Fine-Grained Visual Prompt and Region Self-Distillation for Retrieval-Augmented VQA

Supplementary Material

A. Prompt Templates for Caption Generation

In this section, Fig. 5 and Fig. 6 present the detailed prompt templates used by FVPMR for generating image captions and region captions (based on fine-grained visual prompts), respectively. Note that due to space constraints, the caption generation template shown in Fig. 1 of the main text is a simplified version of Fig. 6, while retaining the core content.

Prompt Template for Global Image Caption Generation

Please describe this image in 10-20 words.

Figure 5. Prompt template for global image caption generation.

Prompt Template for Region Caption Generation

Generate a concise and accurate caption (10–20 words) for the region marked by the red contour.

Input:
- Region Category: {question keyword}
- Question: {question}

Instructions:
1. Describe the region content only (ignore the red contour line).
2. Mention key visual details relevant to the region category: color, shape, state, etc.
3. Add surrounding context (e.g., nearby objects, spatial location, background elements) only if essential to answer the question.

Key principle:
Focus on what helps answer the question best — the region itself or its surroundings. If the region information alone is sufficient to answer the question, describe only the region; otherwise, incorporate the surroundings to fully describe the region.

Output:
Caption: []

Figure 6. Prompt template for region caption generation.

B. Experimental Implementation Details

In this section, we introduce the experimental details from the following aspects: question keyword and definition extraction, fine-grained visual prompt construction, caption generation, and experimental parameter settings.

Question Keyword and Definitions Extraction. We use KeyBERT [12] to extract keywords from the given question and filter to retain keywords with Part Of Speech (POS) as nouns, verbs, and adjectives. Subsequently, we retrieve the corresponding definition descriptions for each keyword in the question from WordNet [32] based on its POS tag, and utilize Sentence-BERT to compute the semantic similarity between the question and each keyword definition. During the knowledge retrieval stage, we retain keyword definitions with similarity scores ≥ 0.35; during the answer generation stage, we retain keyword definitions with similarity scores ≥ 0.5.

Fine-Grained Visual Prompt Construction. We employ Grounded SAM [36] to obtain region segmentation masks. This framework integrates Grounding DINO and SAM [35], enabling automatic localization of relevant regions and generation of corresponding segmentation masks based on input text phrases. Specifically, given a set of question keywords, Grounded SAM first leverages GroundingDINO to generate candidate region bounding boxes based on the keywords. We retain bounding boxes with confidence scores ≥ 0.35, and perform a de-duplication process on overlapping regions with IoU exceeding 0.90 by keeping only the box with the larger area. We further conduct an ablation study on how the bounding-box threshold settings affect retrieval performance in Sec. D.1 of the supplementary material. After obtaining the final set of bounding boxes, we feed them into SAM to generate the corresponding region segmentation masks. Based on these masks, we design two types of visual prompts: red contour prompt and blurred inverse mask visual prompt.

For the red contour-based visual prompt, we annotate the detected regions with red contours, with the contour line width set to 2 pixels. This prompt is used to generate region captions.

For the blurred inverse mask-based visual prompt, we use the $ImageFilter.GaussianBlur()$ function provided by the Pillow library to apply Gaussian blur to the background regions to highlight the target regions. In the Pillow library, the radius parameter is defined as the standard deviation of the Gaussian kernel, which controls the degree of blurring, and we set it to 15 in our experiments. The impact of different radius values on retrieval performance is detailed in Sec. D.2 of the supplementary material.

Caption Generation. In our experiments, consider-

Table 6. Hyper-parameter settings for the knowledge retrieval.

Hyper-parameter	Value
Global image's visual tokens, l_G	50
Region's visual tokens, l_R	25
Query embedding dimension, d_Q	128
Region Number	4
Learning rate	1e-5
Training steps	12000
Batch size	32
Optimizer	Adam

Table 7. Hyper-parameter settings for the answer generation.

Hyper-parameter	Value
Training epoch	5
Learning rate	1e-4
Optimizer	Adam
Hyperparameters for BLIP-2 (T5-XL)	
Batch size	8
Lora_r	8
Lora_alpha	32
Lora_dropout	0.05
Hyperparameters for Qwen2-VL and Qwen2.5-VL	
Batch size (Qwen2.5VL-3B)	6
Batch size (Qwen2VL-7B/Qwen2.5VL-7B)	4
Lora_r	32
Lora_alpha	32
Lora_dropout	0.05

ing the time efficiency and resource consumption for caption generation, we use the relatively smaller Qwen2.5-VL-3B-Instruct [2] as the caption generation model instead of LVLMs with significantly larger parameters, and employ vLLM for deployment and inference. The generation of image and region captions follows the prompt templates provided in Sec. A.

Experimental Setup We conduct experiments on a single NVIDIA A800 GPU. In the knowledge retrieval stage, we adopt ColBERT (v2) [37] as the text encoder and employ the ViT model from CLIP-base-patch32 [33] as the visual encoder, while keeping the visual encoder parameters frozen throughout the experiments. For selecting key regions, we retain the top 4 regions whose region captions have the highest semantic similarity to the given question; if fewer than 4 regions are available, we supplement them by selecting regions corresponding to visual entities in descending order of semantic similarity between the entity categories and the question. The hyper-parameter settings for the knowledge retrieval stage are shown in Tab. 6.

In the answer generation stage, we adopt BLIP-2 (T5-XL)[20] as the primary answer generator and perform lightweight fine-tuning with LoRA [15]. Furthermore, to validate the effectiveness of the proposed method, we also conduct experiments on the OK-VQA dataset with Qwen2VL (7B) [45] and Qwen2.5VL (3B, 7B) [2], all of which are fine-tuned using LoRA. The hyperparameter settings for the answer generation stage are shown in Tab. 7.

Table 8. Comparison of answering performance with baseline methods on FVQA, Infoseek, and E-VQA datasets.

Methods	FVQA	Infoseek	E-VQA
PALI-X [6]	-	21.80	-
PaLM-B	-	-	48.80
BLIP2 T5-XL [20]	61.84	16.04	24.03
Self-Booter [14]	68.13	-	-
PreFLMR [24]	69.42	20.92	55.14
FVPMR	71.58	23.60	56.96

C. Answering Experiments on the Supplementary Datasets

In this section, we report the answering performance of the proposed method on the FVQA, Infoseek, and E-VQA datasets.

For the FVQA [44] dataset, since it is originally designed for answer selection tasks rather than open-ended answer generation, early studies define accuracy as "whether the correct answer node can be selected from all knowledge graph nodes." To adapt to the RAG method, we adopt the answer normalization method used in the OK-VQA dataset to standardize both generated answers and ground-truth answers (including abbreviation expansion, number conversion, article removal, and punctuation cleaning). A KB-VQA question is considered correctly answered when the generated answer exactly matches the ground-truth answer after normalization. This evaluation strategy is applied when assessing baseline methods including BLIP2 T5-XL [20], Self-Booter [14], and PreFLMR [24]. For the Infoseek [7] and E-VQA [31] datasets, we follow PreFLMR's [24] dataset split and knowledge base configuration, and adopt Accuracy and BERT Matching (BEM) metrics respectively to evaluate model performance according to the official evaluation methods. In addition, since Self-Booter [14] does not release model checkpoints on the FVQA dataset, we retrained it; whereas PreFLMR uses text and vision encoder model checkpoints with the same parameter scale as FVPMR to retrieve the Top-5 knowledge. All methods employ BLIP2 T5-XL for answer generation.

As shown in Tab. 8, under the same experimental settings, our method outperforms the baseline methods. Compared to BLIP-2 without external knowledge, FVPMR achieves performance improvements of 9.74%, 7.56%, and

32.93% on the FVQA, Infoseek, and E-VQA datasets respectively, further validating the effectiveness of retrieval-augmented methods in solving KB-VQA questions. Compared to Self-Booter, FVPMR improves accuracy by 3.45% on the FVQA dataset. PreFLMR adopts the same retrieval architecture as our method and uses text and vision encoders with the same parameter scale. Compared to PreFLMR, our method achieves performance improvements of 2.16%, 2.68%, and 1.82% on the three datasets respectively. These results demonstrate that our proposed fine-grained visual prompting and region self-distillation strategy can effectively capture and learn region-level textual and visual features; by more fully leveraging region information, our method enhances knowledge retrieval capability, thereby further improving answering performance.

D. Ablation Experiments

Blurred inverted mask-based FVP is a core component of our method. In the experiments, we first employ Grounded SAM [36] to locate region bounding boxes, then acquire segmentation masks for the corresponding regions, and finally generate fine-grained visual prompts via blurred inverted mask operations. This section analyzes the impact of Grounded SAM's region bounding box threshold and blurred inverted mask parameters on retrieval performance.

D.1. Impact of bounding box threshold on retrieval.

Table 9. The impact of Grounded SAM bounding box threshold and random region selection on retrieval performance.

Experiment Setting	GS (PR@5)
FVPMR	91.95
Baselines	
Top-4 Entity	91.24
Random Entity	90.80
Random Cropping	91.28
Different Bounding Box Thresholds	
Box_Threshold = 0.15	91.20
Box_Threshold = 0.25	91.70
Box_Threshold = 0.35	**91.95**
Box_Threshold = 0.45	91.54
Box_Threshold = 0.55	91.23

We follow the default configuration of Grounded SAM [36] and set the bounding box confidence threshold (Box_Threshold) to 0.35, then input the obtained bounding boxes and images into SAM [35] to acquire region segmentation masks. To evaluate the impact of key regions identified by Grounded SAM on retrieval performance, Tab. 9 presents the PR@5 results under different threshold settings on the OK-VQA dataset, while also listing three baseline methods that do not rely on question keywords to locate key regions: (1) selecting the four visual entities with the highest similarity to the question as key regions (Top-4 Entity); (2) randomly selecting four regions from visual entities (Random Entity); (3) randomly cropping four regions with an area no less than 80×80 from the image (Random Cropping). All the above baseline methods acquire segmentation masks through SAM to construct visual prompts.

As shown in Tab. 9, FVPMR achieves optimal retrieval performance when the threshold is set to 0.35. As the threshold increases, retrieval performance gradually declines, because excessively high thresholds filter out a large number of relevant regions, forcing the model to use less relevant visual entities as substitutes, which introduces noise and reduces retrieval effectiveness. When the threshold reaches 0.55, its retrieval performance is comparable to that of using only the regions corresponding to the top 4 visual entities. Notably, even in this case, the method still significantly outperforms the two baselines—Random Entity and Random Cropping—indicating that locating image regions based on question keywords can effectively incorporate question-relevant visual and textual information, thereby enhancing the model's retrieval capability.

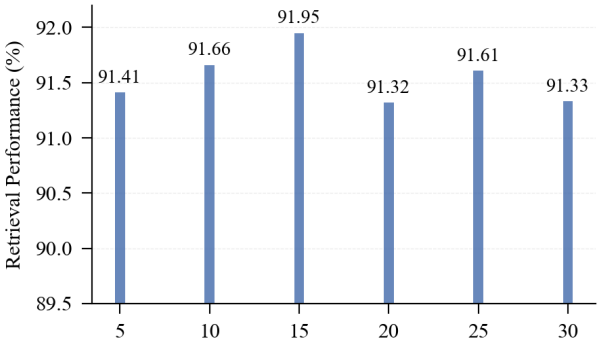

Figure 7. Impact of radius on retrieval performance

D.2. Impact of blurred inverted mask on Retrieval.

In the experiment, after obtaining the segmentation mask, we apply Gaussian blur to the background regions using the $ImageFilter.GaussianBlur(radius)$ function from the Pillow library to highlight the target region. The blur intensity is controlled by the parameter radius, which is defined in the official documentation as the standard deviation of the Gaussian kernel. Fig. 7 illustrates the impact of different radius values on retrieval performance. The experimental results demonstrate that optimal retrieval performance is achieved when radius=15.

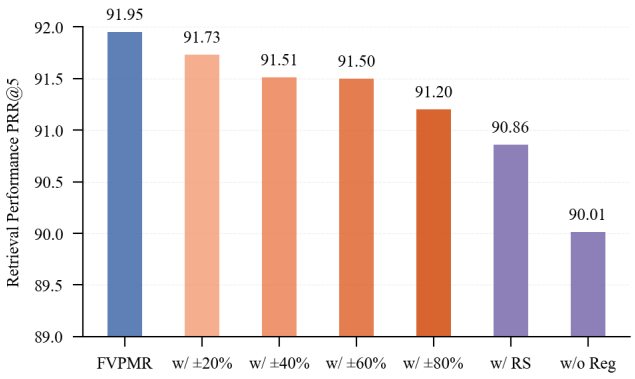

Figure 8. Impact of Region Localization on Retrieval Performance

D.3. Region Localization Robustness Analysis.

In our experiments, we employ Grounded SAM to localize and segment image regions relevant to a given question. Since detection errors are inevitable in real-world scenarios and the KB-VQA dataset lacks fine-grained region-level annotations that would allow direct evaluation of localization errors, we first assume that detection results are perfectly accurate. Building on this, we simulate localization errors on the OK-VQA dataset by applying two types of noise perturbations to the detected bounding boxes—random scaling and random sampling—and further analyze the robustness of the proposed method. Specifically, random scaling resizes the bounding box by ±20%, ±40%, ±60%, and ±80% while keeping its center fixed, whereas random sampling (w/ RS) samples a region of the same size as the original bounding box at a random location within the image, thereby simulating the extreme case of complete localization failure.

As shown in Fig. 8, the retrieval performance of FVPMR declines after noise injection, as the noise alters both the visual and textual features of the original regions. Since random scaling is generated based on the original BBoxes, the semantic center does not deviate completely, resulting in a relatively limited impact on performance. In contrast, randomly sampled regions contain more irrelevant information, leading to a more significant performance drop, though still outperforming the case without region information (w/o Reg). These results demonstrate that FVPMR exhibits strong robustness to moderate localization deviations, but performance degrades further when localization severely deviates. This also confirms the importance of accurate regional localization for retrieval performance.

E. Case Study

In this section, we further analyze the proposed method through several case studies.

In the first case in Fig. 9, the question asks about the purpose of a "Mack truck," so the focus should be on the region related to the "Mack truck" in the image (since the keyword definitions obtained from "Mack truck" have low relevance to the question, keyword definitions are not introduced here). We use Grounded SAM to localize the relevant region and construct FVP with red contours to guide LVLMs in generating caption for that region. The results show that the generated caption accurately characterizes the region marked by the red contour and identifies it as loaded with a "cement mixer" based on the image content, which is highly relevant to the question. In contrast, the caption generated from the cropped region fails to capture this key information about the cement mixer and instead introduces irrelevant content such as "yellow platform." This demonstrates that using FVP to guide LVLMs in generating descriptions is effective—it can combine the question with image content to generate regional text information that better aligns with the current scene, thereby facilitating more accurate retrieval by the model. Furthermore, we construct FVP through blurred inverted masks to guide the model in better learning regional visual features. When retrieval is performed by combining regional visual and textual features (RVT), the model retrieves documents related to "cement," thereby obtaining the correct answer. In contrast, the baseline method without RVT retrieves documents related to "concrete," leading to an incorrect answer.

Similarly, in the second case, the question focuses more on the surrounding environment. Notably, in this case we are unable to localize the relevant region through question keywords, so we select the region corresponding to the visual entity. When generating caption through FVP, the model completely ignores the people dining at the bottom of the image and only describes the surrounding mountains. In contrast, when generating caption using the cropped region, although it considers the mountains appearing in the cropped area, it introduces some weakly related information. Ultimately, when retrieving documents by combining RVT, the model retrieves descriptions about "switzerland" that also mention the "alps," thereby obtaining the correct answer "switzerland."

Since the case in Fig 9 has keywords whose definitions are weakly related to the question, we do not incorporate keyword definitions. We further analyze keyword definitions using the case in Fig 10. As a form of lexical knowledge, keyword definitions can to some extent supplement the information in the question and sometimes even directly mention the answer. In the case in Fig. 10, the question asks about the type of "church," and the keyword definition of "church" explicitly mentions "a place of Christian worship." When we retrieve documents using this information, we obtain documents related to "a place of Christian worship." Without considering the baseline models RVT and KD, the retrieved documents focus on "catholic," which

leads to an incorrect answer.

F. Computational Cost

In this section, we present the time costs of the FVPMR.

Table 10. Time cost of region localization and segmentation, and caption generation.

	time per image (s)
Region Localization	2.57s
Region Segmentation	0.08s
Caption Generation	4.36s

We sequentially fed the images into Grounded SAM and Qwen2.5-VL-3B-Instruct to measure the processing time at each stage. As shown in Tab. 10, we recorded the average time cost per image for region localization, segmentation, and caption generation (including both image-level and region-level captions). The results indicate that caption generation is the most time-consuming step in the entire pipeline. To accelerate the actual processing, we employed batch processing during inference.

Table 11. Training and indexing time for FVPMR. The dataset and corpus used for measuring are OK-VQA and the Google Search corpus.

	train per 1000 steps (h)	indexing time (h)
FLMR	1.20h	0.28h
FVPMR	1.07h	0.13h
w/o FVP2PT	0.85h	0.12h
w/o VTA	1.04h	0.12h

Furthermore, we report the training and indexing time per 1000 steps in Tab. 11, comparing them with FLMR [23]. The results show that both methods maintain roughly consistent training and indexing costs. However, due to differences in experimental environments, this comparison may have certain limitations. Additionally, when the prefix tuning module based on fine-grained visual prompts is removed (*w/o FVP2PT*), the model training time decreases. This is primarily because constructing fine-grained visual prompts through blur inversion masks incurs some additional overhead, but overall this overhead is acceptable. Meanwhile, when we remove the image-text alignment module (*w/o VTA*), the training time remains largely unaffected. This indicates that aligning image and text modalities simultaneously during retrieval does not introduce significant computational overhead—the overhead is even negligible.

Figure 9. Case Study. We present image captions generated based on red contour visual prompts and cropped regions, respectively, and compare them with a baseline model that does not use regional text and visual features (RVT).

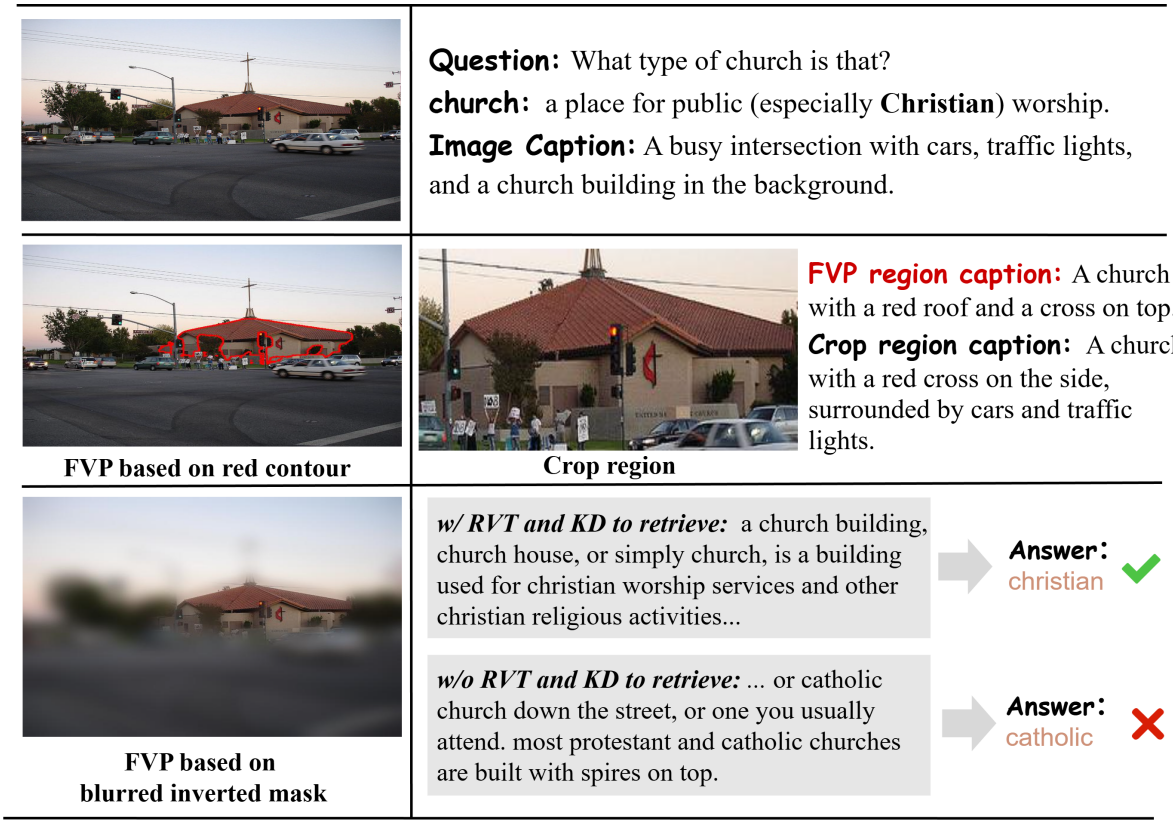

Figure 10. Case analysis on question keyword definitions.