}, s_2^{st}, \ldots, s_{K_s}^{st}\}$.

3.2.5 Fine-grained Feature Matching

To better align the fused text-sketch fine-grained features with the image fine-grained features, we use the similarity scores from Equation 8 to select hard examples within the mini-batch. Specifically, we use the [CLS] feature extracted from the final layer of a six-layer cross-attention module to estimate the likelihood of correspondence between sample pairs, denoted as $p(T^{ts}, V^{se})$ and $p(S^{st}, V^{se})$. The loss function is defined as:

$$L_f = \frac{1}{2}\mathbb{E}_{(V,T,S)\sim D}[H(\hat{p}, p(T^{ts}, V^{se})) + H(\hat{p}, p(S^{st}, V^{se}))] \tag{13}$$

where $\hat{p}$ denotes the two-dimensional one-hot ground-truth label and H the cross-entropy loss.

4 Experiments

4.1 Datasets and Evaluation Metric

We evaluate our model on three scene-level datasets providing aligned text-sketch-image triplets: SFSD, SketchyCOCO, and FSCOCO. SFSD [34] contains 12,100 sketches (9,100 train/3,000 test) with instance-level annotations. For SketchyCOCO [11], following the data preprocessing methodology in SceneSketcher [21], we filter out samples with sparse foreground objects, resulting in 1,015 training and 210 testing tuples. Regarding FSCOCO [4] (7,000 train/3,000 test), we adopt the original MS-COCO captions for the retrieval task. We use Recall at k (R@k) as the retrieval metric, measuring the fraction of queries whose target image appears in the top-k retrieved results.

4.2 Implementation Details

Our ST-Bridge model employs ViT [8] and a 12-layer Transformer for visual and textual encoding, respectively, both initialized with pre-trained weights. Visual inputs are resized to 224×224 and text length is capped at 30 words. Key parameters for the Token Selector and fusion module are set as $\{K_t, K_s, K_v\} = \{4, 20, 20\}$ and $\{K_{sp}, K_{tp}\} = \{3, 2\}$. We use the `ReduceLROnPlateau` scheduler and apply early stopping with 20 epochs of patience to mitigate overfitting. Training and inference are performed on a single NVIDIA A40 GPU, yielding an average retrieval latency of 517.3 ms.

4.3 Comparisons with State-of-the-Art Models

Table 1 compares ST-Bridge with representative text-only, sketch-only and hybrid baselines on SFSD, SketchyCOCO and FSCOCO. ST-Bridge achieves the best results across all datasets and metrics. On SFSD, it improves R@1 from 85.10% (SceneDiff) to 92.81%, while also yielding strong gains at R@5/R@10 (98.95%/99.64%). On SketchyCOCO, ST-Bridge attains 72.38% R@1, markedly surpassing SceneDiff (61.02%) and SceneTrilogy (39.50%), and remains consistently ahead at higher recalls, indicating a more reliable ranking. The advantage persists on FSCOCO, where ST-Bridge reaches 61.29% R@1 versus 46.37% for the best prior method. Overall, the results suggest that semantic bridging and coarse-to-fine alignment substantially narrow the text–sketch gap and lead to more accurate scene-level retrieval.

Method	Input	SFSD			SketchyCOCO			FSCOCO		
		R@1	**R@5**	**R@10**	**R@1**	**R@5**	**R@10**	**R@1**	**R@5**	**R@10**
CLIP	T	16.1	33.57	43.03	25.24	54.29	65.71	9.63	22.52	30.53
CLIP-FT	T	19.32	41.55	56.49	29.66	60.15	71.82	13.42	28.05	36.72
SketchyScene	S	60.01	72.83	80.12	27.51	54.23	74.1	22.85	40.9	51.19
SceneSketcher	S	69.58	82.29	86.4	31.9	66.71	86.2	-	-	-
TASK-former	T+S	78.52	93.13	95.63	38.04	58.18	69.24	40.27	62.65	75.86
SceneTrilogy	T+S	-	-	-	39.5	-	88.7	25.7	-	55.2
SceneDiff	T+S	85.1	96.43	98.85	61.02	83.33	92.76	46.37	67.71	78.52
Ours	**T+S**	**92.81**	**98.95**	**99.64**	**72.38**	**92.38**	**97.14**	**61.29**	**82.9**	**89.15**

Table 1: Comparison with state-of-the-art sketch–text image retrieval methods on SFSD, SketchyCOCO and FSCOCO. **T** denotes a text query, **S** a sketch query, and **T+S** the joint text–sketch query. CLIP-FT denotes a fine-tuned CLIP model. "-" indicates that results are not reported.

A	B	C	D	SFSD		SketchyCOCO		FSCOCO	
				R@1	R@10	R@1	R@10	R@1	R@10
✓				83.8	98.29	50.0	91.43	49.17	82.25
✓	✓			85.56	98.87	50.48	92.38	55.7	85.98
✓	✓	✓		90.74	99.53	69.05	96.67	58.42	87.83
✓	✓	✓	✓	**92.81**	**99.64**	**72.38**	**97.14**	**61.29**	**89.15**

Table 2: Ablation study of ST-Bridge on SFSD, SketchyCOCO and FSCOCO. A: Global Feature Alignment. B: Token Selector with fine-grained feature fusion and matching. C: LSSE with semantic injection via plain cross-attention (without gating). D: LSSE with gated attention.

Dataset	R@k	Text-p_{mask}			Sketch-p_{mask}		
		0.1	0.3	0.5	0.1	0.3	0.5
SFSD	R@1	92.15	91.57	90.72	71.68	37.13	12.87
	R@5	98.93	98.54	97.93	89.23	59.17	27.69
	R@10	99.59	99.34	99.12	92.87	68.18	35.54
Sketchy COCO	R@1	65.71	62.38	59.05	46.19	33.33	24.76
	R@5	91.90	88.57	87.14	80.95	67.62	57.14
	R@10	96.67	96.19	92.38	89.52	78.10	66.67
FSCOCO	R@1	57.32	56.87	54.92	51.12	42.06	28.97
	R@5	80.36	79.68	78.94	74.95	64.12	49.36
	R@10	87.64	86.08	84.27	82.11	73.45	60.08

Table 3: Retrieval performance of ST-Bridge under random masking across datasets.

4.4 Ablation Study

We conduct ablation experiments on SFSD, SketchyCOCO and FSCOCO to analyse the contribution of each key component in ST-Bridge. As shown in Table 2, we progressively introduce four modules: A) Global Feature Alignment, B) Token Selector with fine-grained feature fusion and matching, C) LSSE with plain cross-attention for semantic injection (i.e., without gating), and D) LSSE with gated attention. Starting from the baseline with only Global Feature Alignment (A), the model already achieves competitive performance, indicating that coarse-grained sketch–text fusion provides a strong retrieval prior. Incorporating the fine-grained pipeline (B) consistently improves results across all benchmarks, demonstrating the importance of salient token selection and local cross-modal matching for complex scenes. Adding LSSE without gated fusion (C) leads to a substantial performance gain, particularly on SketchyCOCO (R@1 from 50.48% to 69.05%), confirming that LLM-derived semantic sketch representations effectively narrow the sketch–text semantic gap. Finally, introducing gated feature fusion (D) further boosts performance on all datasets, yielding the best results overall. This suggests that selectively injecting semantic information is crucial for preserving sketch structure while leveraging high-level semantics.

4.5 Retrieval with Incomplete Query Input

We further assess robustness by randomly masking a proportion p_{mask} of either text tokens or sketch patches only at inference time, without any retraining on masked inputs. Table 3 reports results on SFSD, SketchyCOCO, and FSCOCO. ST-Bridge remains notably resilient to incomplete text: even with p_{mask}=0.5, the drop is modest (e.g., SFSD R@1 decreases from 92.15% to 90.72%), suggesting that retrieval is driven by globally consistent semantics rather than isolated keywords. In contrast, masking sketch patches degrades performance more sharply, particularly at higher ratios, which is



Figure 4: Qualitative retrieval results on SFSD, SketchyCOCO and FSCOCO using joint text–sketch queries. For each query, the top retrieved images are shown, with the ground-truth image highlighted.

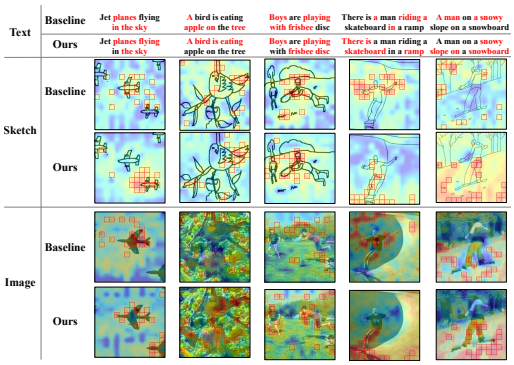

Figure 5: Visualization of token selection across text, sketch and image modalities. We compare the baseline without LSSE (A+B) and the full model (Ours).



Figure 6: Visualisation of fine-grained feature fusion between selected text tokens and sketch tokens.

expected given the sensitivity of structural cues to occlusion. Overall, the model generalises well to partially missing queries despite being trained on complete inputs.

4.6 Qualitative Results

Qualitative Retrieval Results with Text–Sketch Queries. Figure 4 presents qualitative retrieval examples on SFSD, SketchyCOCO and FSCOCO using joint text–sketch queries. For each query, we visualise the top retrieved images, with the ground-truth image highlighted. Overall, the retrieved results are semantically consistent with the input queries, and the ground-truth images are ranked highly across datasets. For instance, in the fourth example, the sketch roughly indicates two horses but fails to capture accurate pose and orientation, while the text compensates for these deficiencies and enables correct retrieval. Additional qualitative results and comparisons with sketch-only and text-only queries are provided in Appendices E and C.

Visualisation of Token Selection across Modalities. Figure 5 visualises the tokens selected by the Token Selector across text, sketch and image modalities. We compare a baseline without LSSE (A+B in the ablation study) against the full model. For text queries, the full model prioritises semantically salient tokens, such as *skateboard*, *slope*, and *snowboard*. For sketch and image modalities, the full model selects tokens that show stronger semantic alignment with the text. For example, in the third column, multiple human figures are highlighted in response to the plural noun *boys*; in the fourth column, attention is focused on the skateboard mentioned in the text; and in the fifth column, tokens near the slope region are emphasised. These observations indicate that LSSE guides token selection towards text-consistent and semantically meaningful regions across modalities.

Analysis of Fine-grained Feature Fusion. Figure 6 further analyses fine-grained feature fusion by visualising correspondences between selected text tokens and sketch tokens. For object-related nouns (e.g., *girl* and *woman*), the full model establishes more accurate associations with the corresponding sketch regions than the baseline. For action-related words, such as *playing* and *standing*, the matched sketch tokens capture relevant body parts and contextual regions, including limbs and the ground. In the final examples, the baseline fails to select critical textual cues, whereas the proposed model successfully links these tokens to appropriate sketch regions. The fusion remains effective for both salient foreground objects (*girl*, *woman*) and background elements (*tree*).

5 Conclusion

In this paper, we propose **ST-Bridge**, a two-stage framework for scene-level sketch–text image retrieval. In the first stage, we distill GPT-4o's scene understanding and semantic reasoning abilities into a lightweight LSSE built on Qwen3-VL-8B, using 17,115 GPT-4o-generated supervision texts. In the second stage, LSSE generates sketch-based semantic tokens aligned with the textual semantic space. By normalizing feature distributions and injecting semantic information into ViT sketch tokens via gated cross-attention, ST-Bridge obtains sketch representations that retain structural cues while enhancing semantic discriminability. It further combines global contrastive alignment with salient-token selection and local matching, enabling coarse-to-fine retrieval from global ranking to phrase-to-detail correspondence. Experiments on SFSD, SketchyCOCO, and FSCOCO show that ST-Bridge significantly outperforms existing methods.

Acknowledgements

This work was supported by the Space Medical Experiment Project of CMSP (Grant Nos. HYZHXMR01005), the National Natural Science Foundation of China (Grant Nos. 62272447, 52502518, 62473356, and 62461160309), the Basic Research Program of the Institute of Software, Chinese Academy of Sciences (ISCAS-JCMS-202404), the CAS Major Project (RCJJ-145-24-14), the Beijing Natural Science Foundation (Grant Nos. L2608050, L242060, and 4254109), the Chongqing Industry and Information Technology R&D Program under Grant YJX-2025001001003, and the Engineering and Physical Sciences Research Council (Grant No. EP/Y028805/1)

References

[1] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025. URL `https://arxiv.org/abs/2511.21631`.

[2] Ahmed Bourouis, Judith E. Fan, and Yulia Gryaditskaya. Open vocabulary semantic scene sketch understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4176–4186, June 2024.

[3] Daniel D Braghis and Haiming Liu. Conversational image search: A sketch-based approach. In *Proceedings of the 2024 International Conference on Multimedia Retrieval*, pages 1265–1269, 2024.

[4] Pinaki Nath Chowdhury, Aneeshan Sain, Ayan Kumar Bhunia, Tao Xiang, Yulia Gryaditskaya, and Yi-Zhe Song. Fs-coco: Towards understanding of freehand sketches of common objects in context. In *European Conference on Computer Vision*, pages 253–270. Springer, 2022.

[5] Pinaki Nath Chowdhury, Ayan Kumar Bhunia, Aneeshan Sain, Subhadeep Koley, Tao Xiang, and Yi-Zhe Song. Scenetrilogy: On human scene-sketch and its complementarity with photo and text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10972–10983, 2023.

[6] Sounak Dey, Anjan Dutta, Suman K Ghosh, Ernest Valveny, Josep Lladós, and Umapada Pal. Learning cross-modal deep embeddings for multi-object image retrieval using text and sketch. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 916–921. IEEE, 2018.

[7] Haoyu Dong, Ze Wang, Qiang Qiu, and Guillermo Sapiro. Using text to teach image retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1643–1652, 2021.

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[9] Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. Vse++: Improving visual-semantic embeddings with hard negatives. *arXiv preprint arXiv:1707.05612*, 2017.

[10] Philip Gage. A new algorithm for data compression. *C Users Journal*, 12(2):23–38, 1994.

[11] Chengying Gao, Qi Liu, Qi Xu, Limin Wang, Jianzhuang Liu, and Changqing Zou. Sketchycoco: Image generation from freehand scene sketches. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5174–5183, 2020.

[12] Prajwal Gatti, Kshitij Parikh, Dhriti Prasanna Paul, Manish Gupta, and Anand Mishra. Composite sketch+ text queries for retrieving objects with elusive names and complex interactions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 1869–1877, 2024.

[13] Ce Ge, Jingyu Wang, Qi Qi, Haifeng Sun, Tong Xu, and Jianxin Liao. Scene-level sketch-based image retrieval with minimal pairwise supervision. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 650–657, 2023.

[14] Rishi Gupta, Mukilan Karuppasamy, Shyam Marjit, Aditay Tripathi, and Anirban Chakraborty. O3slm: Open weight, open data, and open vocabulary sketch-language model. *arXiv preprint arXiv:2511.14368*, 2025.

[15] Chuong Huynh, Jinyu Yang, Ashish Tawari, Mubarak Shah, Son Tran, Raffay Hamid, Trishul Chilimbi, and Abhinav Shrivastava. Collm: A large language model for composed image retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3994–4004, June 2025.

[16] Surgan Jandial, Pinkesh Badjatiya, Pranit Chawla, Ayush Chopra, Mausoom Sarkar, and Balaji Krishnamurthy. Sac: Semantic attention composition for text-conditioned image retrieval. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4021–4030, 2022.

[17] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3128–3137, 2015.

[18] Subhadeep Koley, Ayan Kumar Bhunia, Aneeshan Sain, Pinaki Nath Chowdhury, Tao Xiang, and Yi-Zhe Song. How to handle sketch-abstraction in sketch-based image retrieval? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16859–16869, June 2024.

[19] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 201–216, 2018.

[20] Gen Li, Nan Duan, Yuejian Fang, Ming Gong, and Daxin Jiang. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11336–11344, 2020.

[21] Fang Liu, Changqing Zou, Xiaoming Deng, Ran Zuo, Yu-Kun Lai, Cuixia Ma, Yong-Jin Liu, and Hongan Wang. Scenesketcher: Fine-grained image retrieval with scene sketches. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIX 16*, pages 718–734. Springer, 2020.

[22] Fang Liu, Xiaoming Deng, Changqing Zou, Yu-Kun Lai, Keqi Chen, Ran Zuo, Cuixia Ma, Yong-Jin Liu, and Hongan Wang. Scenesketcher-v2: Fine-grained scene-level sketch-based image retrieval using adaptive gcns. *IEEE Transactions on Image Processing*, 31:3737–3751, 2022.

[23] Yuqi Liu, Pengfei Xiong, Luhui Xu, Shengming Cao, and Qin Jin. Ts2-net: Token shift and selection transformer for text-video retrieval. In *European Conference on Computer Vision*, pages 319–335. Springer, 2022.

[24] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in Neural Information Processing Systems*, 32, 2019.

[25] OpenAI. Gpt-4o system card, 2024. URL `https://arxiv.org/abs/2410.21276`.

[26] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pages 8748–8763. PMLR, 2021.

[27] Patsorn Sangkloy, Wittawat Jitkrittum, Diyi Yang, and James Hays. A sketch is worth a thousand words: Image retrieval with text and sketch. In *European Conference on Computer Vision*, pages 251–267. Springer, 2022.

[28] Jifei Song, Yi-Zhe Song, Tony Xiang, and Timothy M Hospedales. Fine-grained image retrieval: the text/sketch input dilemma. In *BMVC*, page 7, 2017.

[29] Hui Su, Yue Ye, Zhiwei Chen, Mingli Song, and Lechao Cheng. Re-attention transformer for weakly supervised object localization. *arXiv preprint arXiv:2208.01838*, 2022.

[30] Zhenbang Sun, Changhu Wang, Liqing Zhang, and Lei Zhang. Sketch2tag: automatic hand-drawn sketch recognition. In *Proceedings of the 20th ACM International Conference on Multimedia*, pages 1255–1256, 2012.

[31] 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.

[32] Shuanglin Yan, Neng Dong, Liyan Zhang, and Jinhui Tang. Clip-driven fine-grained text-image person re-identification. *IEEE Transactions on Image Processing*, 2023.

[33] Jinyu Yang, Jiali Duan, Son Tran, Yi Xu, Sampath Chanda, Liqun Chen, Belinda Zeng, Trishul Chilimbi, and Junzhou Huang. Vision-language pre-training with triple contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15671–15680, 2022.

[34] Zhengming Zhang, Xiaoming Deng, Jinyao Li, Yukun Lai, Cuixia Ma, Yongjin Liu, and Hongan Wang. Stroke-based semantic segmentation for scene-level free-hand sketches. *The Visual Computer*, 39(12):6309–6321, 2023.

[35] Haowei Zhu, Wenjing Ke, Dong Li, Ji Liu, Lu Tian, and Yi Shan. Dual cross-attention learning for fine-grained visual categorization and object re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4692–4702, 2022.

[36] Ran Zuo, Haoxiang Hu, Xiaoming Deng, Cangjun Gao, Zhengming Zhang, Yukun Lai, Cuixia Ma, Yong-Jin Liu, and Hongan Wang. Scenediff: Generative scene-level image retrieval with text and sketch using diffusion models. 2024.

Appendix

Overview

The organization of the supplementary material is as follows:

- Appendix A: Prompt Design for Teacher and Student Models;
- Appendix B: Details of the Sketch, Text, and Image Encoders;
- Appendix C: Ablation Study on Text and Sketch Inputs;
- Appendix D: Discussion;
- Appendix E: Retrieval Results.

A Prompt Design for Teacher and Student Models

To generate reliable textual annotations for sketch-based scene understanding, we design specific prompts for the teacher model (GPT-4o) and the student model (Qwen3-VL-8B). The overall goal is to ensure that all generated descriptions are strictly grounded in the sketch content and to avoid hallucinated information.

A.1 Prompt Design for the Teacher Model

Scene and instance description based on sketch and reference image: The model is instructed to generate a textual description of the overall scene and the instances it contains by referring to the sketch and its corresponding real image. All descriptions must be grounded in the sketch, and any objects or scenes not present in the sketch are explicitly disallowed. **Attribute inference based on sketch evidence:** Based on the initial description, the model is further instructed to infer missing scene-level or instance-level attributes that are not explicitly mentioned in the text, such as color distribution, pose, background, events, or actions. All inferred attributes must be reasonably deduced from the sketch content, without introducing information beyond the sketch. This prompt design enables the teacher model to produce detailed and faithful textual supervision for training the student model.

A.2 Prompt Design for the Student Model

Scene and instance description based on sketch: The model is instructed to generate a textual description of the scene and instances based solely on the given sketch. Descriptions of objects or scenes not present in the sketch are not allowed. **Attribute inference based on sketch evidence:** Based on the generated description, the model is further instructed to infer missing attributes of the scene and instances (e.g., color, pose, background, events, and actions) using only visual cues available in the sketch.

B Details of the Sketch, Text, and Image Encoders

Sketch Encoder Given a sketch S, we use a pre-trained Vision Transformer (ViT) as the sketch encoder. First, we divide the sketch into $N_s = N_{ps} * N_{ps}$ patches. Next, each patch is mapped to a fixed-length sequence, with a [CLS] token appended at the beginning. Ultimately, the Transformer processes the $N_s + 1$ tokens to extract the sketch feature $S = \{s_g; s_1, \ldots, s_{N_s}\}$, where s_g denotes the global feature of the sketch, and the local feature corresponding to each sketch patch is represented as $\{s_1, \ldots, s_{N_s}\}$.

Text Encoder For a given text T, we tokenize it into N_t word pieces [10] and prepend a special token (e.g., `[SOS]`), following CLIP's text processing pipeline. We then use CLIP's pre-trained text encoder [26] to encode the token sequence and obtain the text representation $T = \{t_g; t_1, \ldots, t_{N_t}\}$, where t_g denotes the global text feature (taken from the final-layer embedding of the special token), and $\{t_1, \ldots, t_{N_t}\}$ are contextualized word-level token features.

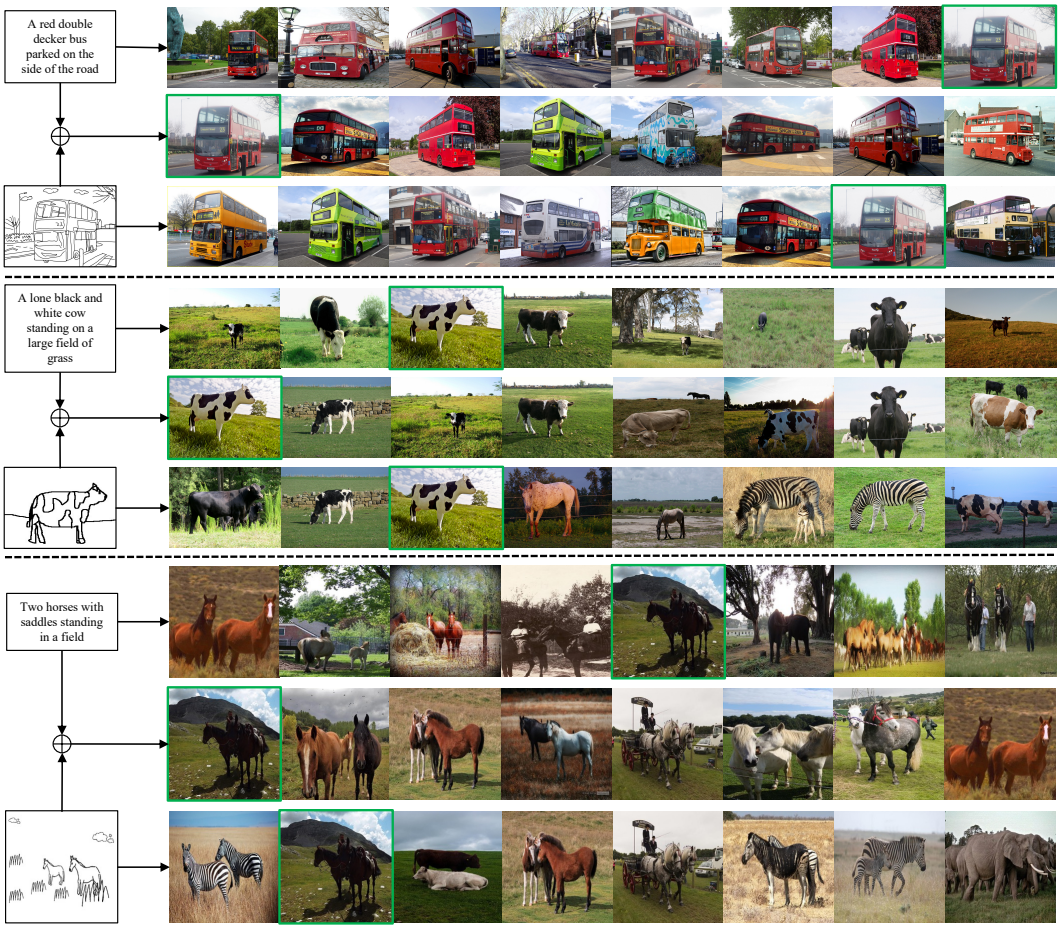

Figure 7: Examples of retrieval results using text, sketch and text-sketch as input

Image Encoder Given an image V, we employ a pre-trained Vision Transformer (ViT) as the image encoder. Similar to sketch, we obtain the image feature representation $V = \{v_g; v_1, \ldots, v_{N_v}\}$, where v_g represents the global features of the entire image, while $\{v_1, \ldots, v_{N_v}\}$ denotes the local features associated with individual image patches.

C Ablation Study on Text and Sketch Inputs

To further validate the model's performance in handling single-modal and multi-modal inputs for image retrieval, we systematically compared three different input: only text, only sketch, and both text and sketch. Figure 7 shows that integrating both textual descriptions and sketches leads to more accurate localization and higher ranking of the target image in the retrieval results. This finding confirms the robustness of our approach in handling multimodal information and underscores the considerable advantage of fusing textual and sketch data in image retrieval tasks.

D Discussion

In the context of image retrieval tasks, particularly those involving sketches and associated textual information, accuracy can be affected by a variety of factors. As illustrated in Figure 8, we show instances where the intended image failed to appear within the top ranks of retrieval. By scrutinizing these instances of suboptimal retrieval outcomes, we can clarify the underlying causes:

1) Information Redundancy / Lack of Specificity: In certain cases, there is an overlap of information between sketches and textual descriptions. Combining information from both modalities may

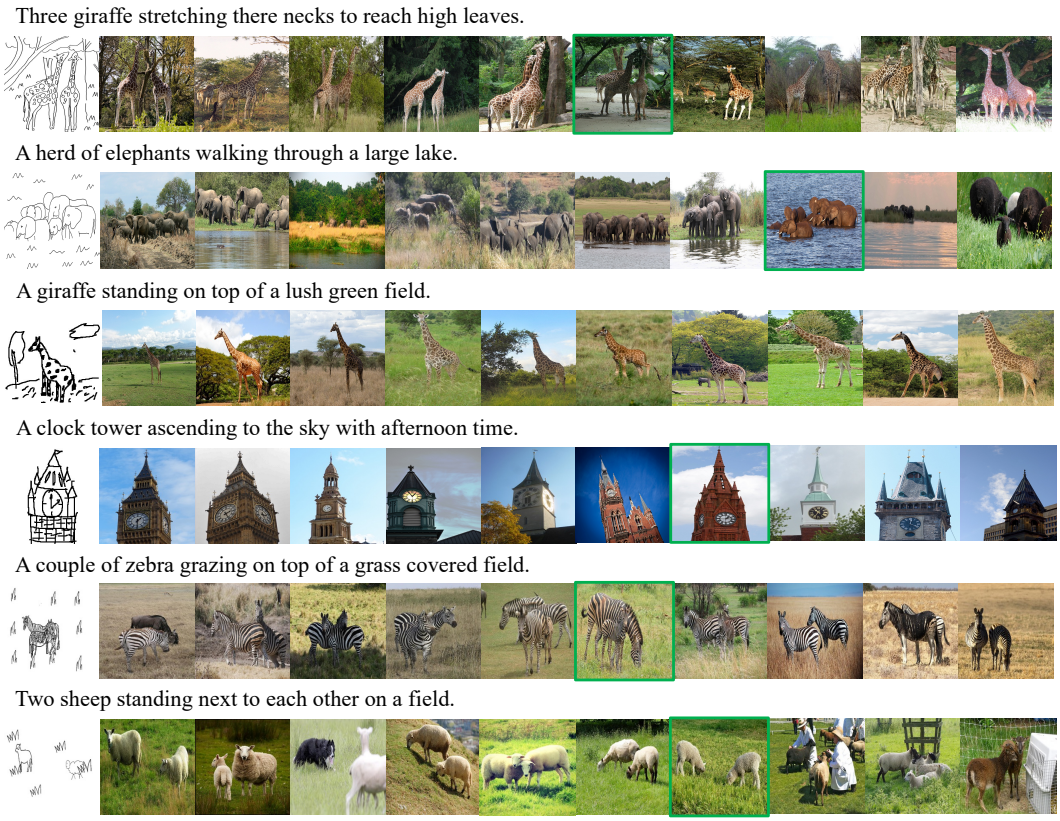

Figure 8: Examples of poor retrieval results.

correspond to multiple correct retrieval results. For instance, in Figure 8, the third row depicts a scenario where both the text and sketch portray a giraffe in a field. However, this depiction lacks enough detail to definitively distinguish the target image. Consequently, the top 10 retrieval outcomes may all align with the given description, but do not match the exact example to be retrieved.

2) Ambiguity: The ambiguity of information arises from the inherent expressiveness of sketches or textual ambiguity. For instance, in the illustration in the second row of Figure 8, while the text refers to an elephant in a lake, the water ripples in the sketch may more closely resemble grass. This visual ambiguity consequently prompts the retrieval system to regard "elephants in grassland" as potential matches.

3) Inaccurate Information: In certain cases, the sketch or text may contain misleading information. For instance, the fifth and sixth rows in Figure 8 depict such scenarios: the text may point to multiple matching images, but specific details in the sketch, such as the orientation of objects, may not match the target image. As a result, the target image may be ranked lower due to incorrect details.

E Retrieval Results

Figures 9, 10, and 11 present additional retrieval results for our model on randomly selected input text-sketch pairs from the SketchyCOCO, SFSD, and FSCOCO datasets, respectively.



Figure 9: The retrieval results of our model on SFSD dataset.



Figure 10: The retrieval results of our model on SketchyCOCO dataset.

An abandoned yellow bus in grassy area next to a hill.



A truck is sitting in the road.



A couple of street signs mounted to the top of a metal pole.



A brown and a black rooster stand on a bed of hay.



A tower of a church above a clock and window.



Two elephants facing each other and one is touching the other with its trunk.



A person that is riding a motorcycle in the dirt.



Figure 11: The retrieval results of our model on FSCOCO dataset.

NeurIPS Paper Checklist

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

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

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

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

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

IMPORTANT, please:

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

1. **Claims**

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

   Answer: [Yes]

   Justification: The main claims in the abstract and introduction are supported by the proposed method, ablation studies, and comparisons on multiple scene-level retrieval datasets.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

Justification: The paper discusses limitations.

Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [N/A]

   Justification: [N/A]

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: The paper provides dataset splits, preprocessing details, model architecture, training settings, evaluation metrics, and implementation details necessary to reproduce the main experimental results.

   Guidelines:

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

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

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

   Answer: [Yes]

   Justification: We provide the code and instructions for data preparation, training, and evaluation in the supplemental material.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: The paper provides dataset splits, preprocessing details, model architecture, training settings, evaluation metrics, and implementation details necessary to reproduce the main experimental results.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: The paper reports standard retrieval metrics following prior work.

   Guidelines:

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: The paper reports the main computational setup, including the use of NVIDIA A40 GPU and the average retrieval latency.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The research uses publicly available datasets and does not involve sensitive personal data, human-subject experiments, or applications that violate the NeurIPS Code of Ethics.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: The paper discusses the potential positive impact of multimodal retrieval for flexible visual search.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

Answer: [N/A]

Justification: [N/A]

Guidelines:

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

12. **Licenses for existing assets**

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

Answer: [Yes]

Justification: The paper properly cites the existing datasets, models, and related works used in the experiments.

Guidelines:

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

13. **New assets**

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

Answer: [Yes]

Justification: The paper proposed model and experimental implementation.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.

- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

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

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments or research with human subjects.

    Guidelines:

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

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

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or human-subject research.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

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

    Answer: [Yes]

    Justification: Using LLMs for distillation learning.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.