QRouter: Question-Conditioned Visual Evidence Routing for Fine-Grained Visual Question Answering

Anonymous authors
Paper under double-blind review

Abstract

Fine-grained visual question answering (VQA) requires models to derive answers from localized but semantically decisive regions in an image, testing their ability to precisely identify and understand question-relevant visual evidence. However, recent vision-language models still encode images as dense visual token sequences, which in fine-grained VQA often contain substantial question-irrelevant information and make it difficult for the model to retrieve sparse but critical evidence for focused and precise reasoning. To address this limitation, we first provide an analytical perspective showing that the visual support produced by question-conditioned routing can concentrate answer-relevant evidence more effectively than dense question-agnostic visual prefixes. Motivated by this analysis, we present QRouter, a framework that introduces question-conditioned grounding as an intermediate structural prior for fine-grained visual question answering. QRouter uses grounded regions to reorganize dense visual features into a compact, question-relevant structured visual sequence consisting of region tokens, routed context tokens, and a background token, which respectively capture localized evidence, supporting context, and global scene information. This representation more explicitly organizes key visual evidence before multimodal reasoning, highlights the information most relevant to the question, suppresses irrelevant distractions, and provides more effective visual support for fine-grained reasoning. The resulting sequence is then processed by an efficient Mamba-based multimodal backbone. Experiments on seven benchmarks show that QRouter consistently improves over strong open-source baselines on both open-ended and grounding-sensitive question answering tasks. The improvements are particularly pronounced on benchmarks that require compositional reasoning, spatial verification, and hallucination suppression.

1 Introduction

Visual question answering (VQA) requires a model to connect language queries with the visual evidence needed for answering. This challenge becomes especially acute in fine-grained settings, where the correct answer often depends on small but semantically decisive regions, such as a functionally relevant object part, a risky area, a spatial relation, or a subtle text cue. As a result, when the model cannot identify such evidence early, multimodal reasoning has to proceed over dense visual inputs, where the relevant signals can be diluted by large amounts of irrelevant content.

Recent vision-language models Alayrac et al. (2022); Li et al. (2023a); Liu et al. (2023b); Dai et al. (2023); Zhu et al. (2023); Bai et al. (2023) have achieved strong performance on visual question answering (VQA) by combining powerful language backbones with visual encoders. However, these methods still typically represent images as dense and largely question-agnostic visual token sequences, leaving the discovery of question-relevant evidence entirely to the downstream multimodal reasoning process. This design is especially limiting for fine-grained VQA, where correct answers often depend on a small amount of local but semantically decisive visual evidence. When such key evidence is mixed with large back-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

ground regions, irrelevant objects, and redundant context within a uniform visual prefix, the model must not only retrieve sparse but critical information from dense inputs, but also remain robust to distracting noise, which can impair both evidence selection and answer accuracy. This challenge becomes particularly pronounced in tasks involving compositional reasoning, spatial verification, functional understanding, and safety-related judgment. This reveals a central gap in fine-grained VQA: existing models lack an explicit mechanism to effectively organize question-relevant visual evidence before multimodal reasoning begins.

To bridge this gap, we start from an analytical perspective (see Sec. 3.1 for details): when question-conditioned visual routing preserves answer-relevant visual evidence more reliably than irrelevant regions, the resulting visual support becomes more concentrated in useful information than a dense and question-agnostic visual prefix, providing a testable analytical basis for evidence selection in fine-grained visual question answering. Motivated by this view, we propose QRouter, a fine-grained VQA framework that uses question-conditioned grounding as an intermediate structural prior. Unlike existing grounding-aware multimodal systems that mainly treat grounding as an output target or interactive capability Lai et al. (2024); Rasheed et al. (2024); Ren et al. (2024), QRouter uses grounded evidence to restructure the visual input itself for downstream reasoning. Given an image-question pair, QRouter uses grounding to generate question-relevant region priors, and then reorganizes the original dense visual input into a compact, structured, and question-relevant visual representation for downstream multimodal reasoning. By explicitly biasing the visual prefix toward question-relevant regions and their supporting context, QRouter reduces the mismatch between dense visual evidence and the sparse semantic cues actually needed for answering. This structured compression allows the backbone to allocate its limited sequential modeling capacity to more informative evidence.

Experiments on seven benchmarks show that QRouter consistently improves over strong open-source baselines, including recent efficient state-space vision-language models. The gains are especially clear on grounding-sensitive and compositional reasoning tasks, where accurate evidence selection is tightly coupled with answer correctness. These results suggest that organizing visual evidence into structured, question-relevant tokens provides a more suitable representation for fine-grained VQA.

Our key contributions are as follows: First, we show through analysis and experiments that explicitly organizing question-relevant visual evidence before multimodal reasoning provides more suitable visual inputs for evidence selection and fine-grained understanding in VQA. Second, we propose QRouter, which introduces question-conditioned grounding as an intermediate structural prior and reorganizes dense visual inputs into a compact sequence of region tokens, routed context tokens, and a background token for fine-grained multimodal reasoning. Third, we show through extensive experiments that QRouter consistently outperforms strong open-source baselines on both open-ended and grounding-sensitive benchmarks, with especially pronounced improvements on tasks requiring compositional reasoning, spatial verification, and hallucination suppression.

2 Related work

2.1 Vision Language Models for Visual Question Answering

Multimodal large language models (MLLMs) have achieved strong performance across visual question answering, image captioning, and cross-modal reasoning by coupling language models with visual encoders Alayrac et al. (2022); Li et al. (2023a); Liu et al. (2023b); Dai et al. (2023); Zhu et al. (2023); Bai et al. (2023). However, most existing MLLMs rely on Transformer-based backbones, whose quadratic self-attention leads to substantial computational overhead when visual prefixes become long. To improve efficiency, prior work has explored compact MLLMs by reducing model scale or compressing visual context while largely preserving the standard Transformer-based multimodal pipeline Zhu et al. (2024b); Chu et al. (2023; 2024); Zhou et al. (2024); Lin et al. (2026). Representative examples such as LLaVA-Phi Zhu et al. (2024b) and MobileVLM Chu et al. (2023) improve efficiency through smaller language backbones, but often with noticeable performance degradation. This has motivated a shift from model compression toward more efficient backbone architectures.

State space models (SSMs) Wang et al. (2024), including S4 Gu et al. (2021), Mamba Gu & Dao (2023), Mamba-2 Dao & Gu (2024), and vision extensions Zhu et al. (2024a), provide an efficient alternative. Recent multimodal studies, including VL-Mamba Qiao et al. (2024), Cobra Zhao et al. (2025), and MambaVLM Chen et al., have shown the promise of Mamba-based backbones for efficient vision-language modeling.

Nevertheless, existing efficient Mamba-based VLMs still mainly process visual evidence as dense, question-agnostic token streams. Their gains largely come from backbone replacement or multimodal fusion design, while explicit question-conditioned region modeling remains underexplored. We bridge this gap by introducing question-conditioned grounding as a structural prior for reorganizing visual inputs before multimodal reasoning.

2.2 Language-Guided Grounding for Visual Reasoning

Language-guided image segmentation provides an important interface between visual perception and language understanding. Early referring image segmentation (RIS) mainly focuses on object-centric grounding based on identity, attributes, and spatial relations, but remains limited in semantic scope for complex question answering Hu et al. (2016); Yu et al. (2016); Kim et al. (2022); Yang et al. (2022). More recent reasoning-aware segmentation methods, such as ReasonSeg and multimodal dense prediction models including LISA Lai et al. (2024), GLaMM Rasheed et al. (2024), and PixelLM Ren et al. (2024), extend grounding beyond literal object references, yet their main emphasis is still mask prediction or grounded output generation rather than downstream visual reasoning. CONVERSEG Sahoo & Gkioxari (2026) grounds higher-level conversational concepts relevant to VQA. Yet, most VQA models still use dense tokens and learn grounding implicitly Anderson et al. (2018); Das et al. (2017); Li et al. (2023b).

Related grounding-aware vision-language models, including Shikra Chen et al. (2023), Kosmos-2 Peng et al. (2023), GroundHog Zhang et al. (2024), LISA Lai et al. (2024), GLaMM Rasheed et al. (2024), and PixelLM Ren et al. (2024), improve multimodal understanding by introducing phrase grounding, mask prediction, or grounded dialogue outputs. However, grounding in these methods is treated as an output or supervision target, rather than as the primary mechanism for restructuring visual inputs. In contrast, our work uses question-conditioned grounding as an intermediate structural prior for VQA: instead of predicting masks as the end goal, we convert grounded regions into structured visual tokens and route them into an efficient Mamba-based reasoning backbone.

3 QRouter: Question-conditioned token routing with grounding priors

3.1 Overview and Analytical Perspective

Overview. Figure 1 summarizes QRouter. Given an image-question pair, the grounding module predicts question-conditioned region priors, which the Region Tokenizer combines with dual-encoder patch features to form region, routed context, and background tokens. The projected structured sequence and question tokens are then processed by a Mamba-based VQA backbone for answer generation.

Analytical perspective. Let $\mathcal{R}$ denote the set of answer-relevant patches for a question, and let $\mathcal{S}$ denote the routed visual support selected before structured token compression. We define the relevance density of a visual support as

$$\mathrm{RD}(\mathcal{S}) = \frac{|\mathcal{S} \cap \mathcal{R}|}{|\mathcal{S}|}. \tag{1}$$

At a high level, our design is motivated by the observation that if question-conditioned routing preserves answer-relevant patches more reliably than irrelevant ones, then the resulting routed support has higher relevance density than a dense question-agnostic prefix. This provides an analytical rationale for using question-conditioned grounding to reorganize visual inputs before reasoning. The formal statement, proof, and empirical validation are provided in Appendix B. In particular, if the routed support preserves answer-relevant patches with

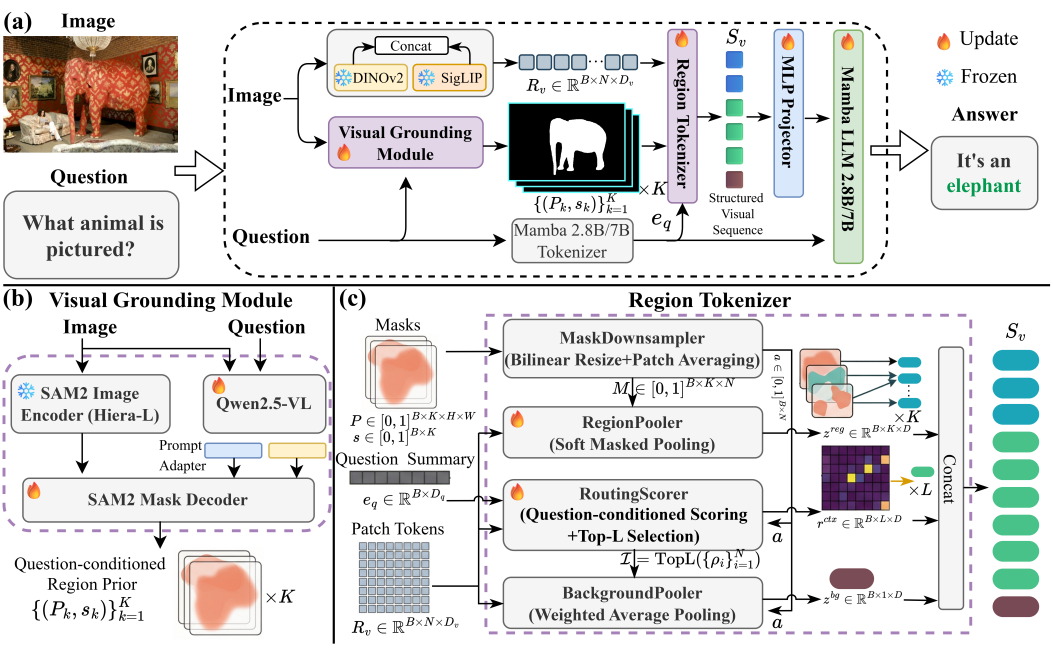


Figure 1: Overview of QRouter. (a) QRouter performs fine-grained Mamba-based VQA on a compact structured visual sequence. (b) The visual grounding module predicts question-conditioned region priors from the input image and question. (c) The Region Tokenization stage converts the region priors and patch features into region tokens, routed context tokens, and a background token, forming the final visual sequence S_v.

recall α and introduces irrelevant patches with false-positive rate β, then $\alpha > \beta$ implies $\mathrm{RD}_{\mathrm{route}} > \mathrm{RD}_{\mathrm{dense}}$.

3.2 Conversational Grounding as Region Prior

Existing VQA models usually consume visual evidence as dense patch tokens and leave the discovery of question-relevant regions entirely implicit. In contrast, conversational grounding Sahoo & Gkioxari (2026) shows that natural-language queries can be localized not only to object instances, but also to relations, affordances, and physics-related concepts. Motivated by this, we use conversational grounding as an intermediate region prior for efficient VQA, rather than as a final segmentation output.

Given an image I and a question q, a grounding module first predicts a set of soft masks with confidence scores:

$$G_\theta(I, q) = \{(P_k, s_k)\}_{k=1}^{K}, \tag{2}$$

where $P_k \in [0,1]^{H\times W}$ is the k-th grounding mask and $s_k \in [0,1]$ is its confidence.

To make grounding compatible with patch-based visual encoding, we project each mask onto the patch grid. Let $\{\Omega_i\}_{i=1}^{N}$ denote the image patches. The patch-level relevance prior is defined as

$$m_{k,i} = \frac{1}{|\Omega_i|} \sum_{(u,v)\in\Omega_i} P_k(u,v), \qquad a_i = 1 - \prod_{k=1}^{K} (1 - s_k m_{k,i}), \tag{3}$$

where $a_i \in [0,1]$ is large when the i-th patch is consistently supported by high-confidence masks. Collecting all patches gives a question-conditioned prior $a = [a_1, \ldots, a_N] \in [0,1]^N$.

Importantly, we do not treat a as the final prediction. Instead, it serves as a structural prior that reorganizes the visual input before multimodal reasoning, allowing the downstream VQA backbone to operate on explicitly question-relevant evidence.

3.3 Region-Aware Visual Tokenization

We next convert the region prior into a compact visual representation suitable for efficient Mamba-based reasoning. Instead of appending dense segmentation features to the backbone, we summarize grounded evidence into a small set of structured tokens. Following efficient Mamba-based VLMs Zhao et al. (2025), we adopt two complementary visual encoders and fuse their patch-aligned features:

$$r_i = [\phi_D(I)_i; \phi_S(I)_i] \in \mathbb{R}^{D_v}, \qquad D_v = D_D + D_S, \qquad i = 1, \ldots, N, \tag{4}$$

where ϕ_D and ϕ_S denote the two visual encoders, D_D and D_S are their feature dimensions, and r_i is the fused feature of patch i.

For each region, we perform confidence-weighted soft pooling over patch features:

$$\tilde{m}_{k,i} = \frac{s_k m_{k,i}}{\sum_{j=1}^{N} s_k m_{k,j} + \epsilon}, \qquad \hat{z}_k^{\text{reg}} = W_r \left(\sum_{i=1}^{N} \tilde{m}_{k,i} r_i \right) + E_{\text{geo}}(b_k), \tag{5}$$

where b_k contains the normalized geometry of the k-th region and $E_{\text{geo}}(\cdot)$ is a lightweight geometry encoder. This produces a compact token aligned with each grounded region.

Routed context tokens. Region tokens alone are often insufficient for VQA, since many questions require supporting context such as neighboring objects or relational anchors. We therefore score each patch conditioned on both the question and the region prior:

$$u_i = w^\top \tanh(W_v r_i + W_a a_i + W_q e_q), \qquad \mathcal{I} = \text{TopL}(\{u_i\}_{i=1}^{N}), \tag{6}$$

where e_q is the pooled question representation and $\mathcal{I}$ contains the indices of the top-L routed context patches. The selected context tokens are modulated by the region prior:

$$\hat{r}_i = r_i + \lambda a_i W_m r_i, \qquad i \in \mathcal{I}, \tag{7}$$

where λ is a scalar coefficient and W_m is learnable.

Background token. To avoid discarding global scene statistics entirely, we compress the remaining patches into a single background token:

$$z^{\text{bg}} = \frac{\sum_{i \notin \mathcal{I}} (1 - a_i) r_i}{\sum_{i \notin \mathcal{I}} (1 - a_i) + \epsilon}. \tag{8}$$

The final visual sequence is then constructed as

$$S_v = [\hat{z}_1^{\text{reg}}, \ldots, \hat{z}_K^{\text{reg}}, \hat{r}_{i_1}, \ldots, \hat{r}_{i_L}, z^{\text{bg}}], \tag{9}$$

where $\{i_1, \ldots, i_L\} = \mathcal{I}$. Thus, the original dense sequence of length N is compressed into $K+L+1$ structured visual tokens, preserving efficiency while explicitly prioritizing question-relevant evidence.

3.4 Question-Conditioned Region Routing in Efficient VQA Backbone

We integrate the structured visual sequence S_v into an efficient Mamba-based VQA backbone. Unlike existing efficient VQA models that prepend dense and question-agnostic visual tokens, our model uses the region-aware sequence in Eq. 9 as the visual prefix.

Let $H_q \in \mathbb{R}^{M \times D}$ denote the question embedding sequence from the text encoder. We project the structured sequence into the same latent space and construct the multimodal input as

$$H_v = \phi(S_v), \qquad H^{(0)} = [H_v + E_{\text{type}},\, H_q], \tag{10}$$

where $\phi(\cdot)$ is a lightweight multimodal projector and E_{type} denotes token-type embeddings for region, context, and background tokens.

We place region tokens before context tokens and before question text, so that the recurrent hidden state of the Mamba backbone is first anchored on highly relevant visual evidence and

Table 1: Comparison with open-source VLMs on public benchmarks. We report four open-ended VQA benchmarks in blue and three grounding-sensitive benchmarks in red. Results marked with $*$ are retrieved from the VLMEvalKit Leaderboard Duan et al. (2024). The best performance is highlighted in bold and the second-best result is underlined.

Model	LLM Backbone	Res.	VQA-v2	GQA	VizWiz	TextVQA	TallyQA	VSR	POPE
Large Scale MLLMs									
OpenFlamingo	MPT-7B	336	52.7	-	27.5	33.6	-	-	42.4*
BLIP-2	Vicuna-13B	224	44.6	41.0	19.6	42.5	-	50.9	85.3
MiniGPT-4	Vicuna-7B	224	32.2	43.5	-	-	-	57.2	48.4*
InstructBLIP	Vicuna-7B	224	76.1	49.2	34.5	50.1	49.8	54.3	84.3
InstructBLIP	Vicuna-13B	224	-	49.5	33.4	50.7	-	52.1	-
Shikra	Vicuna-13B	224	77.4	58.8	-	-	-	-	-
IDEFICS	LLaMA-7B	224	50.9	38.4	35.5	25.9	-	-	81.5*
IDEFICS	LLaMA-65B	224	60.0	45.2	36.0	30.9	-	-	50.3*
Qwen-VL	Qwen-7B	448	78.2	59.3	35.2	63.8	58.7	57.8	62.3*
LLaVA v1.5	Vicuna-7B	336	78.5	62.0	50.0	58.2	62.1	51.5	85.9
Cobra-8B	Mamba-7B	384	79.2	63.9	56.2	59.5	65.8	62.9	87.6
MambaVLM-7B	Vicuna-7B	384	78.6	61.8	55.3	62.6	66.6	68.1	87.9
QRouter-L(ours)	Mamba-7B	384	**83.5**	**70.4**	**61.3**	**64.2**	**68.0**	**68.7**	**92.1**
Small Scale MLLMs									
MoE-LLaVA	StableLM-1.6B	336	76.7	60.3	36.2	50.1	-	-	85.7
MoE-LLaVA	Phi2-2.7B	384	79.9	62.6	43.7	57.0	-	-	85.7
LLaVA-Phi	Phi2-2.7B	336	71.4	36.2	35.9	48.6	-	-	85.0
MobileVLM v2	MobileLLaMA-2.7B	336	-	61.1	-	57.5	-	-	84.7
Cobra-3.5B	Mamba-2.8B	384	77.8	62.3	49.7	58.2	58.2	58.4	88.4
MambaVLM-3B	Mamba-2.8B	384	76.3	59.5	52.5	50.2	59.1	66.7	87.7
QRouter-B(ours)	Mamba-2.8B	384	**82.1**	**68.2**	**53.9**	**61.4**	**60.8**	**68.4**	**91.3**

then refined by supporting context. Given the multimodal input $H^{(0)}$, answer generation follows the standard autoregressive formulation:

$$p(Y \mid I, q) = \prod_{t=1}^{T} p(y_t \mid H^{(0)}, y_{<t}). \tag{11}$$

Unlike prior efficient VQA architectures, QRouter replaces the dense visual prefix with a compact question-conditioned sequence of region, routed context, and background tokens. This focuses visual evidence while preserving Mamba's linear-time inference. Additional formulations and tensor shapes are provided in Appendix B.3.

4 Experiments

4.1 Experimental Setup

Datasets and Metrics. We evaluate on seven benchmarks covering both general VQA and grounding-sensitive settings: VQA-v2 Goyal et al. (2017), GQA Hudson & Manning (2019), TextVQA Singh et al. (2019), VizWiz Gurari et al. (2018), VSR Liu et al. (2023a), POPE Li et al. (2023b), and TallyQA Acharya et al. (2019). These benchmarks collectively cover general visual reasoning, compositional understanding, text reading, real-world assistive QA, spatial reasoning, hallucination robustness, and counting. Following prior work, we report accuracy on all seven benchmarks using their standard evaluation protocols.

Baselines. We compare with representative open-source vision-language models, including general MLLMs such as LLaVA v1.5 Liu et al. (2024), InstructBLIP Dai et al. (2023), and Qwen-VL Bai et al. (2023), compact models such as LLaVA-Phi Zhu et al. (2024b), MobileVLM v2 Chu et al. (2024), and recent state-space baselines including Cobra Zhao et al. (2025), and MambaVLM Chen et al.. For fair comparison, we primarily focus on models with similar parameter scale or efficiency-oriented design.

Implementation Details. For visual representation, we adopt DINOv2 Oquab et al. (2023) and SigLIP Zhai et al. (2023) as dual visual encoders, whose patch features are fused and

reorganized by the proposed RegionTokenizer into a compact structured visual sequence, including region tokens, routed context tokens, and a background token, before multimodal projection. For grounding, we use SAM2 Hiera-L Ravi et al. (2024) with Qwen2.5-VL Bai et al. (2025) to produce question-conditioned masks, which are converted into patch-level priors for region routing rather than treated as final outputs. For language modeling, we use Mamba-2.8B-Zephyr as the default backbone and additionally report a larger variant based on Mamba-7B. Full implementation details are provided in Appendix C.

4.2 Comparison on Open-ended Benchmarks

As shown in Table 1, our method achieves the strongest overall performance on the four open-ended benchmarks among both large-scale and compact open-source VLMs.

In the large-scale setting, QRouter-L consistently outperforms all compared baselines, reaching 83.5 on VQA-v2, 70.4 on GQA, 61.3 on VizWiz, and 64.2 on TextVQA. Compared with Cobra-8B, the gains are especially notable, with improvements of +4.3, +6.5, +5.1, and +4.7, respectively. QRouter-L also surpasses MambaVLM-7B on all four benchmarks, showing that the advantage does not come from the Mamba backbone alone, but from better organizing question-relevant visual evidence before multimodal reasoning.

The same trend holds in the small-scale setting. QRouter-B consistently outperforms compact baselines including LLaVA-Phi, MobileVLM v2, Cobra-3.5B, and MambaVLM-3B, reaching 82.1 on VQA-v2 and 68.2 on GQA. In particular, the gains over Cobra-3.5B are consistent across all four benchmarks, with especially clear improvements on VQA-v2 and GQA, indicating that question-conditioned region routing remains effective even under limited model capacity.

Across both settings, the largest relative gains are observed on GQA, VizWiz, and TextVQA. These benchmarks are particularly sensitive to compositional reasoning, cluttered visual evidence, or small but critical local regions, which matches the motivation of our framework: grounding-guided region routing helps the Mamba backbone focus on semantically relevant evidence instead of processing the image as a uniform dense token stream.

4.3 Fine-Grained Analysis on Grounding-Sensitive Questions

We next evaluate on three grounding-sensitive benchmarks, namely TallyQA, VSR, and POPE, which place stronger emphasis on localized counting, spatial verification, and hallucination suppression than standard open-ended VQA. These benchmarks therefore provide a more direct test of our central claim: our approach is particularly effective on questions whose answers depend on a small amount of question-relevant visual evidence.

As shown in Table 1, QRouter achieves the strongest overall results on all three benchmarks in both model scales. In the large-scale setting, the gains are especially clear on VSR, where QRouter-L outperforms Cobra-8B by +5.8, showing that explicit region organization is particularly beneficial for spatial reasoning. A similar trend holds in the small-scale setting: QRouter-B consistently improves over Cobra-3.5B, with especially large gains on VSR (+10.0) and clear improvements on POPE (+2.9). These results suggest that grounding-aware visual reorganization provides an effective inductive bias for evidence selection in compact as well as large Mamba-based VQA models.

Table 2: Fine-grained performance comparison on GQA structural types. Results are reported on the test-dev split.

Model	Backbone	Query	Compare	Choose	Logical	Verify
CNN	-	1.55	36.34	0.85	47.18	47.02
LSTM	-	22.69	57.79	57.15	61.73	65.78
CNN+LSTM	-	31.80	56.62	61.40	62.05	67.00
Gemma3	Gemma3-4B	25.97	17.38	62.93	19.38	45.63
SmolVLM2	SmolLM2-1.7B	37.28	48.61	66.74	61.50	65.82
Cobra	Mamba-2.8B	<u>48.82</u>	<u>62.99</u>	<u>82.11</u>	<u>76.71</u>	<u>81.75</u>
QRouter-B	Mamba-2.8B	**53.39**	**72.33**	**90.43**	**83.92**	**87.88**
Improv. (Δ)		+4.57	+9.34	+8.32	+7.21	+6.13

To further analyze the gains on GQA, we report fine-grained results on five official structural categories in Table 2. QRouter improves over Cobra on all categories, with the largest gains

Table 3: Ablation Study on Core Components and Training Strategies. Additional ablation results are provided in Appendix D.2.

Model Variant	Region Tokens	Routed Context	Background Tokens	GQA (Acc)	VSR (Acc)	POPE (Acc)
Baseline	✗	✗	✗	61.7	58.7	88.6
+ Region Tokens	✓	✗	✗	64.3	63.8	89.0
+ Routed Context	✓	✓	✗	67.4	67.3	90.1
QRouter-B (One Stage)	✓	✓	✓	67.5	68.0	90.4
QRouter-B (Two Stage)	✓	✓	✓	**68.2**	**68.4**	**91.3**

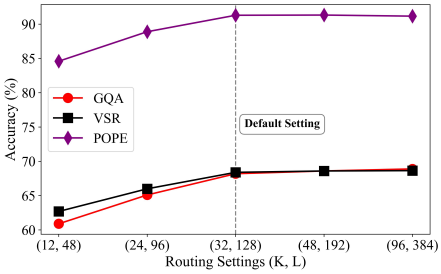


Figure 2: Budget sensitivity with different routing settings (K,L).

on Compare (+9.34), Choose (+8.32), and Logical (+7.21). This pattern is consistent with our design: explicitly promoting question-relevant regions and routed context is especially helpful for reasoning over alternatives, relations, and compositional dependencies.

4.4 Ablation Study

Component Analysis. Table 3 shows each component contributes positively. Adding Region Tokens brings clear gains over the baseline on all three benchmarks, confirming the value of promoting question-relevant grounded regions into dedicated visual tokens. Introducing Routed Context further improves performance, indicating that focused region evidence alone is insufficient and that supporting contextual cues remain important for robust reasoning. Finally, adding the Background Token yields consistent additional gains, suggesting that a compact global scene summary still provides useful complementary information. Overall, the results support our structured visual design, where grounded regions, routed context, and global background play distinct but complementary roles.

Training Strategies. Table 3 shows that the two-stage training strategy is better than one-stage training under the same architecture. QRouter-B (Two Stage) improves over QRouter-B (One Stage) on GQA, VSR, and POPE, indicating that staged optimization leads to better alignment between grounding and reasoning. A likely reason is that the model first learns stable region-aware reasoning in the VQA backbone, and only then introduces lightweight adaptation in the grounding branch, reducing optimization interference in early training.

Budget Sensitivity. Figure 2 studies the effect of different routing budgets by varying the numbers of region tokens and routed context tokens. As the budget increases from $(12, 48)$ to $(32, 128)$, performance improves consistently on all three benchmarks, showing that overly small visual budgets are insufficient to retain the region-level summaries and supporting context needed for reliable reasoning. The gains are most evident on GQA and VSR, while further enlarging the budget beyond $(32, 128)$ yields only marginal improvements and the POPE curve is nearly saturated. This suggests diminishing returns once enough question-relevant foreground and contextual information has been preserved. Therefore, we adopt $(K, L) = (32, 128)$ as the default setting, as it provides a good balance between reasoning performance and visual token efficiency.

4.5 Efficiency and Qualitative Analysis

Table 4 reports post-grounding latency and throughput, covering region tokenization, routing, visual projection, and answer generation. QRouter remains substantially more efficient than Transformer-based compact baselines such as MoE-LLaVA and LLaVA-Phi. At the same time, QRouter-B is nearly on par with Cobra-3.5B in post-grounding latency (0.0862 vs. 0.0857 s/sample) and throughput (143.5035 vs. 144.3407 tokens/s), despite the additional routing and region tokenization computations. This result is noteworthy because QRouter reduces the visual prefix from 729 tokens to 161, although this reduction does not translate into a substantial post-grounding speedup. This suggests that, in our setting, the additional routing computation offsets part of the efficiency benefit provided by the shorter visual prefix. Overall, QRouter preserves the fast inference characteristics of efficient Mamba-based VQA systems while reorganizing visual evidence into a more question-relevant structure.

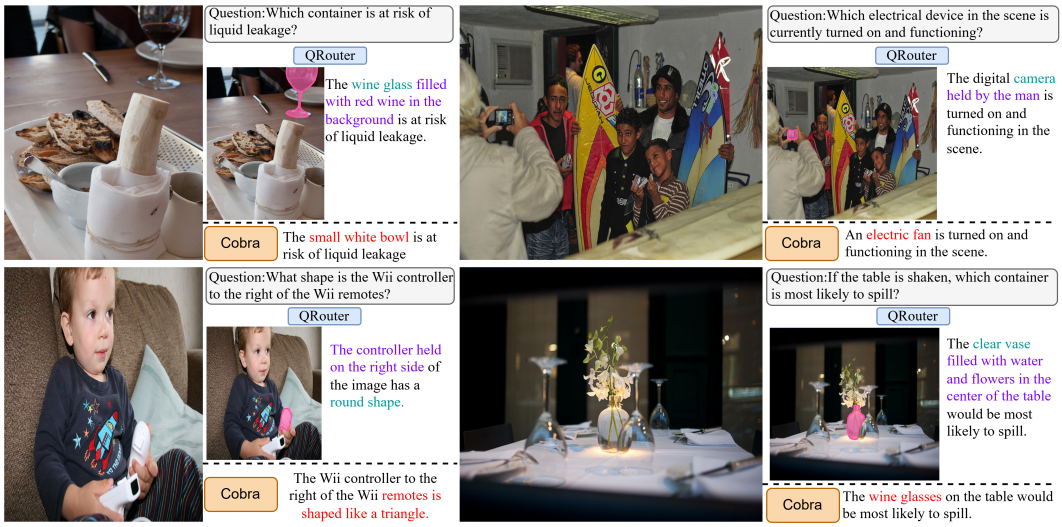


Figure 3: Visualization of qualitative results. We elaborately design question-answering cases covering physics, safety, and functions to challenge the models' deep compositional reasoning. Compared with Cobra, our model generates more accurate and intent-grounded descriptions. Text in green indicates correct answers, red denotes inaccurate predictions, and purple highlights a more detailed scene understanding.

This trade-off is favorable for our goal: QRouter achieves stronger VQA performance while keeping its post-grounding inference cost close to Cobra and well below Transformer-based alternatives. A breakdown of grounding latency, post-grounding VQA latency, and complete online latency is provided in Appendix D.6.

Table 4: Post-grounding latency and throughput comparison.

Model	Backbone	Tokens	Latency (s/sample)	Throughput (tokens/s)
MoE-LLaVA	Phi2-2.7B	576	0.3986	31.0336
LLaVA-Phi	Phi2-2.7B	576	0.2661	46.4863
Cobra-3.5B	Mamba-2.8B	729	**0.0857**	**144.3407**
QRouter-B	Mamba-2.8B	161	<u>0.0862</u>	<u>143.5035</u>
QRouter-L	Mamba-7B	161	0.1418	87.2355

Visualization of the challenging examples. As shown in Figure 3, QRouter consistently outperforms the Cobra baseline across challenging scenarios. In the first and second examples involving functional states and liquid leakage, Cobra provides inaccurate answers (i.e., hallucinating an electric fan and misidentifying an empty bowl), whereas QRouter accurately grounds the functioning camera and the wine glass in the background. In the third example, QRouter correctly grounds the controller on the right and identifies its round shape, whereas Cobra misclassifies it as triangular. Finally, in the physics-based scenario, QRouter correctly recognizes the physical state of the inverted empty glasses and accurately identifies the water-filled vase. More experimental results and visualizations can be found in Appendix D.

5 Conclusion

In this work, we proposed QRouter, a visual question answering framework for fine-grained reasoning. Our method introduces question-conditioned grounding as an intermediate structural prior and reorganizes dense visual inputs into a compact structured sequence of region tokens, routed context tokens, and a background token. This design enables the model to focus on question-relevant visual evidence before multimodal reasoning. Our analysis establishes a testable condition under which routing increases relevance density. Pixel- and patch-level evaluations empirically support the predicted evidence concentration and show a positive association between grounding quality and answer accuracy. Across diverse VQA benchmarks, QRouter consistently outperforms competitive open-source baselines, with especially clear gains on grounding-sensitive and compositional reasoning tasks, while preserving the favorable efficiency characteristics of compact Mamba-based backbones. These results suggest that explicitly organizing question-relevant visual evidence is an effective direction for efficient fine-grained visual question answering.

AI use statement

Generative AI tools were used primarily to improve the clarity, grammar, and readability of the manuscript. They were also used to provide feedback on methodological exposition, experimental design, and consistency between the manuscript and the accompanying implementation, as well as to assist with limited code editing. Generative AI was not used to generate datasets, conduct experiments, or fabricate or alter experimental results. The authors made all final research decisions and take full responsibility for the content of this work.

Ethics statement

This work uses publicly available datasets and pretrained models and does not involve human-subject experiments, newly collected personal data, or the release of a new dataset. We follow the licenses and intended research use of all employed resources. Like other multimodal large language models, QRouter may inherit biases from its training data or produce inaccurate and hallucinated outputs; therefore, deployment in high-stakes settings would require additional validation and appropriate safeguards. We identify no additional ethical concerns specific to the proposed method.

Reproducibility statement

We have taken several steps to facilitate reproducibility. Sections 3 and 4 describe the model architecture, training strategy, datasets, baselines, and evaluation settings. Appendix B provides the assumptions, complete theoretical proof, empirical validation, and detailed model formulations. Appendix C documents the training data, stage-wise procedure, hyperparameters, hardware, prompt templates, decoding settings, and benchmark-specific evaluation protocols. Appendix D provides additional evaluation details, ablations, controlled experiments, grounding metrics, and the online inference latency protocol. We also provide anonymous source code as supplementary material, including the model implementation, configuration files, and training and evaluation scripts needed to reproduce the reported experiments. The public datasets and benchmark splits used in this work are cited in the paper, and their official evaluation protocols are followed unless otherwise stated.

References

Manoj Acharya, Kushal Kafle, and Christopher Kanan. Tallyqa: Answering complex counting questions. In Proceedings of the AAAI conference on artificial intelligence, volume 33, pp. 8076–8084, 2019.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. Advances in neural information processing systems, 35:23716–23736, 2022.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 6077–6086, 2018.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A frontier large vision-language model with versatile abilities. arXiv preprint arXiv:2308.12966, 1(2):3, 2023.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025. URL https://arxiv.org/abs/2502.13923.

Honghao Chen, Yibing Song, Shoufa Chen, Chongjian GE, and Kaiqi Huang. Unleashing the power of selective state space models in vision-language models.

Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm's referential dialogue magic. arXiv preprint arXiv:2306.15195, 2023.

Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 1290–1299, 2022.

Xiangxiang Chu, Limeng Qiao, Xinyang Lin, Shuang Xu, Yang Yang, Yiming Hu, Fei Wei, Xinyu Zhang, Bo Zhang, Xiaolin Wei, et al. Mobilevlm: A fast, strong and open vision language assistant for mobile devices. arXiv preprint arXiv:2312.16886, 2023.

Xiangxiang Chu, Limeng Qiao, Xinyu Zhang, Shuang Xu, Fei Wei, Yang Yang, Xiaofei Sun, Yiming Hu, Xinyang Lin, Bo Zhang, et al. Mobilevlm v2: Faster and stronger baseline for vision language model. arXiv preprint arXiv:2402.03766, 2024.

Wenliang Dai, Junnan Li, Dongxu Li, Anthony Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale N Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning. Advances in neural information processing systems, 36:49250–49267, 2023.

Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. arXiv preprint arXiv:2405.21060, 2024.

Abhishek Das, Harsh Agrawal, Larry Zitnick, Devi Parikh, and Dhruv Batra. Human attention in visual question answering: Do humans and deep networks look at the same regions? Computer Vision and Image Understanding, 163:90–100, 2017.

Haodong Duan, Junming Yang, Yuxuan Qiao, Xinyu Fang, Lin Chen, Yuan Liu, Xiaoyi Dong, Yuhang Zang, Pan Zhang, Jiaqi Wang, et al. Vlmevalkit: An open-source toolkit for evaluating large multi-modality models. In Proceedings of the 32nd ACM international conference on multimedia, pp. 11198–11201, 2024.

Dengxian Gong, Yuanzheng Wu, Haobo Yuan, Zhengdong Hu, Tao Zhang, Yikang Zhou, Shihao Chen, Quanzhu Niu, Kai Wang, Jason Li, Haochen Wang, Lu Qi, Shunping Ji, and Ming-Hsuan Yang. Pixeleyes: Decoupling perception and reasoning for pinpoint visual evidence seeking, 2026. URL https://arxiv.org/abs/2607.00115.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 6904–6913, 2017.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. arXiv preprint arXiv:2312.00752, 2023.

Albert Gu, Karan Goel, and Christopher Ré. Efficiently modeling long sequences with structured state spaces. arXiv preprint arXiv:2111.00396, 2021.

Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3608–3617, 2018.

Ronghang Hu, Marcus Rohrbach, and Trevor Darrell. Segmentation from natural language expressions. In European conference on computer vision, pp. 108–124. Springer, 2016.

Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 6700–6709, 2019.

Namyup Kim, Dongwon Kim, Cuiling Lan, Wenjun Zeng, and Suha Kwak. Restr: Convolution-free referring image segmentation using transformers. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 18145–18154, 2022.

Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9579–9589, 2024.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In International conference on machine learning, pp. 19730–19742. PMLR, 2023a.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In Proceedings of the 2023 conference on empirical methods in natural language processing, pp. 292–305, 2023b.

Bin Lin, Zhenyu Tang, Yang Ye, Jinfa Huang, Junwu Zhang, Yatian Pang, Peng Jin, Munan Ning, Jiebo Luo, and Li Yuan. Moe-llava: Mixture of experts for large vision-language models. IEEE Transactions on Multimedia, 2026.

Fangyu Liu, Guy Emerson, and Nigel Collier. Visual spatial reasoning. Transactions of the Association for Computational Linguistics, 11:635–651, 2023a.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. Advances in neural information processing systems, 36:34892–34916, 2023b.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 26296–26306, 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193, 2023.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. arXiv preprint arXiv:2306.14824, 2023.

Yanyuan Qiao, Zheng Yu, Longteng Guo, Sihan Chen, Zijia Zhao, Mingzhen Sun, Qi Wu, and Jing Liu. Vl-mamba: Exploring state space models for multimodal learning. arXiv preprint arXiv:2403.13600, 2024.

Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. Advances in neural information processing systems, 34:13937–13949, 2021.

Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S Khan. Glamm: Pixel grounding large multimodal model. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 13009–13018, 2024.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. arXiv preprint arXiv:2408.00714, 2024.

Zhongwei Ren, Zhicheng Huang, Yunchao Wei, Yao Zhao, Dongmei Fu, Jiashi Feng, and Xiaojie Jin. Pixellm: Pixel reasoning with large multimodal model. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 26374–26383, 2024.

Michael Ryoo, AJ Piergiovanni, Anurag Arnab, Mostafa Dehghani, and Anelia Angelova. Tokenlearner: Adaptive space-time tokenization for videos. Advances in neural information processing systems, 34:12786–12797, 2021.

Aadarsh Sahoo and Georgia Gkioxari. Conversational image segmentation: Grounding abstract concepts with scalable supervision. arXiv preprint arXiv:2602.13195, 2026.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 8317–8326, 2019.

Xiao Wang, Shiao Wang, Yuhe Ding, et al. State space model for new-generation network alternative to transformers: A survey. arXiv preprint arXiv:2404.09516, 2024.

Penghao Wu and Saining Xie. V*: Guided visual search as a core mechanism in multimodal llms. In 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 13084–13094. IEEE, 2024.

Zhao Yang, Jiaqi Wang, Yansong Tang, Kai Chen, Hengshuang Zhao, and Philip HS Torr. Lavt: Language-aware vision transformer for referring image segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 18155–18165, 2022.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In European conference on computer vision, pp. 69–85. Springer, 2016.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 11975–11986, 2023.

Yichi Zhang, Ziqiao Ma, Xiaofeng Gao, Suhaila Shakiah, Qiaozi Gao, and Joyce Chai. Groundhog: Grounding large language models to holistic segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 14227–14238, 2024.

Han Zhao, Min Zhang, Wei Zhao, Pengxiang Ding, Siteng Huang, and Donglin Wang. Cobra: Extending mamba to multi-modal large language model for efficient inference. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 39, pp. 10421–10429, 2025.

Baichuan Zhou, Ying Hu, Xi Weng, Junlong Jia, Jie Luo, Xien Liu, Ji Wu, and Lei Huang. Tinyllava: A framework of small-scale large multimodal models. arXiv preprint arXiv:2402.14289, 2024.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. arXiv preprint arXiv:2304.10592, 2023.

Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. Vision mamba: Efficient visual representation learning with bidirectional state space model. In Forty-first International Conference on Machine Learning, 2024a.

Yichen Zhu, Minjie Zhu, Ning Liu, Zhiyuan Xu, and Yaxin Peng. Llava-phi: Efficient multi-modal assistant with small language model. In Proceedings of the 1st International Workshop on Efficient Multimedia Computing under Limited, pp. 18–22, 2024b.

A Appendix Overview

Appendix Overview. To facilitate navigation and improve readability, we briefly summarize the organization of the appendix.

Appendix B.1 presents the evidence-concentration analysis, its empirical validation, and the detailed formulations of QRouter.

Appendix C documents the two-stage training setup, implementation details, hyperparameters, and prompt and evaluation protocols.

Appendix D.1 extends the GQA analysis to semantic types and compositional reasoning depth.

Appendix D.2 reports additional ablations on the visual and grounding encoders.

Appendix D.3 provides additional qualitative comparisons and an exact-match error analysis.

Appendix D.4 evaluates sensitivity to mismatched grounding through a question-shuffled control and qualitative interventions.

Appendix D.5 examines the empirical association between grounding quality and answer accuracy on VizWiz-VQA-Grounding.

Appendix D.6 decomposes online latency into grounding, post-grounding VQA, and complete online inference.

Finally, Appendix E compares QRouter with recent agentic visual-search methods and clarifies the distinction between their multi-turn search paradigm and our fixed-budget, question-conditioned routing framework.

B Theoretical Analysis, Empirical Validation, and Additional Formulations

B.1 A Testable Analytical Perspective on Evidence Concentration

We provide a simple analytical perspective on why question-conditioned routing can help fine-grained VQA. Rather than proving end-task accuracy directly, the goal here is to characterize when QRouter exposes a visual support that is more concentrated in answer-relevant evidence than a dense question-agnostic prefix.

Setup. Consider an image represented by N visual patches. For a given question, let $\mathcal{R} \subseteq \{1, \dots, N\}$ denote the set of answer-relevant patches, where $|\mathcal{R}| = m$ and $0 < m < N$. We interpret the dense baseline as exposing all N patches to the multimodal backbone. In contrast, QRouter induces a routed support set $\mathcal{S}$ consisting of the patches emphasized by its question-conditioned region prior and routed context selection before structured token compression.

Lemma 1. The relevance density of a dense question-agnostic visual prefix is

$$\mathrm{RD}_{\mathrm{dense}} = \frac{m}{N}. \tag{12}$$

Proof. Since the dense baseline exposes all N patches and exactly m of them are answer-relevant, the fraction of relevant evidence in its visual support is m/N. □

Lemma 2. Define the relevant-patch recall and irrelevant-patch false-positive rate of the routed support by

$$\alpha = \frac{|\mathcal{S} \cap \mathcal{R}|}{m}, \qquad \beta = \frac{|\mathcal{S} \setminus \mathcal{R}|}{N - m}. \tag{13}$$

Then the relevance density of the routed support is

$$\mathrm{RD}_{\mathrm{route}} = \frac{|\mathcal{S} \cap \mathcal{R}|}{|\mathcal{S}|} = \frac{\alpha m}{\alpha m + \beta(N - m)}. \tag{14}$$

Proof. By definition, $|\mathcal{S} \cap \mathcal{R}| = \alpha m$ and $|\mathcal{S} \setminus \mathcal{R}| = \beta(N - m)$. Therefore,

$$|\mathcal{S}| = |\mathcal{S} \cap \mathcal{R}| + |\mathcal{S} \setminus \mathcal{R}| = \alpha m + \beta(N - m),$$

which gives the stated expression. □

Proposition 3. Assume $0 < m < N$. If

$$\alpha > \beta, \tag{15}$$

then

$$\mathrm{RD}_{\mathrm{route}} > \mathrm{RD}_{\mathrm{dense}}. \tag{16}$$

Proof. Using Lemma 1 and Lemma 2,

$$\mathrm{RD}_{\mathrm{route}} > \mathrm{RD}_{\mathrm{dense}} \iff \frac{\alpha m}{\alpha m + \beta(N - m)} > \frac{m}{N}.$$

Since $m > 0$ and $N > m$, cross-multiplication yields

$$\alpha N > \alpha m + \beta(N - m) \iff \alpha(N - m) > \beta(N - m) \iff \alpha > \beta.$$

Hence the claim follows. □

Discussion. Proposition 3 does not claim that QRouter is universally optimal or that higher relevance density automatically guarantees higher task accuracy. Instead, it yields a concrete and testable condition: if question-conditioned routing preserves answer-relevant patches more reliably than irrelevant ones, then the resulting routed support is more concentrated in useful evidence than a dense question-agnostic prefix. This perspective captures evidence concentration rather than completeness of evidence coverage; in practice, effective routing should also maintain adequate recall over answer-relevant patches. When routing is used to form a compact visual support under a constrained token budget, this provides a simple mechanism for why QRouter can reduce the mismatch between dense visual inputs and the sparse semantic cues needed for fine-grained VQA. Moreover, the condition $\alpha > \beta$ can be empirically examined on held-out grounding supervision by measuring patch-level recall and false-positive rate. More broadly, this analytical perspective is also consistent with prior work suggesting that retaining a compact set of informative visual tokens can improve the efficiency–accuracy trade-off in vision models Rao et al. (2021); Ryoo et al. (2021).

B.2 Empirical Validation of Evidence Concentration

The proposition above provides a testable condition for evidence concentration rather than a guarantee of end-task correctness. We evaluate this condition on all 2,373 examples from the VizWiz-VQA-Grounding test set, which provides human-annotated answer-grounding masks. Following the notation in Appendix B.1, R denotes the set of annotated relevant patches, $S = \{i : a_i \geq 0.5\}$ denotes the patch support induced by the aggregated grounding prior, and N denotes the total number of image patches. The K predicted soft masks are aggregated using the confidence-weighted noisy-OR rule in Appendix B.3. For pixel-level evaluation, both the aggregated prediction and the normalized ground-truth mask are binarized at 0.5. For patch-level evaluation, the ground-truth mask is area-downsampled to the visual patch grid, and patches with at least 0.5 coverage are treated as relevant.

At the pixel level, we compare the predicted grounding mask with the human-annotated mask using mean intersection over union (mIoU), precision, and recall. At the patch level, we measure the relevant-patch recall α, the irrelevant-patch false-positive rate β, and the relevance densities $\mathrm{RD}_{\mathrm{route}}$ and $\mathrm{RD}_{\mathrm{dense}}$. All quantities are computed before answer scoring and therefore evaluate evidence selection independently of final VQA accuracy.

As shown in Table 5, QRouter achieves 81.6% mIoU, 89.2% precision, and 87.4% recall at the pixel level. At the patch level, $\alpha = 92.4\%$ substantially exceeds $\beta = 6.2\%$, satisfying the selectivity condition $\alpha > \beta$. Consistent with the proposition, routing increases the relevance density from $\mathrm{RD}_{\mathrm{dense}} = 20.3\%$ to $\mathrm{RD}_{\mathrm{route}} = 79.1\%$, corresponding to an approximately $3.9\times$ increase in the proportion of relevant evidence.

Table 5: Empirical evaluation of evidence selection on all 2,373 examples in the VizWiz-VQA-Grounding test set. All values are percentages. Pixel-level metrics evaluate grounding localization, whereas patch-level metrics evaluate the visual support exposed to the reasoning backbone.

Pixel-level grounding			Patch-level evidence selection			
mIoU ↑	Precision ↑	Recall ↑	α ↑	β ↓	$\mathrm{RD}_{\mathrm{route}}$	$\mathrm{RD}_{\mathrm{dense}}$
81.6	89.2	87.4	92.4	6.2	79.1	20.3

These results provide empirical support for the proposition's selectivity condition and predicted relevance-density ordering on this benchmark. However, they should not be interpreted as a strict causal validation. In particular, the analysis does not establish that increasing relevance density necessarily causes higher answer accuracy, nor does it imply that the observed relationship holds universally across datasets or routing configurations.

B.3 Additional Formulations

For completeness, we provide additional formulations omitted from the main text, including the detailed construction of the grounding prior, the full definition of region-aware tokenization, and the tensor shapes used throughout the model.

Batch notation. For a minibatch of size B, let the fused patch features be denoted by

$$R \in \mathbb{R}^{B\times N\times D_v}. \tag{17}$$

where B is the batch size, N is the number of image patches, and D_v is the fused visual feature dimension. The pooled question representation is denoted by

$$e_q \in \mathbb{R}^{B\times D_q}. \tag{18}$$

Detailed grounding prior construction. Let the grounding branch output a set of raw masks and confidence scores

$$\{(P_k^{\mathrm{raw}}, s_k^{\mathrm{raw}})\}_{k=1}^{K_0}, \qquad P_k^{\mathrm{raw}} \in [0,1]^{H\times W}, \quad s_k^{\mathrm{raw}} \in [0,1]. \tag{19}$$

We first discard low-confidence predictions using a threshold τ_s, and then retain at most K candidates with the highest confidence:

$$\{(P_k, s_k)\}_{k=1}^{K} = \mathrm{TopK}\left(\{(P_k^{\mathrm{raw}}, s_k^{\mathrm{raw}}) \mid s_k^{\mathrm{raw}} \ge \tau_s\}\right). \tag{20}$$

We set the confidence threshold to $\tau_s = 0.5$. Here, K_0 denotes the sample-dependent number of raw candidate masks produced by the grounding model before confidence filtering, and is therefore not treated as a fixed hyperparameter.

To align the soft masks with patch-based visual encoding, we downsample each mask onto the patch grid. Let Ω_i denote the spatial support of the i-th patch. The patch-wise coverage of the k-th mask is

$$m_{k,i} = \frac{1}{|\Omega_i|}\sum_{(u,v)\in\Omega_i} P_k(u,v), \qquad m_{k,i} \in [0,1]. \tag{21}$$

Stacking over all patches gives the mask vector

$$m_k = [m_{k,1}, \ldots, m_{k,N}] \in [0,1]^N, \qquad M = [m_1, \ldots, m_K] \in [0,1]^{K\times N}. \tag{22}$$

We then aggregate all retained masks into a single question-conditioned patch prior using a confidence-weighted noisy-OR rule:

$$a_i = 1 - \prod_{k=1}^{K}\left(1 - s_k m_{k,i}\right), \qquad a = [a_1, \ldots, a_N] \in [0,1]^N. \tag{23}$$

In batch form, the aggregated prior is

$$a \in [0,1]^{B\times N}. \tag{24}$$

Detailed region token construction. For each retained region, we compute confidence-weighted normalized pooling coefficients

$$\tilde{m}_{k,i} = \frac{s_k m_{k,i}}{\sum_{j=1}^{N} s_k m_{k,j} + \epsilon}, \qquad \sum_{i=1}^{N} \tilde{m}_{k,i} = 1. \tag{25}$$

The pooled region feature is

$$z_k^{\mathrm{reg}} = \sum_{i=1}^{N} \tilde{m}_{k,i} r_i \in \mathbb{R}^{D_v}. \tag{26}$$

To preserve coarse spatial information, we attach a geometry descriptor

$$b_k = \left[x_k^{\min},\ y_k^{\min},\ x_k^{\max},\ y_k^{\max},\ \alpha_k\right] \in \mathbb{R}^5, \tag{27}$$

where the first four entries are normalized box coordinates extracted from the support of P_k, and

$$\alpha_k = \frac{1}{HW} \sum_{u=1}^{H} \sum_{v=1}^{W} P_k(u, v) \tag{28}$$

is the normalized mask area.

The geometry encoder is implemented as a lightweight two-layer MLP:

$$E_{\mathrm{geo}}(b_k) = W_2^{\mathrm{geo}}\, \sigma(W_1^{\mathrm{geo}} b_k + b_1^{\mathrm{geo}}) + b_2^{\mathrm{geo}}, \qquad E_{\mathrm{geo}}(b_k) \in \mathbb{R}^{D_v}, \tag{29}$$

where

$$W_1^{\mathrm{geo}} \in \mathbb{R}^{D_h^{\mathrm{geo}} \times 5}, \qquad W_2^{\mathrm{geo}} \in \mathbb{R}^{D_v \times D_h^{\mathrm{geo}}}.$$

The final region token is

$$\hat{z}_k^{\mathrm{reg}} = W_r z_k^{\mathrm{reg}} + E_{\mathrm{geo}}(b_k), \qquad W_r \in \mathbb{R}^{D_v \times D_v}. \tag{30}$$

Stacking all region tokens gives

$$Z^{\mathrm{reg}} = [\hat{z}_1^{\mathrm{reg}}, \ldots, \hat{z}_K^{\mathrm{reg}}] \in \mathbb{R}^{B \times K \times D_v}. \tag{31}$$

Detailed routed context construction. Given the fused patch tokens r_i, the region prior a_i, and the question summary e_q, we compute routing scores

$$u_i = w^\top \tanh(W_v r_i + W_a a_i + W_q e_q)\,, \tag{32}$$

followed by normalized routing weights

$$\rho_i = \frac{\exp(u_i)}{\sum_{j=1}^{N} \exp(u_j)}. \tag{33}$$

The routed context set is chosen by top-L selection:

$$\mathcal{I} = \mathrm{TopL}(\{\rho_i\}_{i=1}^{N}) = \{i_1, \ldots, i_L\}, \tag{34}$$

where $\rho_{i_\ell} \geq \rho_j$ for all $j \notin \mathcal{I}$. The selected context tokens are modulated by the region prior through a residual gating form:

$$\hat{r}_i = r_i + \lambda a_i W_m r_i = (\mathbf{E} + \lambda a_i W_m)\, r_i, \qquad i \in \mathcal{I}, \tag{35}$$

where λ is a scalar coefficient. This design preserves the original patch feature when a_i is small while amplifying patches that are simultaneously selected by the routing scorer and supported by the grounding prior.

Stacking the routed context tokens gives

$$R^{\mathrm{ctx}} = [\hat{r}_{i_1}, \ldots, \hat{r}_{i_L}] \in \mathbb{R}^{B \times L \times D_v}. \tag{36}$$

Background token and final visual sequence. The remaining patches are compressed into a single background token:

$$z^{\mathrm{bg}} = \frac{\sum_{i \notin \mathcal{I}} (1 - a_i) r_i}{\sum_{i \notin \mathcal{I}} (1 - a_i) + \epsilon} \in \mathbb{R}^{D_v}. \tag{37}$$

In batch form,

$$z^{\mathrm{bg}} \in \mathbb{R}^{B \times 1 \times D_v}. \tag{38}$$

Finally, the structured visual sequence is

$$S_v = [Z^{\mathrm{reg}},\ R^{\mathrm{ctx}},\ z^{\mathrm{bg}}] \in \mathbb{R}^{B \times (K+L+1) \times D_v}. \tag{39}$$

After multimodal projection, the visual prefix becomes

$$H_v = \phi(S_v) \in \mathbb{R}^{B \times (K+L+1) \times D}, \tag{40}$$

where D is the hidden width of the multimodal backbone.

C Additional Experimental Details

C.1 Training Data and Two-Stage Training Setup

Following Cobra and MambaVLM, we adopt the same QA/instruction-tuning mixture for answer generation, consisting of the 665K LLaVA-1.5 mixed dataset, 220K LVIS-Instruct-4V, and 400K LRV-Instruct. This mixture is used throughout training to learn answer generation and multimodal reasoning. To further introduce region-aware supervision, we additionally use conversational image segmentation data from CONVERSEG/CIS. Each CIS sample contains an image, a segmentation prompt, and the corresponding mask. Rather than converting CIS into pseudo-VQA pairs, we use it to directly supervise the intermediate grounding prior in QRouter. Specifically, for a CIS sample, we treat the segmentation prompt as a grounding query and convert the ground-truth mask $M^\star \in \{0,1\}^{H \times W}$ into a patch-level target prior aligned with the visual patch grid:

$$a_i^\star = \frac{1}{|\Omega_i|} \sum_{(u,v) \in \Omega_i} M^\star(u,v), \qquad i = 1, \ldots, N, \tag{41}$$

where Ω_i denotes the spatial support of the i-th visual patch. This produces a target prior $a^\star = [a_1^\star, \ldots, a_N^\star] \in [0,1]^N$, which is directly matched to the predicted question-conditioned prior. Our training follows a two-stage setup. In Stage I, we train the model only on the QA/instruction-tuning mixture for 20k optimization steps, allowing it to first learn stable multimodal answer generation. The best checkpoint from Stage I is then used to initialize Stage II. In Stage II, we continue training for 10k optimization steps on the same QA mixture while additionally incorporating CIS supervision. During this stage, QA batches and CIS batches are sampled with a ratio of 2:1. For QA batches, the model is optimized with the standard autoregressive VQA loss. For CIS batches, we supervise the predicted grounding prior with the target patch-level prior using

$$\mathcal{L}_{\mathrm{CIS}} = \mathcal{L}_{\mathrm{BCE}}(a, a^\star) + \lambda_{\mathrm{dice}}\, \mathcal{L}_{\mathrm{Dice}}(a, a^\star). \tag{42}$$

This stage-wise design allows QA data to continuously supervise answer generation while CIS data improves the quality of the intermediate grounding prior, enabling QRouter to benefit from conversational grounding supervision without treating segmentation itself as the final task.

C.2 Hyperparameters and Implementation Details

Our codebase is implemented in PyTorch with 2.1.2 and CUDA 11.8. Training is conducted on 8 NVIDIA A800 80GB GPUs, and inference is performed on a single NVIDIA A800 80GB GPU. The multimodal projector is implemented as a two-layer MLP with hidden dimension 2048. We use AdamW with cosine decay; detailed optimization hyperparameters are summarized in Table 6. Due to compute constraints, we report the main results from a single run for each model configuration.

Table 6: Hyperparameters and implementation details.

Item	Value
Image resolution	384×384
Visual encoders	DINOv2 + SigLIP
Language backbone	Mamba-2.8B-Zephyr / Mamba-7B
Grounding image encoder	SAM2 Hiera-L
Grounding LLM	Qwen2.5-VL
Optimizer	AdamW
Learning rate	2×10^{-5}
Weight decay	0.1
Warmup ratio	0.03
Learning rate schedule	Cosine decay
Global batch size	128
Projector	Two-layer MLP
Projector hidden dimension	2048
Decoding strategy	Greedy decoding
Temperature	0.0
Top-p	1.0
Beam size	1

Table 7: Training data and stage-wise usage. 665K + 220K + 400K denotes the QA/instruction-tuning mixture, consisting of the 665K LLaVA-1.5 mixed dataset, 220K LVIS-Instruct-4V, and 400K LRV-Instruct.

Dataset	Type	Stage I	Stage II	Purpose
665K + 220K + 400K	VQA supervision	✓	✓	Answer generation
CONVERSEG	Grounding supervision		✓	Grounding prior learning

C.3 Prompt

Following prior efficient VLMs such as Cobra and MambaVLM, we use a simple and unified prompt template for VQA training and inference. Given an image-question pair, the text prompt is formatted as follows:

```
Question: {question}
Answer:
```

For most open-ended benchmarks, the model generates a short free-form answer conditioned on the input image and question. For binary benchmarks such as VSR and POPE, we use the same prompt format but normalize the generated output into binary labels during evaluation. For TallyQA, numerical answers are kept in their generated form and evaluated using the official protocol. For all remaining benchmarks, we follow the prompt templates and evaluation procedures specified by their official implementations, without introducing additional task-specific prompt engineering. Unless otherwise specified, all reported results follow the standard evaluation settings of the corresponding benchmarks.

During inference, we use greedy decoding with no sampling. Predictions are lowercased and stripped of leading and trailing punctuation before benchmark-specific evaluation, following standard answer normalization whenever applicable. For GQA, we append a short-answer instruction after the question to encourage concise responses that better match the benchmark annotations:

```
Question: {question}. Answer the question using a single word or phrase.
Answer:
```

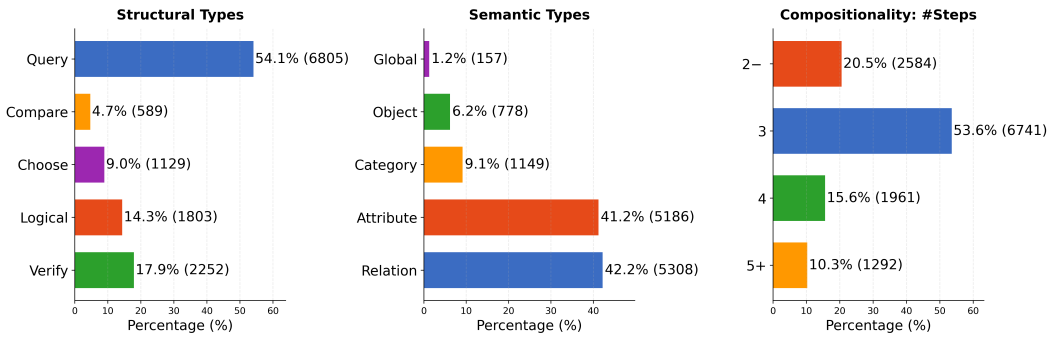


Figure 4: Question distribution of the GQA official test-dev split along three analysis dimensions. The left panel shows the distribution over Structural Types; the middle panel shows the distribution over Semantic Types; and the right panel shows the distribution over Compositionality (#Steps). Here, 2— indicates questions with at most 2 reasoning steps in the semantic program, while 5+ indicates questions with at least 5 steps.

D Additional Results

In this section, we provide additional results that complement the main paper. We first report detailed GQA analyses, encoder ablations, and qualitative comparisons. We then evaluate sensitivity to mismatched grounding, examine the association between grounding quality and answer accuracy, and report a complete online inference latency breakdown.

D.1 Detailed Results on the GQA Dataset

Figure 4 summarizes the distribution of GQA questions on the official test-dev split across three analysis dimensions: structural types, semantic types, and compositionality (#Steps).

In addition to the structural-type analysis in the main paper, Table 8 reports fine-grained results on the semantic-type and compositionality dimensions. For models with repeated runs, we report the mean and sample standard deviation over five independent runs, since these runs constitute a finite sample of training randomness rather than the full population of possible outcomes. These results provide a more detailed view of where the gains of QRouter come from. Notably, the improvements are not uniform across all semantic categories and compositionality groups. QRouter-B achieves its largest gains on relation-centric questions and on categories that require stronger contextual grounding, while it is weaker on some object- and attribute-focused questions. A similar trend appears along the compositionality axis: QRouter is more beneficial for questions with longer reasoning programs, whereas it underperforms Cobra in the $\leq$ 2-step group. This suggests that question-conditioned region routing is particularly helpful when answering depends on structured evidence selection and multi-step reasoning, but may be less advantageous for simpler questions that can often be resolved from immediately salient local evidence.

D.2 Additional Ablation Study

Table 9 shows that the full model performs best on all three benchmarks, confirming the benefit of combining dual visual encoders with a stronger grounding encoder. Using only SigLIP or only DINOv2 leads to clear performance drops, suggesting that the two visual encoders provide complementary information for region routing and reasoning. Replacing SAM2 with Mask2Former Cheng et al. (2022) also reduces performance, indicating that stronger grounding quality further improves QRouter, although the method remains effective with a weaker grounding encoder.

Table 8: Fine-grained performance comparison on GQA along two additional official dimensions: Semantic Types and Compositionality (#Steps). The compositionality groups are defined by the number of reasoning steps in the semantic program, where 2− indicates questions with ≤ 2 steps and 5+ indicates questions with ≥ 5 steps. For models with repeated runs, we report the mean and standard deviation over five runs. For CNN, LSTM, and CNN+LSTM, we report the original single-run numbers from prior work. Results are reported on the official test-dev split.

Model	Backbone	Semantic Types					Compositionality: #Steps			
		Global	Object	Category	Attribute	relation	2−	3	4	5+
CNN	-	8.64	47.33	3.56	22.66	11.60	-	-	-	-
LSTM	-	27.22	74.33	22.33	48.28	33.24	-	-	-	-
CNN+LSTM	-	56.57	75.90	37.49	50.91	39.45	-	-	-	-
Gemma3	Gemma3-4B	28.62 ±1.84	24.84 ±1.92	51.05 ±1.65	34.40 ±2.01	26.25 ±1.77	42.50 ±1.88	33.80 ±2.12	23.60 ±1.69	9.21 ±1.95
SmolVLM2	SmolLM2-1.7B	34.50 ±1.73	47.80 ±2.05	57.20 ±1.81	51.50 ±1.68	45.45 ±1.94	57.02 ±1.79	52.00 ±2.08	43.01 ±1.86	26.84 ±1.62
Cobra	Mamba-2.8B	62.42 ±1.24	**89.33** ± 1.15	52.65 ±1.32	**67.47** ± 1.18	55.52 ±1.27	**75.08** ± 1.12	55.91 ±1.35	57.93 ±1.22	77.32 ±1.19
QRouter-B	Mamba-2.8B	**80.89** ± 1.16	86.76 ±1.28	**64.06** ± 1.13	61.61 ±1.31	**72.32** ± 1.25	65.40 ±1.17	**67.84** ± 1.21	**64.81** ± 1.34	**80.34** ± 1.29

Table 9: Ablation on visual and grounding encoders. We study the contribution of the dual visual encoders and the grounding encoder by comparing the full model with three variants: using only SigLIP, using only DINOv2, and replacing SAM2 with a weaker Mask2Former grounding encoder. Results are reported on representative benchmarks.

Visual Encoders	Grounding Encoder	GQA	VSR	POPE	Notes
DINOv2 + SigLIP	SAM2 Hiera-L	**68.2**	**68.4**	**91.3**	Full model
SigLIP only	SAM2 Hiera-L	60.3	53.6	86.4	Remove DINOv2
DINOv2 only	SAM2 Hiera-L	59.9	53.0	89.1	Remove SigLIP
DINOv2 + SigLIP	Mask2Former	66.4	64.7	90.6	Replace SAM2

D.3 Additional Qualitative Examples

Figure 5 provides additional qualitative comparisons between QRouter and Cobra on grounding-sensitive questions. Across these examples, QRouter consistently highlights more question-relevant regions and produces answers that are better aligned with the underlying visual evidence. The improvements are especially clear in cases that require subtle local grounding, functional understanding, or safety-related reasoning, where incorrect focus often leads to plausible but unsupported answers.

In the first and third examples, QRouter correctly identifies small but decisive visual cues, such as the bag in the bicycle basket and the bowl of soup next to the coffee cup, while Cobra is distracted by visually salient but irrelevant alternatives. In the second and fourth examples, QRouter also shows stronger region-level grounding for questions involving object use and intended function, correctly focusing on the hamburger held by the boy and the upside-down stop sign, respectively. These examples further support our central claim that explicitly organizing question-relevant visual evidence before reasoning leads to more reliable fine-grained VQA behavior.

Error Analysis on Exact-Match Evaluation. We also observe that some errors made by both QRouter and Cobra under exact-match evaluation do not necessarily reflect genuine failures in visual grounding or reasoning. Instead, they often arise from surface-form mismatches between the model output and the benchmark annotation. Typical cases include synonym variation (e.g., bicycle vs. bike), singular-plural mismatch (e.g., tree vs. trees), and closely related human descriptors (e.g., woman or lady vs. women). In such cases, the predicted answer is semantically consistent with the image and the question, but is still counted as incorrect due to the strict exact-match protocol. We therefore view these cases primarily as an artifact of answer normalization and annotation granularity, rather than as evidence of deficient question-relevant grounding. This observation suggests that the current evaluation may underestimate model performance on fine-grained VQA when semantically equivalent answers are expressed with different lexical forms.

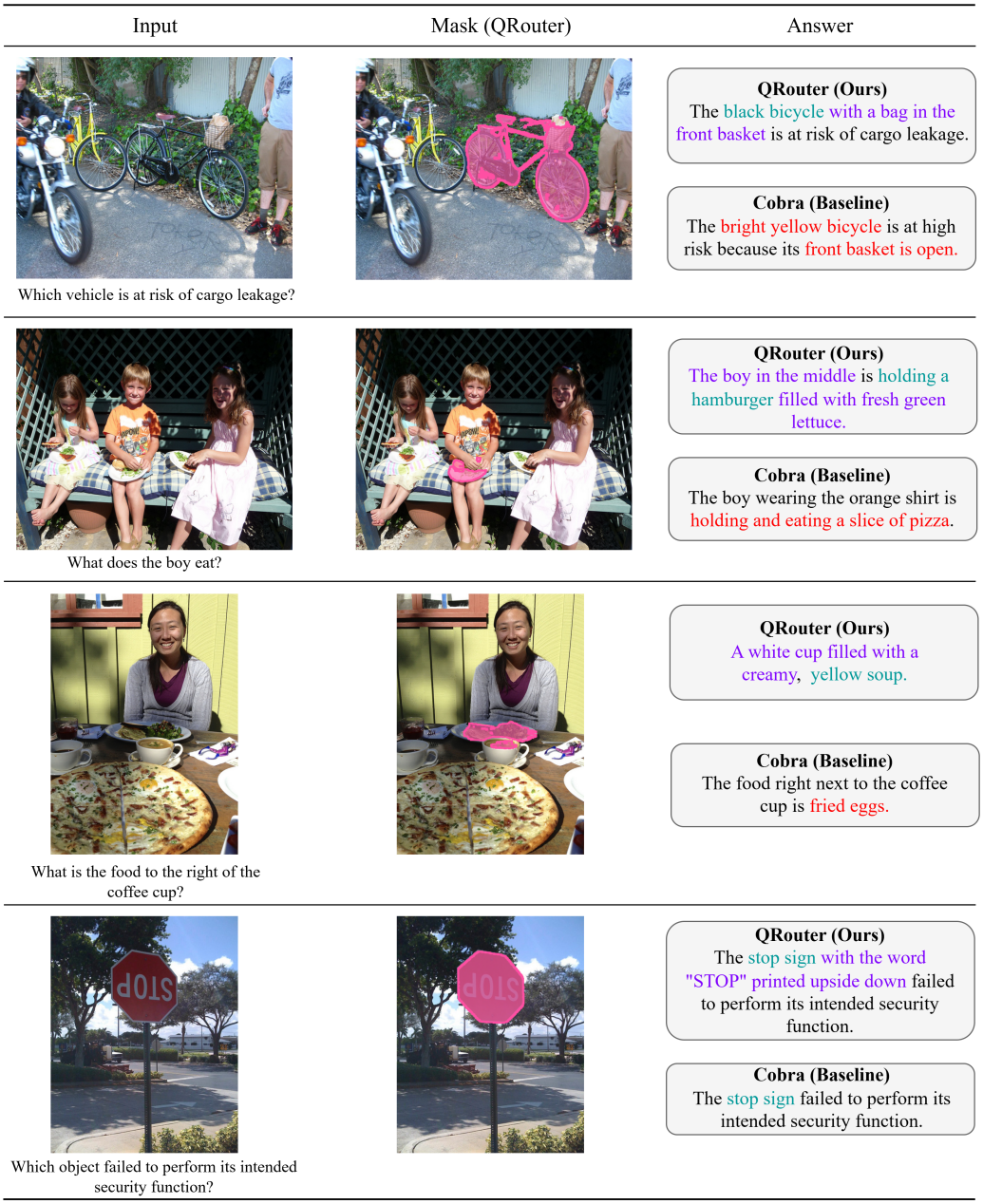


Figure 5: Additional qualitative examples on grounding-sensitive questions. QRouter produces question-relevant masks and more accurate answers than Cobra, especially for cases involving subtle local evidence, functional understanding, and safety-related reasoning. Text in green indicates correct answers, red denotes inaccurate predictions, and purple highlights a more detailed scene understanding.

D.4 Sensitivity to Mismatched Grounding Queries

To test whether QRouter depends on the semantic alignment between the VQA question and the grounding prior, we conduct a question-shuffled grounding control on GQA test-dev. For each example, we retain the original image and answer-generation question, but replace the query supplied to the grounding module with the question from the next example under a fixed cyclic ordering. Self-matches are explicitly excluded. All other model components and inference settings remain unchanged. This intervention preserves the grounding archi-

Table 10: Question-shuffled grounding control on GQA test-dev. The original image and answer-generation question remain unchanged, while only the grounding query is randomly permuted across examples.

Matched Grounding Acc. (%) ↑	Question-Shuffled Grounding Acc. (%) ↑	Δ Acc. (pp)
68.2	31.8	−36.4

tecture and model-generated mask format while selectively disrupting question-grounding alignment.

As shown in Table 10, disrupting question-grounding alignment reduces accuracy from 68.2% to 31.8%, a decrease of 36.4 percentage points. The grounding module and structured token pipeline remain present under both conditions. Therefore, the gain cannot be explained solely by additional grounding capacity or the presence of region tokens. The grounding prior must also be semantically aligned with the question being answered. The remaining accuracy may reflect language priors, global and background context, or visual evidence that is incidentally preserved after shuffling.

We complement this dataset-level intervention with the qualitative analysis in Figure 6. Here, the mismatched queries are related to the current image but irrelevant to the evidence required by the original VQA question. These semantically plausible mismatches allow us to examine how an incorrect grounding prior propagates through the structured visual sequence.

Figure 6 shows that semantically mismatched grounding priors can affect QRouter in two different ways. In some cases, the model still produces the correct answer even when the mask is generated from an answer-irrelevant query. For example, QRouter correctly answers the table next to the bed and the keyboard used for typing. These cases suggest that QRouter does not collapse into a pure grounding bottleneck. Although grounding priors guide visual evidence organization, the RegionTokenizer still preserves complementary information through routed context tokens and the background token. In addition, the answering module continues to receive the original VQA question, which can provide a corrective signal when the mismatched prior conflicts with the target question. However, the figure also shows clear failure cases where incorrect priors mislead the model. When the mismatched query highlights plausible but answer-irrelevant regions, the induced structured visual sequence becomes less aligned with the evidence required by the original question. As a result, QRouter can be biased toward the wrong visual support and produce incorrect answers, such as predicting bananas instead of the apple, the fender instead of the cat, or the phone keypad instead of the banana. These failures indicate that the semantic prior from the grounding module is indeed transmitted through the structured visual tokens and can propagate to downstream reasoning.

Together, the quantitative accuracy drop and qualitative failure cases provide controlled evidence that question-grounding alignment is functionally important to QRouter. However, this intervention does not establish that correct grounding is sufficient for every answer, or that every prediction error is caused exclusively by grounding. Instead, it shows that disrupting grounding alignment substantially impairs the same model while revealing the complementary role of residual visual and linguistic context.

D.5 Grounding Quality and Answer Accuracy

We further examine whether grounding quality is associated with downstream answer accuracy. We evaluate the full QRouter on all 2,373 examples from the VizWiz-VQA-Grounding test set. For each example, grounding IoU is computed between the predicted grounding support and the human-annotated answer-grounding mask. We then divide the examples into five fixed-width IoU intervals and evaluate answer accuracy within each interval. Within each IoU interval, the 95% confidence intervals are estimated using 10,000 nonparametric percentile-bootstrap resamples with replacement.

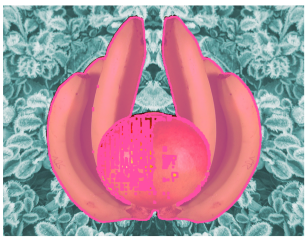
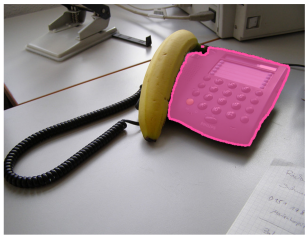
Input	Mask (From Mismatched Query)	Answer
 What piece of furniture is placed next to the bed?	 Are the windows in the room open or closed? (Mismatched)	QRouter : Table Ground Turth : Table
 Which fruit is most likely to roll because of its round shape?	 What is the long, curved fruit in the picture? (Mismatched)	QRouter : Bananas Ground Turth : Apple
 What part of the laptop is used for typing?	 What animal is lying above the laptop? (Mismatched)	QRouter : keyboard Ground Turth : The keyboard
 What animal is standing on the car?	 Which part of the car reflects light? (Mismatched)	QRouter : Fender Ground Turth : Cat
 Which object performs the substitute function in the image?	 Which part of the phone contains the number buttons? (Mismatched)	QRouter : Phone keypad Ground Turth : Banana

Figure 6: Qualitative intervention analysis with mismatched grounding queries. For each example, the input column shows the original image and VQA question, while the middle column shows the mask generated from a VQA-style query that is related to the image but irrelevant to the evidence required by the original question.

Table 11: Answer accuracy stratified by grounding IoU on all 2,373 examples in the VizWiz-VQA-Grounding test set. Confidence intervals are computed by nonparametric bootstrap within each IoU interval.

			Official Soft Score (%)		Most-Common Match (%)	
Grounding IoU	Num.	Mean IoU (%)	mAcc.	95% CI	mAcc.	95% CI
$[0.0, 0.2)$	54	18.73	25.74	$[19.63, 32.22]$	14.81	$[7.70, 26.60]$
$[0.2, 0.4)$	141	33.58	36.45	$[32.34, 40.64]$	17.73	$[12.31, 24.86]$
$[0.4, 0.6)$	255	58.91	61.33	$[58.51, 64.20]$	19.61	$[15.20, 24.92]$
$[0.6, 0.8)$	387	69.72	69.48	$[67.08, 71.86]$	48.06	$[43.13, 53.03]$
$[0.8, 1.0]$	1,536	95.02	88.21	$[87.35, 89.06]$	69.21	$[66.85, 71.46]$

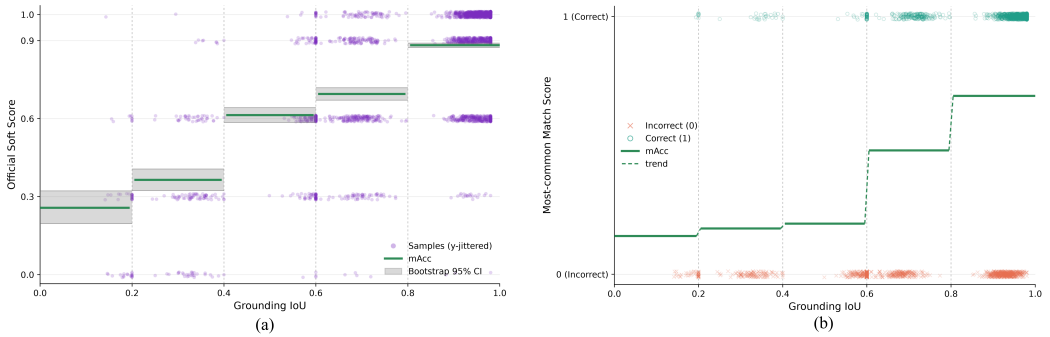


Figure 7: Sample-level relationship between grounding IoU and answer accuracy on the VizWiz-VQA-Grounding test set. (a) Official VQA soft scores. Points are vertically jittered for visibility; green segments indicate the mean score in each IoU interval, and shaded regions show bootstrap 95% confidence intervals. (b) Binary most-common-answer matches. Correct and incorrect predictions are vertically jittered around one and zero, respectively. Green segments indicate the mean match accuracy in each interval. Jitter is used only for visualization and does not affect any reported statistic.

We report two complementary answer metrics. The official soft score follows the standard VQA evaluation protocol and assigns partial credit according to agreement with the ten human answers. The resulting per-example scores take values in $\{0, 0.3, 0.6, 0.9, 1.0\}$. We additionally report a stricter most-common-answer match score, which assigns a value of one only when the normalized model prediction matches the most frequent normalized human answer, and zero otherwise. The latter metric verifies that the observed relationship is not an artifact of partial-credit scoring.

Table 11 shows that both answer metrics increase monotonically with grounding quality. From the lowest to the highest IoU interval, the official soft score increases from 25.74% to 88.21%, a gain of 62.47 percentage points. The stricter most-common-answer match score increases from 14.81% to 69.21%, a gain of 54.40 points. The confidence intervals of the lowest and highest intervals do not overlap under either metric. Furthermore, 1,923 examples (81.0%) achieve a grounding IoU of at least 0.6, including 1,536 examples (64.7%) in the highest-IoU interval.

Figure 7 provides the corresponding sample-level view. Although correct and incorrect answers occur throughout the IoU range, high-IoU examples contain a visibly greater proportion of correct predictions. This pattern is consistent under both the partial-credit VQA metric and the stricter binary metric. The result therefore connects the evidence-selection measurements in Appendix B.2 with downstream VQA performance: examples with better localized evidence tend to yield more accurate answers.

This analysis provides empirical evidence of a positive association between grounding quality and answer accuracy. It does not, by itself, establish a strict causal relationship. Factors such as question difficulty, image quality, and answer ambiguity may influence both grounding

Table 12: Online inference latency using the same NVIDIA A800 setup as Table 4. All values are average seconds per image–question pair. Complete online latency includes both grounding-prior construction and the post-grounding VQA pipeline. Lower is better.

Model	Grounding (s/sample) ↓	Post-grounding VQA (s/sample) ↓	Complete Online (s/sample) ↓
QRouter-B	0.4386	0.0862	0.5248
QRouter-L	0.4386	0.1418	0.5804

and answering. Together with the controlled question-shuffled experiment in Appendix D.4, however, the results support question-aligned grounding as a measurable and functionally important intermediate mechanism in QRouter.

D.6 Online Inference Efficiency Breakdown

To clarify the computational scope of the latency evaluation, we decompose online inference into two stages. The grounding stage includes online Qwen2.5-VL and SAM2 Hiera-L inference for constructing the question-conditioned grounding prior. The post-grounding VQA stage covers the remaining QRouter pipeline, including visual feature processing, region tokenization and routing, multimodal projection, and answer decoding. The latency values reported for QRouter in Table 4 correspond to this post-grounding stage.

The grounding stage was profiled over all 12,578 image–question pairs in GQA test-dev and required 5,516.71 seconds in total, corresponding to 0.4386 seconds per sample. Including this stage, the complete online latency is 0.5248 seconds for QRouter-B and 0.5804 seconds for QRouter-L, as summarized in Table 12. Both variants therefore remain below 0.6 seconds per sample under fully online inference. Moreover, the two variants share the same grounding front end. Scaling the reasoning backbone from 2.8B to 7B parameters increases the complete latency by only 0.0556 seconds. These results show that the compact routed representation keeps the downstream reasoning stage efficient, while making the cost of question-conditioned evidence selection explicit.

E Comparison with Agentic Visual Search

We further clarify the relationship between QRouter and two recent works that share our motivation of identifying question-relevant visual evidence, while showing how our formal relevance-density analysis and empirical validation provide a complementary analytical perspective on this broader principle. However, their multi-turn agentic search paradigm differs from QRouter's fixed-budget evidence organization, so we discuss them conceptually rather than treat them as directly comparable baselines.

V* Wu & Xie (2024) performs adaptive visual search by recursively exploring high-resolution images and storing selected crops in a Visual Working Memory. PixelEyes Gong et al. (2026) extends this multi-turn paradigm using mask-guided cropping, Semantic-Region BFS, and a specialized segmentation tool. Both methods can adaptively allocate computation, but require sequential tool interactions and variable-length reasoning trajectories.

In contrast, QRouter does not formulate VQA as an iterative search process. It constructs question-conditioned grounding priors once and reorganizes the visual input into a fixed sequence of region, routed-context, and background tokens. This design avoids multi-turn planning, backtracking, growing visual contexts, and search-trajectory supervision, while providing predictable inference cost. Moreover, the background token preserves complementary global information that may be discarded by crop-based search. QRouter therefore offers a complementary fixed-budget solution for fine-grained VQA, whereas agentic methods are particularly suited to extreme, zero-hint high-resolution search requiring iterative refinement.