SEEING IS NOT RELATING: LEARNING RELATIONAL REPRESENTATIONS FOR MULTI-IMAGE UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-image understanding requires vision–language models to connect visual information across images. Controlled diagnostics across four model families show that models can correctly recognize task-relevant local details, including attributes and within-image relations, yet still fail at cross-image comparison. Complementary information interventions show that providing the correct relation target substantially improves task performance, highlighting the importance of reliable cross-image relations. Motivated by these findings, we construct structured supervision covering entities, attributes, within-image predicates, and cross-image comparisons to learn relational features that complement pretrained visual content. We propose Relation-Aware Dual-Stream Fusion (RDSF), which learns a dedicated relation stream, exchanges context across images, and integrates the resulting features with native visual tokens through token-aligned residual fusion. The pretrained backbone remains frozen throughout training. Across four Qwen3.5 scales, RDSF improves the six-benchmark mean by 1.99–6.51 percentage points. Additional evaluations on InternVL3.5 and LLaVA-OV1.5 show gains in the four-benchmark mean, supporting the applicability of complementary relational learning across model architectures.

1 INTRODUCTION

Recent vision–language models have made progress in multi-image understanding (Wang et al., 2025; Li et al., 2025b). While single-image and multi-image tasks share the need to recognize visual entities and attributes within each image, multi-image understanding additionally requires establishing relations across images. Tasks such as object correspondence, attribute comparison, and joint condition verification depend on connecting information from different images, beyond recognizing their contents individually (Fu et al., 2024; Jiang et al., 2024; Suhr et al., 2019). This distinction motivates a targeted approach: building on pretrained visual representations while explicitly learning complementary features for cross-image relational understanding. We investigate whether a dedicated relation stream can augment a pretrained vision–language model's multi-image capabilities while keeping its existing backbone frozen.

To examine the role of cross-image relations, we combine information interventions with independent local semantic checks across four model families. Even on examples where models correctly read all candidate codes, identifying the correct relation target remains challenging: Qwen3.5-27B, for example, achieves an accuracy of approximately 56%. Providing the correct relation target substantially improves final-answer accuracy, whereas supplying the candidate-to-code mapping offers little benefit (Figure 1(d)). Figure 1(a–c) illustrates how an incorrect cross-image correspondence can lead to a wrong answer despite correctly read candidate codes, and how supplying the correct correspondence enables a correct response. We further probe task-relevant attributes and within-image relations through independent local questions. Cross-image comparison errors persist even when models answer all these questions correctly (Figure 1(d); Appendix A). Together, these diagnostics show that correctly recognizing task-relevant information within each image does not ensure reliable cross-image comparison. This motivates learning complementary relational representations that help pretrained vision–language models connect and use visual content across images.

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.

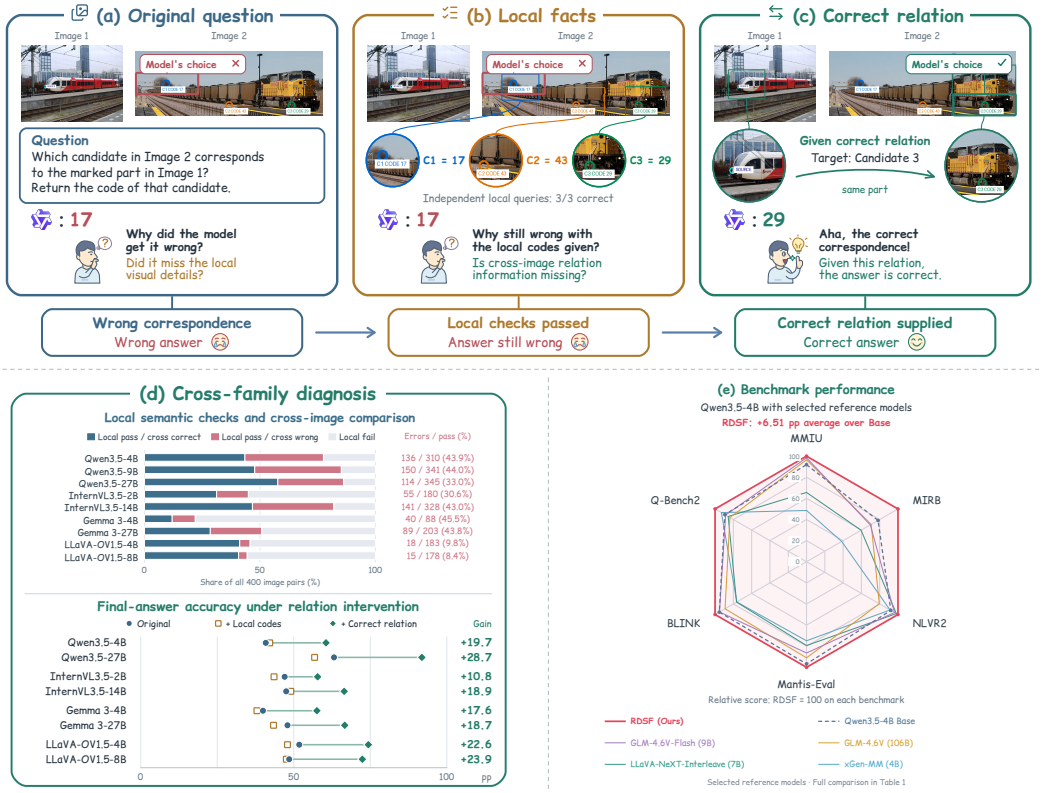

Figure 1: **Cross-image relation diagnosis and RDSF performance.** (a–c) Independent prompts for the same example: original, supplied candidate-code mappings, and a supplied correct relation. (d) Local semantic checks and cross-image comparison (top), and answer accuracy under information interventions on candidate-code local-pass subsets (bottom). (e) Six-benchmark performance of Qwen3.5-4B with and without RDSF and selected reference models, normalized to RDSF = 100 per benchmark.

To strengthen cross-image relational understanding, we construct structured supervision covering entities, attributes, within-image predicates, and cross-image entity and relation comparisons. Building on this supervision, we propose Relation-Aware Dual-Stream Fusion (RDSF) to learn complementary relational features and integrate them with pretrained visual representations (Figure 2). RDSF combines a dedicated relation stream with the backbone's native content stream through cross-image context exchange and token-aligned residual fusion. We adopt a two-stage procedure: first learning the relation module, then training its fusion interface with question–answer supervision, while keeping the pretrained backbone frozen throughout.

We evaluate RDSF on six multi-image understanding benchmarks with Qwen3.5 backbones ranging from 2B to 27B parameters. RDSF improves the six-benchmark mean at all four scales, with gains of 1.99–6.51 percentage points over the corresponding Base models (Table 1). RDSF also improves InternVL3.5 and LLaVA-OV1.5 in our cross-architecture evaluation, supporting its applicability to different backbone architectures. Ablation studies further show the benefits of structural pretraining, cross-image context, and token-aligned fusion. Our main contributions are as follows:

1. Relational diagnosis. Across four model families, we show that correctly recognizing task-relevant attributes and within-image relations does not ensure reliable cross-image comparison. Providing the correct relation target substantially improves task performance, highlighting the importance of reliable cross-image relations.
2. Relational learning and integration. We construct structured supervision covering entities, attributes, and within- and cross-image relations, and propose RDSF to integrate the learned

features with native visual content. Cross-image context exchange and token-aligned residual fusion enable this integration while keeping the pretrained backbone frozen.

3. Evaluation across scales and architectures. RDSF improves average multi-image performance across multiple model scales and three backbone families. Ablations examine the contributions of structured pretraining, cross-image context, and fusion granularity.

2 Related Work

Multi-Image Understanding. Work on multi-image understanding has pursued improvements in instruction data, model architecture, and preference learning. Mantis (Jiang et al., 2024) develops co-reference, comparison, reasoning, and temporal understanding through multi-image instruction tuning, while LLaVA-Interleave (Li et al., 2025a) organizes multi-image, video, and 3D instruction data in a common interleaved format. For longer image sequences, mPLUG-Owl3 (Ye et al., 2025) introduces Hyper Attention to integrate visual information efficiently. Preference optimization provides another route: MIA-DPO (Liu et al., 2025) augments single-image examples with distracting images and uses attention-aware response selection to construct preference pairs; CcDPO (Li et al., 2025b) combines preference supervision over multi-image context and local visual cues.

Compositional Visual–Language Understanding. Compositional understanding concerns how entities, attributes, and relations combine to determine meaning. Winoground (Thrush et al., 2022) and ARO (Yuksekgonul et al., 2023) expose failures of vision–language models to distinguish changes in these combinations. To make training sensitive to such distinctions, NegCLIP (Yuksekgonul et al., 2023) introduces composition-aware hard negatives, and SPARCL (Li & Li, 2025) synthesizes image–text examples with subtle semantic variations. Our diagnostics examine a related question in multi-image reasoning: whether models can identify cross-image relations on examples where they pass the local semantic checks in Appendix A.4.

Structured Relational Representation Learning. Scene graphs offer explicit entity and relation supervision. SGVL (Herzig et al., 2023) trains visual tokens to predict scene-graph information and derives fine-grained caption supervision from the same graphs. Structure-CLIP (Huang et al., 2024) combines graph-guided negative sampling with an encoder that incorporates scene-graph knowledge. RDSF builds on structured supervision to learn a dedicated relation stream for multi-image question answering. The learned features exchange context across images and augment the native visual tokens through token-aligned fusion, with the pretrained backbone kept frozen.

3 Relation-Aware Dual-Stream Fusion

We propose **Relation-Aware Dual-Stream Fusion (RDSF)**, which constructs complementary content and relation streams from a single forward pass through the visual encoder (Figure 2). The relation stream learns representations through structured supervision and exchanges context across images (Section 3.1); token-aligned residual fusion integrates these features with the native visual tokens (Section 3.2). We first train the relation module, then freeze it and train the fusion interface with question–answer supervision. All pretrained backbone parameters remain frozen throughout.

3.1 Visual Relation Modeling

We learn a shared relation space through structured supervision and use it to exchange cross-image context while preserving the spatial layout of visual features.

Entity and Predicate Representations. Given $M \geq 2$ images and a backbone b, let $\mathcal{L}_b$ denote the selected intermediate layer or layers of its frozen visual encoder. We collect their spatially aligned features for image I_i as $Z_i^{(\mathcal{L}_b)} \in \mathbb{R}^{P_i \times d_v^{(b)}}$, concatenating along the channel dimension when multiple layers are used; $d_v^{(b)}$ denotes the resulting readout width. A mapping shared across input images projects these features into a relation space of dimension d_r:

$$X_i = \psi_\phi\left(Z_i^{(\mathcal{L}_b)}\right) \in \mathbb{R}^{P_i \times d_r}. \tag{1}$$

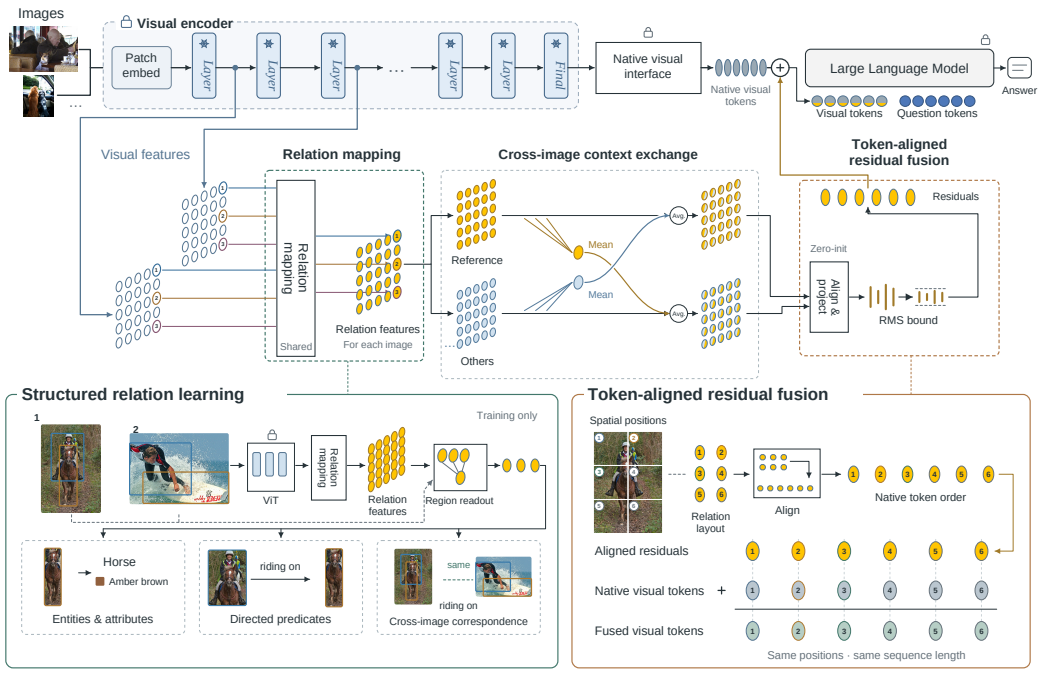

Figure 2: RDSF architecture. Aligned relation residuals augment native visual tokens without extending the sequence. Lower panels detail structured relation learning and token-wise fusion.

For multi-layer readouts, ψ_ϕ includes layer-specific projections and their fusion. During structured training, annotated regions and ordered entity pairs provide entity and predicate readouts:

$$u_a = \mathrm{Pool}(X_i, a), \qquad e_{a\to b} = \eta_\omega([u_a; u_b; u_b - u_a; u_a \odot u_b])\,. \tag{2}$$

Here, Pool aggregates patch features within region a using normalized weights, and η_ω is a predicate MLP. Ordered inputs preserve subject–object roles. These readouts transfer structured supervision to the underlying dense patch representations.

Structured Relation Learning. As illustrated in the lower-left inset of Figure 2, supervision covers both within-image structure and cross-image correspondence:

$$\mathcal{L}_{\mathrm{rel}} = \mathcal{L}_{\mathrm{entity}} + \mathcal{L}_{\mathrm{predicate}} + \mathcal{L}_{\mathrm{cross}}. \tag{3}$$

The three components supervise object categories and attributes; directed predicates, inverse relations, and symmetry consistency; and cross-image category/attribute agreement, predicate correspondence, and instance distinctions, respectively. Each component is a weighted combination of its constituent objectives. We train ψ_ϕ, η_ω, and the auxiliary heads while keeping the backbone frozen. At inference, only the learned dense patch features are used for cross-image computation; region annotations, the predicate encoder, and auxiliary heads are not required. Appendix B.1 details the projection networks, region pooling, auxiliary heads, and constituent training objectives.

Cross-Image Message Construction. The first input image serves as the reference and exchanges context with every remaining image. We summarize each image over its valid patch set Ω_i:

$$\bar{X}_i = \frac{1}{|\Omega_i|} \sum_{p\in\Omega_i} X_{i,p}. \tag{4}$$

The resulting relation maps are

$$R_{i,p} = \begin{cases} \dfrac{1}{2}\left(X_{1,p} + \dfrac{1}{M-1}\displaystyle\sum_{j=2}^{M} \bar{X}_j\right), & i = 1, \\[2ex] \dfrac{1}{2}\left(X_{i,p} + \bar{X}_1\right), & i > 1. \end{cases} \tag{5}$$

Thus, reference patches receive the mean context of the other images, while each remaining image receives reference context. The resulting relation maps $\mathcal{R} = \{R_i\}_{i=1}^{M}$ preserve patch positions and are passed with their validity masks to token-aligned fusion (Section 3.2).

3.2 Token-Aligned Dual-Stream Fusion

RDSF integrates relation features through the backbone's native visual interface. As illustrated in the lower-right inset of Figure 2, we align the relation maps from Section 3.1 with the native visual tokens and add bounded residuals without extending the visual sequence. The fused tokens replace the native visual tokens, while the question follows the backbone's original text pathway.

Token-Aligned Relation Projection. Let $V_i \in \mathbb{R}^{K_i \times d_b}$ denote the native visual tokens of image I_i in backbone b. We construct compatible relation residuals as

$$\widetilde{R}_i = \mathcal{A}_b(R_i), \qquad \Delta V_i = \mathcal{P}_\theta(\widetilde{R}_i) \in \mathbb{R}^{K_i \times d_b}. \tag{6}$$

The alignment operator $\mathcal{A}_b$ organizes relation features according to the native visual-token layout, and the trainable projector $\mathcal{P}_\theta$ maps them into the corresponding feature space. Together, they produce relation residuals that match the native visual tokens in spatial correspondence, sequence order, and channel dimension, enabling token-wise fusion without altering the original visual sequence. Implementation details are provided in Appendix B.2.

Norm-Bounded Residual Fusion. Let V and ΔV collect the native tokens and relation residuals across images, and let Ω index valid visual positions. Defining $\mathrm{RMS}_\Omega(D) = \|D_\Omega\|_F / \sqrt{|\Omega| d_b}$, we use the following fusion rule during training:

$$\widehat{\Delta V} = \frac{\gamma \Delta V}{\max(1, \mathrm{RMS}_\Omega(\gamma \Delta V)/\tau)}, \qquad V^* = V + \widehat{\Delta V}. \tag{7}$$

We first scale the relation residual ΔV by the learnable coefficient γ. If the scaled residual's RMS exceeds τ, the denominator uniformly rescales it to have RMS τ; otherwise, it remains unchanged. The bounded residual is then added to the original visual tokens V to obtain V^*. We zero-initialize the projector's output layer so that fusion starts from the original visual tokens, with $V^* = V$. Training settings and inference-time residual scaling are detailed in Appendix C.2.

Relation-to-Language Alignment. In the second stage, the learned relation branch and pretrained backbone remain frozen. We train only the fusion projector, including its relation embedding, and γ, with answer gradients propagated through the frozen language model. Training minimizes the mean answer-token cross-entropy with regularization, using the settings in Appendix C.2.

4 Experiments

4.1 Experimental Setup

Models and Training. Our main evaluation uses Qwen3.5-2B, 4B, 9B, and 27B (Qwen Team, 2026). For cross-architecture comparisons, we additionally evaluate InternVL3.5-2B, 8B, and 38B (Wang et al., 2025), and LLaVA-OV1.5-4B and 8B (An et al., 2025). We train a separate relation stream and token-aligned fusion module for each backbone. The pretrained vision encoder, native visual interface, and language decoder remain frozen throughout training. We follow the two-stage training procedure in Section 3. Data and training details are in Appendix C.

Benchmarks and Metrics. Our main evaluation covers six benchmarks addressing complementary aspects of multi-image understanding. MMIU (Meng et al., 2025) and Mantis-Eval (Jiang et al., 2024) assess multi-image understanding across diverse tasks; MIRB (Zhao et al., 2024) and NLVR2 (Suhr et al., 2019) cover cross-image and compositional reasoning; and BLINK (Fu et al., 2024) and Q-Bench2 (Zhang et al., 2024) assess visual perception and low-level image comparison. We report benchmark accuracy and the unweighted six-benchmark mean; evaluation protocols are detailed in Appendix C.4.

Baselines. For each evaluated backbone, Base denotes the original model without RDSF. We also compare with leading open-source general-purpose LVLMs, including recent releases such as Qwen3.8 and Gemma 4, alongside models enhanced for multi-image understanding, such as Mantis-Idefics2 (Jiang et al., 2024) and mPLUG-Owl3 (mPLUG Team, 2024). The external comparison models for the main evaluation are listed in Table 1.

4.2 Main Results

Improvements over the Original Backbones. Table 1 shows broad improvements over the original Qwen3.5 backbones, including a gain of 14.14 percentage points on MIRB for the 4B model. These results demonstrate that relation-aware augmentation strengthens multi-image understanding while keeping the pretrained backbones frozen.

Comparison with Existing LVLMs. Table 1 shows RDSF's competitive performance against general-purpose and multi-image-enhanced LVLMs. The 4B model outperforms GLM-4.6V (106B) on all six benchmarks, demonstrating strong multi-image performance with a compact backbone.

Table 1: **Comparison on six multi-image benchmarks.** Scores are percentages; Mean is the unweighted six-benchmark average. **Bold** and underlining mark the highest and second-highest reported values in each column among the displayed models. Shaded rows use RDSF; parentheses give the change from the corresponding Base in percentage points.

Model	Params	MMIU	MIRB	NLVR2	Mantis-Eval	BLINK	Q-Bench2	Mean (6)
General-purpose open-source LVLMs								
Qwen3.8	27B	58.30	52.43	84.46	73.73	60.39	67.20	66.09
Gemma 4	12B	47.89	44.40	76.73	69.59	59.92	71.80	61.72
Gemma 4	26B-A4B	52.49	45.54	79.72	71.89	60.76	77.00	64.57
Molmo2-O	7B	51.70	50.38	93.91	65.90	50.50	78.40	65.13
GLM-4.6V-Flash	9B	53.66	45.32	84.80	61.75	58.76	70.30	62.43
GLM-4.6V	106B	52.66	45.94	70.19	64.98	55.02	66.60	59.23
LLaVA-OneVision-2	8B	51.14	58.11	85.90	68.20	61.91	74.20	66.58
Idefics3	8B	30.49	27.09	72.21	41.01	42.35	43.20	42.73
InternVL3.5	241B-A28B	61.30	60.77	**94.45**	77.00	61.13	79.60	72.37
Multi-image-enhanced LVLMs								
Mantis-Idefics2	8B	45.60	36.12	89.71	57.14	49.05	74.80	58.74
mPLUG-Owl3 (241101)	7B	51.05	40.80	90.80	60.37	51.50	77.70	62.04
LLaVA-NeXT-Interleave	7B	35.77	38.77	86.62	56.68	47.24	68.10	55.53
xGen-MM (BLIP-3 Interleave)	4B	26.47	24.96	80.55	53.46	47.08	74.00	51.09
Qwen3.5: original backbones and RDSF								
Qwen3.5 (Base)	2B	39.31	29.91	65.45	54.84	48.24	52.00	48.29
+ RDSF (Ours)	2B	44.57 (+5.26)	33.95 (+4.04)	75.23 (+9.77)	57.60 (+2.76)	50.13 (+1.89)	64.40 (+12.40)	54.31 (+6.02)
Qwen3.5 (Base)	4B	50.12	50.84	81.30	69.12	58.92	71.00	63.55
+ RDSF (Ours)	4B	54.52 (+4.41)	64.98 (+14.14)	88.19 (+6.89)	71.43 (+2.30)	61.76 (+2.84)	79.50 (+8.50)	70.06 (+6.51)
Qwen3.5 (Base)	9B	58.97	56.73	81.94	75.12	66.33	77.90	69.50
+ RDSF (Ours)	9B	61.58 (+2.61)	**65.28** (+8.55)	90.80 (+8.86)	80.65 (+5.53)	67.02 (+0.68)	81.70 (+3.80)	74.50 (+5.00)
Qwen3.5 (Base)	27B	**64.20**	58.97	92.87	78.80	69.07	83.10	74.50
+ RDSF (Ours)	27B	63.75 (-0.44)	65.10 (+6.13)	93.02 (+0.16)	**81.57** (+2.76)	**70.38** (+1.32)	**85.10** (+2.00)	**76.49** (+1.99)

4.3 Performance across Model Scales and Architectures

Consistent Gains across Model Scales. As shown in Figure 3(a), RDSF consistently improves the six-benchmark mean across Qwen3.5 models from 2B to 27B. The gains persist on the strongest backbone, demonstrating the effectiveness of relation-aware augmentation across model scales.

Magnitude and Quality of Prediction Changes. Table 2 shows that correctness changes become less frequent as model size increases, while the conditional correction-to-disruption ratio R rises. Here, R is the probability of correcting a Base error divided by the probability of introducing an error on a Base-correct example. These trends indicate a more favorable balance between error correction and preservation of correct predictions on larger backbones.

Table 2: **Paired prediction analysis across model scales.** Change rate measures correctness flips; R denotes the conditional correction-to-disruption ratio defined in the text.

Model	Change rate (%)	Ratio R
2B	29.88	1.41
4B	22.52	3.16
9B	14.56	4.67
27B	8.37	**4.74**

Consistent Gains across Backbone Architectures. As shown in Figure 3(b), RDSF consistently improves the four-benchmark mean over the corresponding Base models across Qwen3.5, InternVL3.5, and LLaVA-OV1.5. These results demonstrate the cross-architecture applicability of RDSF. Per-benchmark results are provided in Appendix D.1.

4.4 Qualitative Analysis

Figure 4 compares Base and RDSF on examples containing two to four images. RDSF corrects errors in cross-image matching, comparison, spatial reasoning, and visual analogy. These examples illustrate the benefits of relation-aware fusion across diverse multi-image tasks.

4.5 Ablation and Diagnostic Studies

Relation-Mapping Variants. RDSF outperforms the random-initialization variant on Qwen3.5-9B (Table 3(a)). A matched initialization comparison in Appendix D.3 further supports the benefit of structural pretraining.

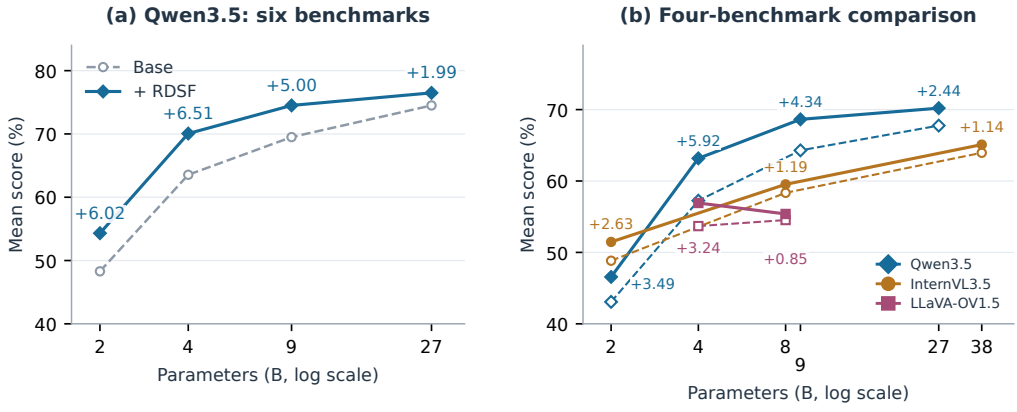

Figure 3: **Performance across model scales and architectures.** Equal-weight mean scores on (a) six benchmarks for Qwen3.5 and (b) MMIU, MIRB, Mantis-Eval, and BLINK across three backbone families. Dashed lines with open markers denote Base; solid lines with filled markers denote RDSF. Labels indicate gains in percentage points.

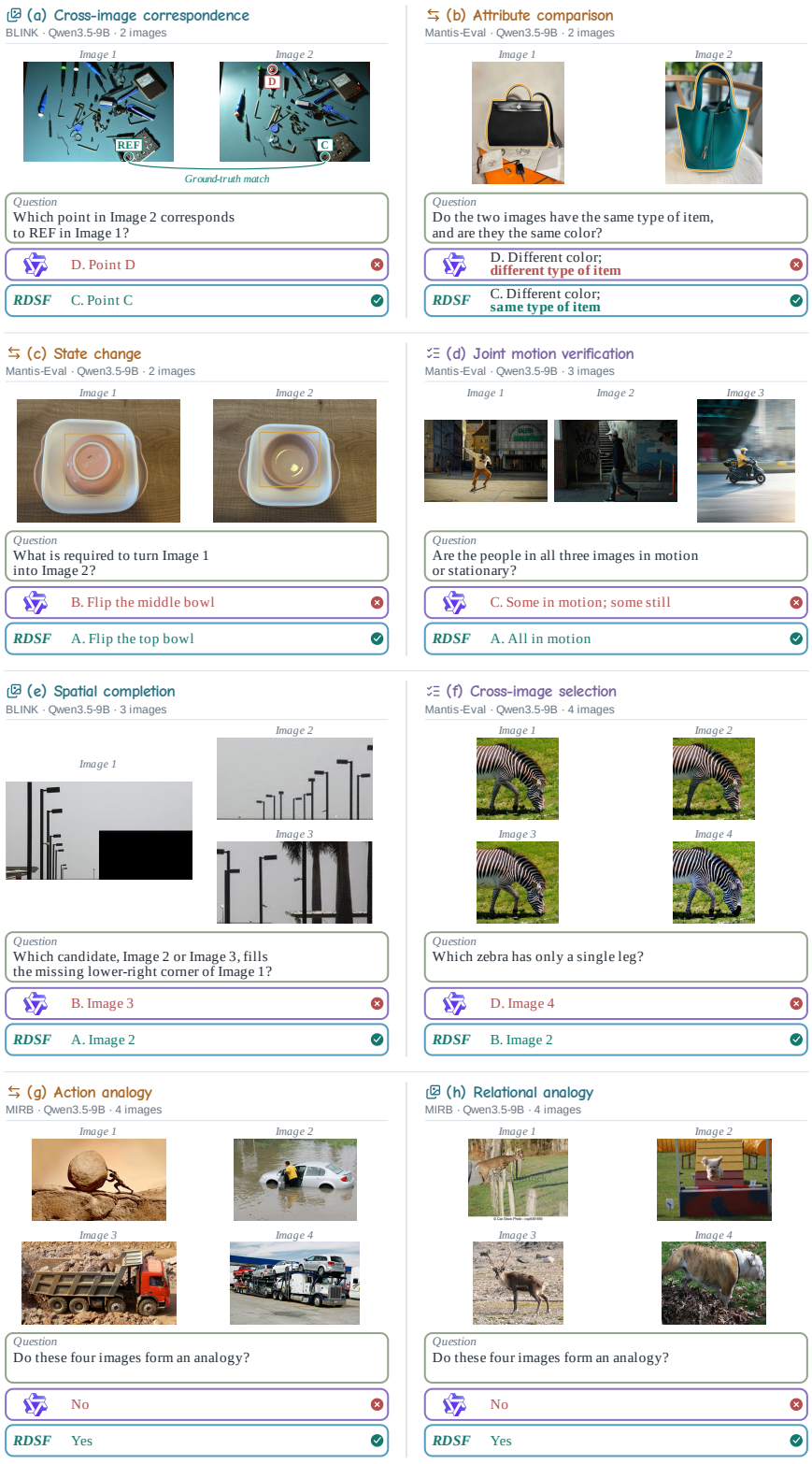

Figure 4: **Qualitative examples.** Selected Base errors corrected by RDSF on Qwen3.5-9B, using two (a–c), three (d–e), and four (f–h) input images.

Effect of Cross-Image Context. We replace context from other images with context from the same image in the relation stream. This reduces performance on all six benchmarks (Table 3(a)), showing the benefit of exchanging context across images before fusion for multi-image understanding.

Effect of the Adaptation Strategy. On Qwen3.5-9B, two-stage LoRA (Hu et al., 2022) improves the six-benchmark mean over Base by 3.66 points, supporting relation-oriented supervision (Table 3(a)). RDSF adds 1.34 points with fewer QA-stage trainable parameters and data passes, providing complementary support for dedicated relational learning and fusion.

Cross-Image Comparison. On 341 pairs where Qwen3.5-9B Base passes all primary local checks, RDSF improves cross-image accuracy from 56.01% to 67.16% (Table 3(b); Appendix A.4). This supports RDSF's ability to compare correctly recognized information across images.

Fusion granularity is examined in Appendix D.2.

Table 3: **Variant comparisons and diagnostic results on Qwen3.5-9B.** (a): unweighted six-benchmark means. Rel. fusion denotes relation-stream fusion; initialization and context refer to that stream. Shading marks RDSF. (b): cross-image accuracy on the same 341 pairs passing Base's primary local checks (%). LoRA uses its final checkpoint; details are in Appendices A.4, C.3 and D.

(a) Model variant	Adaptation	Relation init.	Relation context	Mean (6)
Base	None	—	—	69.50
Two-stage LoRA	LoRA	—	—	73.16
Random initialization	Rel. fusion	Random	Cross-image	73.48
Within-image context	Rel. fusion	Pretrained	Within-image	70.54
RDSF (ours)	Rel. fusion	Pretrained	Cross-image	**74.50**

(b) Cross-image comparison	Base	RDSF
Accuracy (%)	56.01	**67.16**

Limitations. RDSF relies on structured annotations from GQA and Visual Genome for relational pretraining, so its supervision reflects the relation types and visual domains represented in these sources. Generalization to underrepresented relations and domains warrants further investigation. In addition, the relation module exchanges cross-image context without explicitly modeling temporal dynamics. The current multi-image evaluation therefore does not establish its effectiveness for continuous video understanding or long-horizon temporal reasoning. Broader relational supervision and explicit temporal modeling are promising directions for future work.

5 Conclusion

We presented RDSF, a framework for strengthening multi-image understanding through complementary relational learning. Our diagnostics show that successful recognition of task-relevant information within individual images does not ensure reliable cross-image comparison, motivating targeted efforts to improve how models connect visual information across images. RDSF combines structured relational supervision with cross-image context exchange and token-aligned residual fusion, while keeping the pretrained backbone frozen. Experiments demonstrate improvements in average performance across four Qwen3.5 scales and two additional backbone families. Ablations examine the contributions of structured pretraining, cross-image context, and fusion granularity. Together, these results support learning and integrating complementary relational features as an effective approach to enhancing pretrained vision–language models' multi-image capabilities.

AI USE STATEMENT

We used AI tools for language polishing and a vision–language model to assist with data quality checks during dataset construction (Appendix C.1). The authors take full responsibility for the final content of the paper.

REFERENCES

Xiang An, Yin Xie, Kaicheng Yang, Wenkang Zhang, Xiuwei Zhao, Zheng Cheng, Yirui Wang, Songcen Xu, Changrui Chen, Didi Zhu, Chunsheng Wu, Huajie Tan, Chunyuan Li, Jing Yang, Jie Yu, Xiyao Wang, Bin Qin, Yumeng Wang, Zizhen Yan, Ziyong Feng, Ziwei Liu, Bo Li, and Jiankang Deng. LLaVA-OneVision-1.5: Fully open framework for democratized multimodal training. *arXiv preprint arXiv:2509.23661*, 2025. URL https://arxiv.org/abs/2509.23661.

Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. HPatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3852–3861, 2017. doi: 10.1109/cvpr.2017.410.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. BLINK: Multimodal large language models can see but not perceive. In *European Conference on Computer Vision*, pp. 148–166, 2024. doi: 10.1007/978-3-031-73337-6_9.

Roei Herzig, Alon Mendelson, Leonid Karlinsky, Assaf Arbelle, Rogerio Feris, Trevor Darrell, and Amir Globerson. Incorporating structured representations into pretrained vision & language models using scene graphs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 14077–14098, 2023. doi: 10.18653/v1/2023.emnlp-main.870.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Yufeng Huang, Jiji Tang, Zhuo Chen, Rongsheng Zhang, Xinfeng Zhang, Weijie Chen, Zeng Zhao, Zhou Zhao, Tangjie Lv, Zhipeng Hu, and Wen Zhang. Structure-CLIP: Towards scene graph knowledge to enhance multi-modal structured representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 2417–2425, 2024. doi: 10.1609/aaai.v38i3.28017.

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.

Dongfu Jiang, Xuan He, Huaye Zeng, Cong Wei, Max Ku, Qian Liu, and Wenhu Chen. Mantis: Interleaved multi-image instruction tuning. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=skLtdUVaJa.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual Genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017. doi: 10.1007/s11263-016-0981-7.

Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. LLaVA-Interleave: Tackling multi-image, video, and 3D in large multimodal models. In *International Conference on Learning Representations*, 2025a.

Haoxin Li and Boyang Li. Enhancing vision-language compositional understanding with multimodal synthetic data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24849–24861, 2025. doi: 10.1109/cvpr52734.2025.02314.

Xudong Li, Mengdan Zhang, Peixian Chen, Xiawu Zheng, Yan Zhang, Jingyuan Zheng, Yunhang Shen, Ke Li, Chaoyou Fu, Xing Sun, and Rongrong Ji. Zooming from context to cue: Hierarchical preference optimization for multi-image MLLMs. In *Advances in Neural Information Processing Systems*, volume 38, pp. 223–258, 2025b. doi: 10.52202/085713-0008.

Ziyu Liu, Yuhang Zang, Xiaoyi Dong, Pan Zhang, Yuhang Cao, Haodong Duan, Conghui He, Yuanjun Xiong, Dahua Lin, and Jiaqi Wang. MIA-DPO: Multi-image augmented direct preference optimization for large vision-language models. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=f7WBRSuf9l.

Fanqing Meng, Jin Wang, Chuanhao Li, Quanfeng Lu, Hao Tian, Tianshuo Yang, Jiaqi Liao, Xizhou Zhu, Jifeng Dai, Yu Qiao, Ping Luo, Kaipeng Zhang, and Wenqi Shao. MMIU: Multimodal multi-image understanding for evaluating large vision-language models. In *International Conference on Learning Representations*, 2025.

Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. SPair-71k: A large-scale benchmark for semantic correspondence. *arXiv preprint arXiv:1908.10543*, 2019. URL https://arxiv.org/abs/1908.10543.

mPLUG Team. mPLUG-Owl3-7B-241101: Official model card, 2024. URL https://huggingface.co/mPLUG/mPLUG-Owl3-7B-241101. 241101 release; accessed September 12, 2026.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL https://qwen.ai/blog?id=qwen3.5.

Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3D: Large-scale learning and evaluation of real-life 3D category reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10881–10891, 2021. doi: 10.1109/iccv48922.2021.01072.

Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 6418–6428, 2019. doi: 10.18653/v1/P19-1644.

Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5228–5238, 2022. doi: 10.1109/cvpr52688.2022.00517.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, Zhaokai Wang, Zhe Chen, Hongjie Zhang, Ganlin Yang, Haomin Wang, Qi Wei, Jinhui Yin, Wenhao Li, Erfei Cui, Guanzhou Chen, Zichen Ding, Changyao Tian, Zhenyu Wu, Jingjing Xie, Zehao Li, Bowen Yang, Yuchen Duan, Xuehui Wang, Zhi Hou, Haoran Hao, Tianyi Zhang, Songze Li, Xiangyu Zhao, Haodong Duan, Nianchen Deng, Bin Fu, Yinan He, Yi Wang, Conghui He, Botian Shi, Junjun He, Yingtong Xiong, Han Lv, Lijun Wu, Wenqi Shao, Kaipeng Zhang, Huipeng Deng, Biqing Qi, Jiaye Ge, Qipeng Guo, Wenwei Zhang, Songyang Zhang, Maosong Cao, Junyao Lin, Kexian Tang, Jianfei Gao, Haian Huang, Yuzhe Gu, Chengqi Lyu, Huanze Tang, Rui Wang, Haijun Lv, Wanli Ouyang, Limin Wang, Min Dou, Xizhou Zhu, Tong Lu, Dahua Lin, Jifeng Dai, Weijie Su, Bowen Zhou, Kai Chen, Yu Qiao, Wenhai Wang, and Gen Luo. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025. URL https://arxiv.org/abs/2508.18265v2.

Jiabo Ye, Haiyang Xu, Haowei Liu, Anwen Hu, Ming Yan, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. mPLUG-Owl3: Towards long image-sequence understanding in multi-modal large language models. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=pr37sbuhVa.

Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. When and why vision-language models behave like bags-of-words, and what to do about it? In *International*

Conference on Learning Representations, 2023. URL https://openreview.net/forum?id=KRLUvxh8uaX.

Zicheng Zhang, Haoning Wu, Erli Zhang, Guangtao Zhai, and Weisi Lin. Q-Bench+: A benchmark for multi-modal foundation models on low-level vision from single images to pairs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):10404–10418, 2024. doi: 10.1109/TPAMI.2024.3445770.

Bingchen Zhao, Yongshuo Zong, Letian Zhang, and Timothy Hospedales. Benchmarking multi-image understanding in vision and language models: Perception, knowledge, reasoning, and multi-hop reasoning. *arXiv preprint arXiv:2406.12742*, 2024.

A Cross-Image Relational Diagnostics

A.1 Data and Tasks

We construct a controlled diagnostic suite to distinguish local information reading from cross-image relation identification. The evaluation set contains 2,475 examples from SPair-71k (Min et al., 2019), CO3D (Reizenstein et al., 2021), HPatches (Balntas et al., 2017), and GQA/Visual Genome (Hudson & Manning, 2019; Krishna et al., 2017), covering nine task types summarized in Table 4. Ground-truth relations are derived from the original correspondence, geometric, or scene-graph annotations.

Each example presents a query target and two or three candidate targets, each assigned a distinct numerical code. We decompose the task into three steps: reading the candidates' codes, identifying the candidate that satisfies the specified relation, and returning that candidate's code. This construction allows us to examine whether relation-target selection remains difficult when the model can correctly read the local codes required for the final response. Figure 5 illustrates three representative task types.

Table 4: Composition of the diagnostic evaluation set.

Task	Source	Examples
Semantic-part correspondence	SPair-71k	524
Same-instance matching	CO3D	80
Region correspondence	SPair-71k, CO3D	604
Viewpoint proximity	CO3D	52
Camera-motion direction	CO3D	29
Homography correspondence	HPatches	600
Category and attribute matching	GQA/Visual Genome	67
Object identification in directed relations	GQA/Visual Genome	247
Subject identification in directed relations	GQA/Visual Genome	272
Total		2,475

A.2 Experimental Design

We evaluate eight native models from four model families, without additional training or RDSF augmentation. The three stages are queried independently; model responses from one stage are not passed to the next. An example passes the local-reading check only when the model correctly answers every candidate-code query. We then measure relation-identification and final-answer accuracy on each model's local-pass subset.

To examine the contribution of different information types, we keep the images, question, and answer options fixed and vary only the supplementary information:

- **Original:** No supplementary information.
- **LF (local facts):** Provides the complete candidate-to-code mapping without identifying the correct target.
- **CR (correct relation):** Identifies the correct relation target without directly supplying its code.

- **WR (wrong relation):** Identifies another candidate as the relation target.
- **RR (mismatched relation):** Supplies a target from a different source example of the same task type, selected to be incorrect for the current example.

The primary comparison is between CR and LF: the former supplies the relation target, whereas the latter supplies local code information. WR and RR additionally examine responses to incorrect target cues.

A.3 Results and Analysis

Table 5 summarizes the diagnostic results. Across the eight models, local-reading pass rates range from 80.24% to 97.45%, whereas relation-identification accuracy on the local-pass subsets is only 38.67%–55.76%. Together with the local semantic checks in Appendix A.4, these results show that recognizing task-relevant information within individual images does not ensure reliable cross-image comparison.

Providing the correct relation target improves final-answer accuracy by 10.77–28.69 percentage points. The LF intervention changes accuracy by -6.34 to $+1.49$ percentage points. Both WR and RR yield lower accuracy than Original for all eight models.

Table 5: Local reading, relation identification, and information interventions. Accuracies are percentages; gains are percentage points.

Model	Local pass	Relation	Original	LF	CR	CR gain
Qwen3.5-4B	88.53	40.99	40.85	42.13	60.52	+19.67
Qwen3.5-27B	97.45	55.76	63.14	56.80	91.83	+28.69
InternVL3.5-2B	87.39	43.46	47.02	43.55	57.79	+10.77
InternVL3.5-14B	92.16	44.59	47.52	49.01	66.46	+18.94
Gemma 3-4B	80.24	38.67	39.93	37.97	57.55	+17.62
Gemma 3-27B	91.43	43.75	47.95	43.39	66.64	+18.69
LLaVA-OV1.5-4B	88.24	47.39	51.79	47.94	74.36	+22.57
LLaVA-OV1.5-8B	95.56	49.94	48.50	47.57	72.39	+23.89

Motivated by these findings, RDSF learns a dedicated relation stream to connect information across images and complement the backbone's native visual features.

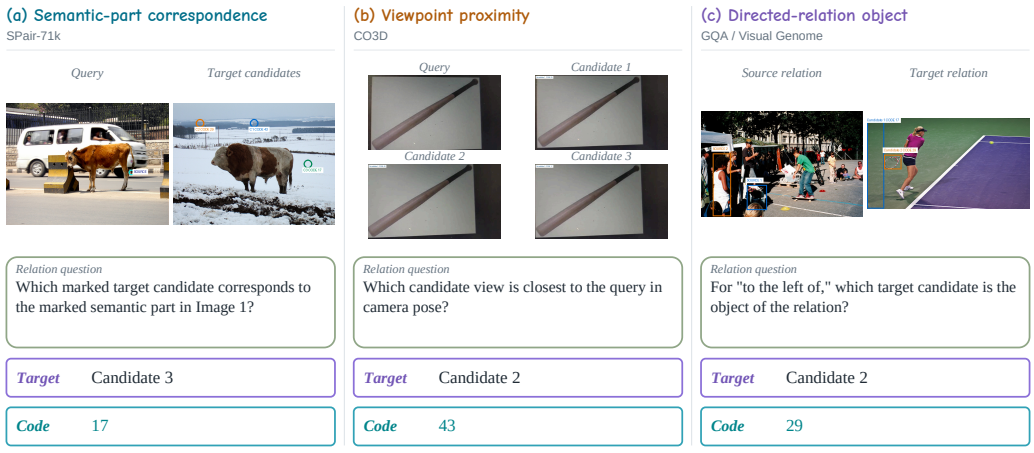

Figure 5: **Diagnostic task examples.** (a) Semantic-part correspondence, (b) viewpoint proximity, and (c) object identification in directed relations.

A.4 TASK-RELEVANT LOCAL SEMANTIC CHECKS

Data and questions. We directly examine whether models can compare information across images when they correctly identify task-relevant attributes and within-image relations. We reuse 50 four-image groups from the confirmation split described in Appendix C.1, comprising 200 images and 400 ordered image pairs. Color binding, horizontal position, and multi-step reasoning each contribute 80 pairs; vertical position, depth, containment, and holding each contribute 40. The cross-image answers are balanced between 200 yes and 200 no labels. These images are separate from the QA-alignment training images and originate from the structural-pretraining source pool.

Each pair has primary local questions, additional consistency questions, and a cross-image comparison question. For example, local questions ask whether the sink is to the left or right of the toilet in each image; the cross-image question asks whether this horizontal relation is the same in both images. Consistency questions check the same facts through inverse or equivalent formulations. Each model answers 1,120 primary local questions, 960 consistency questions, and 400 cross-image questions. Every question is issued independently with the same image pair and image order, without passing earlier answers to subsequent questions. Answers follow the recorded attribute and relation states of the source groups.

Evaluation. We evaluate the eight native models from Appendix A.2 and additionally include Qwen3.5-9B. All nine models use the same next-token option-scoring protocol; unknown responses count as incorrect. A pair passes the primary local checks only if all its primary local answers are correct. The strict subset additionally requires every consistency answer to be correct. Table 6 reports local accuracy per question and cross-image accuracy over all pairs and over each model's own passing subsets.

Table 6: Local semantic checks and cross-image comparison for nine Base models. Accuracies are percentages; passing-subset sizes are out of 400 pairs.

Model	Local accuracy	Local pass N	Cross all	Cross local pass	Strict pass N	Cross strict pass
Qwen3.5-4B	91.43	310	56.00	56.13	246	55.69
Qwen3.5-9B	94.38	341	56.50	56.01	311	55.63
Qwen3.5-27B	94.91	345	64.00	66.96	334	67.37
InternVL3.5-2B	74.91	180	55.25	69.44	117	77.78
InternVL3.5-14B	93.04	328	54.75	57.01	302	57.28
Gemma 3-4B	54.02	88	52.25	54.55	44	59.09
Gemma 3-27B	77.77	203	56.00	56.16	132	61.36
LLaVA-OV1.5-4B	71.25	183	52.00	90.16	155	93.55
LLaVA-OV1.5-8B	71.25	178	54.50	91.57	150	94.00

Results. The Qwen3.5 models and InternVL3.5-14B answer 91.43%–94.91% of primary local questions correctly, yet their cross-image accuracy on the passing subsets is 56.01%–66.96%. For Qwen3.5-9B, 150 of the 341 locally passing pairs still produce a cross-image error; under the strict checks, errors remain on 138 of 311 pairs. Thus, answering the checked attributes and within-image relations correctly does not ensure successful cross-image comparison.

The answer-label composition of the passing subsets varies across models (Table 7). In particular, LLaVA's passing subsets are dominated by yes examples: always predicting yes would exceed its observed conditional accuracy. These label counts accompany the conditional results when interpreting differences across models.

RDSF on the diagnostic. We additionally evaluate the standard Qwen3.5-9B RDSF checkpoint used in the main benchmark comparison, using the same questions and option-scoring protocol as Base. The paired comparison fixes the 341 pairs on which Base passes every primary local question; the strict comparison fixes the 311 pairs passing both primary and consistency checks. Both models are evaluated on these same sets. Table 8 also reports the full set and a supplementary intersection where both models pass the primary checks.

Table 7: Cross-image answer labels in each Base model's passing subsets.

Model	Local pass: yes / no	Strict pass: yes / no
Qwen3.5-4B	163 / 147	133 / 113
Qwen3.5-9B	182 / 159	169 / 142
Qwen3.5-27B	172 / 173	169 / 165
InternVL3.5-2B	136 / 44	101 / 16
InternVL3.5-14B	169 / 159	161 / 141
Gemma 3-4B	65 / 23	38 / 6
Gemma 3-27B	116 / 87	83 / 49
LLaVA-OV1.5-4B	173 / 10	154 / 1
LLaVA-OV1.5-8B	172 / 6	150 / 0

Table 8: Paired cross-image comparison for Qwen3.5-9B. Each row uses the same image pairs for Base and RDSF. Accuracies are percentages; gains are percentage points, computed before rounding.

Comparison set	N	Base	RDSF	Gain
All pairs	400	56.50	65.50	+9.00
Base primary pass	341	56.01	67.16	+11.14
Base strict pass	311	55.63	68.17	+12.54
Both models primary pass	331	56.80	68.58	+11.78

On the fixed primary-pass set, RDSF corrects 65 Base errors and introduces 27, reducing total errors from 150 to 112. The accuracy gain is 11.14 percentage points (95% CI: 7.10–15.36); it is 12.54 points (8.36–17.00) under the strict checks. Intervals use 20,000 paired bootstrap resamples, stratified by task and sampled at the four-image source-group level. Primary local-question accuracy is 94.38% for Base and 95.45% for RDSF. These paired results support improved cross-image comparison on examples where Base already answers the checked local questions correctly.

B Implementation Details of RDSF

B.1 Implementation of Structured Relation Supervision

Projection networks and region pooling. This section specifies the relation mapping and training-time readouts introduced in the main text. In the single-layer Qwen implementation, the relation mapping consists of LayerNorm, a linear projection from the visual feature dimension to d_r, and a two-layer MLP of width d_r, with GELU between the two layers. The predicate encoder is a $4d_r \to d_r \to d_r$ MLP applied to the ordered entity-pair features defined in the main text. We use $d_r = 256$.

Region readout uses the patch indices and normalized weights associated with each annotation:

$$u_a = \sum_{p \in \mathcal{P}(a)} w_{a,p} X_{i,p}, \qquad \sum_{p \in \mathcal{P}(a)} w_{a,p} = 1. \tag{8}$$

Here, $\mathcal{P}(a)$ denotes the valid patches associated with region a. Pooling weights are supplied as part of the supervision rather than predicted by an auxiliary classifier. Region-to-patch associations are established during data preparation.

Auxiliary prediction heads. Linear prediction heads map entity and predicate representations to object categories, attributes, predicate classes, and inverse-relation classes. Cross-image entity and predicate pairs use the comparison features:

$$\chi(h, h') = [h \odot h';\ |h - h'|] \in \mathbb{R}^{2d_r}. \tag{9}$$

These features are passed to pair-type classification and instance-separation heads. The comparison function is invariant to exchanging its inputs, while each predicate representation retains its internal subject–object ordering.

Loss computation and valid supervision. Object categories, predicates, inverse relations, and cross-image pair types use cross-entropy losses averaged over entries with valid targets. Attribute prediction uses masked multi-label binary cross-entropy: annotated attributes are positive targets, while absent attributes are treated as negatives only when the annotation is marked complete. This avoids interpreting missing annotations as evidence of absence.

For entity pairs annotated with symmetric relations, we constrain the predicate distributions before and after exchanging the endpoints:

$$\mathcal{L}_{\mathrm{sym}} = \frac{1}{|\mathcal{E}_{\mathrm{sym}}|\, C_{\mathrm{pred}}} \sum_{(a,b)\in\mathcal{E}_{\mathrm{sym}}} \left\| p_{a\to b} - p_{b\to a} \right\|_1 . \tag{10}$$

Here, $\mathcal{E}_{\mathrm{sym}}$ is the set of pairs with valid symmetric-relation annotations, and C_{pred} is the number of predicate classes. Instance separation uses a binary classification objective over same-category, different-instance pairs and same-instance controls. The constituent losses are weighted to form the structured learning objective in the main text; terms without valid supervision contribute zero. Target construction and loss weights are provided in Appendices C.1 and C.2, respectively.

B.2 Implementation of Visual Interface Alignment

This section specifies the alignment operator $\mathcal{A}_b$ and projector $\mathcal{P}_\theta$ introduced in the main text. Alignment follows the spatial grouping and sequence order of the backbone's native visual tokens, with validity masks propagated alongside the features.

For Qwen's 2×2 spatial-merger interface, each visual position k corresponds to four relation patches. The aligner concatenates them in the native merging order and projects them into the visual-token space:

$$g_{i,k} = \mathrm{Concat}_{p\in\mathcal{G}_{i,k}} R_{i,p} \in \mathbb{R}^{4d_r}, \tag{11}$$

$$a_{i,k} = \mathrm{GELU}(W_{\mathrm{in}}\,\mathrm{LN}(g_{i,k}) + b_{\mathrm{in}}) + t_{\mathrm{rel}}, \tag{12}$$

$$\Delta v_{i,k} = W_{\mathrm{out}} a_{i,k}. \tag{13}$$

Here, $a_{i,k} \in \mathbb{R}^{d_r}$, t_{rel} is a trainable relation embedding shared across positions, and W_{out} maps the hidden features to the backbone's visual-token dimension. A valid merged group contains four valid patches; fully padded groups are excluded from computations over valid positions.

In this implementation, the residual is added to the tokens after native visual merging. Residual scaling, magnitude bounding, and initialization follow the definitions in the main text.

B.3 Auxiliary Objectives for Question–Answer Alignment

We define the native-output and residual regularizers used during question–answer alignment.

Native-output preservation. Under the same teacher-forced text prefix, the preservation term measures the difference between native and augmented distributions at answer positions using $\mathrm{KL}(p_{\mathrm{base}} \| p_{\mathrm{RDSF}})$. Gradients are stopped through the native distribution, which serves as a fixed reference for the fusion module.

Residual regularization. Residual regularization acts on the update actually applied after scaling and magnitude bounding:

$$\mathcal{L}_{\mathrm{rms}} = \mathrm{RMS}_{\Omega}(\widehat{\Delta V}). \tag{14}$$

These terms are optimized as part of the QA objective. Loss coefficients, the residual bound, and optimization settings are provided in Appendix C.2.

C Training and Evaluation Settings

C.1 Training Data Construction

Data sources and relational pretraining. We construct relational pretraining data from the images and scene-graph annotations of GQA (Hudson & Manning, 2019) and Visual Genome (Krishna

Table 9: Cross-image pair supervision for relational pretraining.

Pair type	Supervision categories
Entity pairs	Same category, shared visible attribute, attribute contrast, unrelated, unknown
Relation pairs	Same predicate, predicate contrast, unrelated, unknown

Table 10: Distribution of QA training examples by question type.

Question type	QA examples
Verification	1,901
Selection	1,618
Query and reporting	857
Counting	624
Total	5,000

et al., 2017). For each annotated object, we extract its category, visible attributes, and bounding box to provide entity supervision. For each directed relation, we retain the subject, predicate, and object together with their corresponding image regions to provide relational supervision. These annotations jointly support learning object semantics and inter-object relations. Attribute annotations are treated as incomplete, and unannotated attributes are not automatically regarded as negative labels.

We construct cross-image pairs of objects and subject–predicate–object relations. Object pairs are labeled according to their categories and attributes, while relation pairs are labeled according to their predicates (Table 9). We derive these labels from the source annotations using semantic vocabularies and predefined contrast rules. We use an unknown label when the annotations do not provide enough information.

Multi-image construction and training scale. Each base pair contains two images providing structural supervision and is expanded into inputs containing 2, 4, or 16 images. Additional images serve as distractors without changing the target pair annotations. Each image-count setting is constructed using three sampling seeds, yielding nine structured examples per base pair. The data are partitioned by source group so that derived examples from the same group remain in the same partition.

The resulting relational pretraining pool contains 836,316 multi-image structured examples, with training settings provided in Appendix C.2.

Question–answer alignment data. Question–answer alignment uses 5,000 QA examples covering entity and attribute recognition, spatial relations, object interactions, attribute binding, and compositional reasoning. These include horizontal and vertical position, depth ordering, containment, holding, and two-step questions that identify an object by its spatial relation before querying its attribute. The questions take four forms: verification, selection, querying and reporting, and counting. Table 10 summarizes their distribution.

To construct questions that compare relational states across images, we form groups of four images sharing object categories and a relational condition. Each group contains two images satisfying the condition and two not satisfying it. Pairing these images produces four states: both satisfy the condition, only the first does, only the second does, and neither does. For each state, we construct questions about relation consistency, which images satisfy the condition, the state in each image, and the number of images satisfying the condition. Within each group, each question type uses the same wording across states, so that answer changes reflect changes in visual relations.

We augment the QA examples by reversing the image order for selected questions, with answers determined by the verified relational states and image order. Each QA example is used independently during training.

Quality control and data partitioning. The relational facts used to form these image groups are proposed from the source annotations and checked with a vision–language model for the visibility

Table 11: Two-stage training configurations.

Setting	Relational pretraining	QA alignment
Trainable modules	Relation mapping, predicate encoder, prediction heads	Fusion interface and γ
Optimization steps	5,000	5,000
Peak learning rate	1×10^{-4}	5×10^{-5}
Warmup steps	250	250
Weight decay	0.01	0.01
Gradient norm clipping	1.0	1.0

and clarity of the relevant objects, attributes, and relations. An additional object check without bounding-box guidance identifies referential ambiguity; candidates with unclear relations or non-unique referents are excluded. Programmatic checks further verify answer consistency with the relational states and the corresponding answers after image-order reversal.

We partition source-image groups into training, development, and confirmation sets, keeping all QA examples derived from the same group in the same split. The development and confirmation sets contain 640 and 1,600 QA examples, respectively. During QA alignment, model inputs contain only images and questions, with answers serving as supervision targets. Scene graphs, bounding boxes, and fact-verification records are used for data construction and are not supplied as QA inputs.

C.2 Training Configurations

Optimization. Table 11 summarizes the optimization settings for the two training stages. The pretrained backbone remains frozen throughout, and the relation branch is also frozen during QA alignment. Both stages use AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.95$, and $\epsilon = 10^{-8}$, with linear learning-rate warmup followed by cosine decay. We use BF16 forward computation, FP32 trainable parameters, and random seed 17.

Batching and inputs. Relational pretraining uses backbone-specific batching. Qwen3.5-2B, 9B, and 27B use per-update budgets of at most 32 images and 65,536 visual tokens; the corresponding limits for 4B are 128 images and 262,144 visual tokens. InternVL3.5 uses a global batch of 128 images per update. LLaVA-OV1.5 uses eight structured examples per update, cycling through 2-, 4-, and 16-image inputs. For QA alignment, all backbones use the same fixed sequence of 5,000 examples, with one QA example per update and no gradient accumulation.

For QA training, the image-processor pixel limits are 200,704–401,408 for Qwen3.5, while InternVL3.5 images are resized to 448×448 using bicubic interpolation. The pixel limits for LLaVA-OV1.5-4B and 8B are 3,136–2,560,000 and 3,136–3,240,000, respectively. Benchmark evaluation protocols are provided in Appendix C.4.

Backbone configurations. All models use a relation dimension of 256 and a base fusion hidden dimension of 256. Table 12 lists the visual readout layers, interface dimensions, and adopted relational-pretraining checkpoints; QA evaluation uses the final checkpoint at step 5,000. Layer indices are one-based. InternVL3.5-2B averages two independently projected layer readouts. Additional fusion residual blocks have dimensions $256 \to h \to 256$, with zero-initialized final layers to give an identity mapping at initialization.

Loss coefficients. For the structured objectives in Appendix B, object-category, cross-image entity-pair, directed-predicate, and cross-image relation-pair classification each receive unit weight. Attribute, inverse-relation, symmetry-consistency, and instance-separation losses have weights 0.5, 0.25, 0.1, and 0.25, respectively.

QA alignment uses mean answer-token cross-entropy with unit weight. The native-output and residual regularizers defined in Appendix B.3 have weights 0.1 and 0.01, respectively. The residual RMS bound is $\tau = 0.05$.

Fusion initialization. We initialize the fusion projector's output layer to zero and γ to one.

Table 12: Backbone-specific relation and fusion configurations.

Backbone	Layers	d_v	d_b	h	Pretraining step	α
Qwen3.5-2B	10	1,024	2,048	—	3,000	1.0
Qwen3.5-4B	10	1,024	2,560	—	4,500	1.0
Qwen3.5-9B	27	1,152	4,096	—	3,500	0.75
Qwen3.5-27B	27	1,152	5,120	—	3,500	1.0
InternVL3.5-2B	12, 20	$2 \times 1{,}024$	2,048	512	4,500	1.0
InternVL3.5-8B	24	1,024	4,096	256	4,500	1.0
InternVL3.5-38B	24	3,200	5,120	1,024	5,000	1.0
LLaVA-OV1.5-4B	24	1,024	2,560	256	4,500	1.0
LLaVA-OV1.5-8B	24	1,024	4,096	512	5,000	1.0

d_v is the visual readout width, d_b the native visual-token dimension, and h the hidden width of an additional fusion residual block. An em dash indicates no additional residual block. α is the fixed inference-time scaling factor defined below.

Inference-time residual scaling. QA training uses unit residual scaling. At inference, a fixed coefficient α is applied to the relation residual after RMS bounding:

$$V_{\text{infer}} = V + \alpha\widehat{\Delta V}. \tag{15}$$

The values of α are listed in Table 12. This coefficient is not optimized by gradient descent and is distinct from the learned scale γ applied before bounding.

Reported configurations. All reported RDSF evaluations for a given backbone use the configuration in Table 12, with the QA checkpoint and inference-time residual coefficient fixed across benchmarks.

C.3 Baseline and Ablation Configurations

Table 13 summarizes the baseline and ablation configurations and QA-stage updates for Qwen3.5-9B. The original backbone weights remain frozen in all configurations.

Table 13: Baseline and ablation configurations for Qwen3.5-9B.

Configuration	Relation-mapping initialization	Modules updated during QA training
Native (Base)	No relation mapping	No additional training
RDSF	Structural pretraining	Fusion interface
Two-stage LoRA	No relation mapping	Language-model LoRA adapters
Pretrained initialization	Structural pretraining	Relation mapping and fusion interface
Random initialization	Random initialization	Relation mapping and fusion interface

Initialization variants. We compare the *Random initialization* and *Pretrained initialization* variants on Qwen3.5-9B. Both update the relation mapping and fusion interface during QA training, keeping the original backbone frozen. Their architecture, fusion initialization, QA sequence, and training objectives are matched. Standard RDSF keeps its pretrained relation mapping frozen during QA training.

Matched settings. The two initialization conditions use matched QA training data, example order, and update count. Both are trained for 5,000 steps and evaluated using their final checkpoints, with the same evaluation inputs and scoring rules. They share the inference-time residual coefficient specified in Appendix C.2.

Cross-image context ablation. We replace context aggregated from other images with within-image context in the relation stream and retrain the fusion interface on Qwen3.5-9B. Both conditions keep the backbone and pretrained relation mapping frozen, with matched initialization, QA training

sequences, objectives, and 5,000 updates. The context replacement is applied during both training and inference; the backbone's native multi-image processing remains unchanged.

Two-stage LoRA comparison. We construct 20,000 instruction examples from structural annotations and train LoRA for one epoch, followed by three epochs on the same 5,000 QA examples used for RDSF alignment. QA training continues the adapters learned in the structural stage. We apply LoRA with rank 16, $\alpha_{\mathrm{LoRA}} = 32$, and zero dropout to the attention, linear-attention, and MLP projections of Qwen3.5-9B's language model, keeping the original backbone weights frozen. Both stages minimize answer-token cross-entropy and use AdamW with a peak learning rate of 5×10^{-5}, a global batch of eight, and approximately 5% linear warmup followed by cosine decay. The stages use 2,500 and 1,875 updates, respectively, with the optimizer and schedule restarted at the stage boundary. Optimizer coefficients, image-pixel limits, and random seed follow Appendix C.2. The backbone uses BF16 weights and the adapters use FP32 parameters. We evaluate the final checkpoint after all three QA epochs, using the same benchmark inputs and scoring rules as Base and RDSF.

Table 14 compares QA-stage trainable parameters and data exposure. These quantities exclude RDSF's relational pretraining and do not constitute an end-to-end compute-matched comparison.

Table 14: **QA-stage adaptation settings on Qwen3.5-9B.** Sample presentations count repeated uses of the same QA examples. Parameter counts include only parameters updated during QA adaptation.

Setting	Two-stage LoRA	RDSF
Trainable parameters	43,278,336	1,313,281
Distinct QA examples	5,000	5,000
QA epochs	3	1
QA sample presentations	15,000	5,000
Global batch size	8	1
QA updates	1,875	5,000

Fusion-granularity ablation. We compare token-aligned and image-global fusion on Qwen3.5-2B, 4B, 9B, and 27B. Let $R_{i,p}$ denote the relation message at position p of image i, after cross-image message construction, and let Ω_i denote the valid patch positions of that image. Token-aligned fusion preserves these position-specific messages. Image-global fusion replaces them with their within-image mean:

$$\bar{R}_i = \frac{1}{|\Omega_i|} \sum_{q \in \Omega_i} R_{i,q}, \qquad R_{i,p}^{\mathrm{global}} = \bar{R}_i, \quad p \in \Omega_i. \tag{16}$$

Pooling is performed independently within each image, before Qwen's 2×2 spatial grouping. Padding is excluded from the mean, and padded outputs are zeroed. The resulting features pass through the same normalization, projection, scaling, and residual-fusion operations. Both conditions retain the backbone's native visual tokens and the cross-image context already incorporated into the relation messages. They have the same trainable parameter structure and visual-token sequence length. Each fusion mode is used consistently during training and inference.

Matched training conditions. For each backbone, the two conditions share the same pretrained relation weights, fusion initialization, and fixed sequence of 5,000 QA examples with seed 17. The token-aligned results reuse the archived runs under the ablation objective specified below. Each image-global run is trained independently from the corresponding archived step-0 state. The backbone and relation branch remain frozen; only the fusion interface and its learned scale γ are updated. Both conditions follow the QA optimization settings in Appendix C.2: AdamW with a peak learning rate of 5×10^{-5}, 250 steps of linear warmup followed by cosine decay, and 5,000 updates with one QA example per update and no gradient accumulation.

Shared ablation objective. Both conditions use

$$\mathcal{L}_{\mathrm{abl}} = \mathcal{L}_{\mathrm{answer}} + 0.1\, \mathcal{L}_{\mathrm{keep}} + 0.01\, \mathcal{L}_{\mathrm{rms}}. \tag{17}$$

Here, $\mathcal{L}_{\text{answer}}$ is the mean answer-token cross-entropy. The term $\mathcal{L}_{\text{keep}}$ preserves the native answer distribution through $\mathrm{KL}(p_{\text{base}} \| p_{\text{fused}})$, and $\mathcal{L}_{\text{rms}}$ regularizes the applied relation residual, as defined in Appendix B.3. The residual RMS bound is fixed at $\tau = 0.05$.

Checkpoint selection and evaluation. We evaluate the final step-5,000 checkpoint for each condition. Both conditions use unit residual scaling during training and the same inference-time coefficient within each backbone pair: $\alpha = 0.75$ for 9B and $\alpha = 1.0$ for 2B, 4B, and 27B. Evaluation examples, input formatting, and decoding settings are matched between conditions. Both conditions use the same archived scoring protocol.

C.4 Benchmark Evaluation Protocols

Table 15 summarizes the evaluation populations, sample counts, and prediction modes for the six benchmarks. For MMIU, we exclude 11 records whose answer labels fall outside the declared options, retaining 11,687 valid examples. We use the test set released by TIGER-Lab for NLVR2 and only the publicly labeled validation set for BLINK.

Table 15: Evaluation protocols for the six benchmarks.

Benchmark	Evaluation population	Examples	Prediction mode
MMIU	Valid examples in the public evaluation set	11,687	Option scoring
MIRB	Test	969	254 multiple-choice + 715 open-ended
NLVR2	Test	6,967	Binary option scoring
Mantis-Eval	Test	217	200 multiple-choice + 17 open-ended
BLINK	Validation	1,901	Option scoring
Q-Bench2	Validation	1,000	Option scoring

Input processing. We preserve each benchmark's image order, questions, and answer options where applicable, and use backbone-specific image preprocessing and conversation templates.

Prediction and scoring. We use option scoring for multiple-choice questions and greedy decoding for open-ended questions. Base and RDSF are evaluated with the same answer normalization and benchmark-specific scoring rules. We report accuracy for all benchmarks except MIRB, whose score is computed by averaging task accuracies within each of its four dimensions and then averaging across dimensions.

D Additional Results

D.1 Benchmark-Level Gains across Backbone Architectures

Figure 6 provides the benchmark-level breakdown of the cross-architecture comparison in Figure 3(b). It reports RDSF gains over the corresponding Base models on MMIU, MIRB, Mantis-Eval, and BLINK, using the same configurations as the main-text comparison.

D.2 Fusion Granularity across Model Scales

Figure 7 summarizes the six-benchmark means. Token-aligned fusion achieves higher six-benchmark means across all four Qwen3.5 scales, improving by 1.07, 1.75, 0.20, and 2.30 percentage points for 2B, 4B, 9B, and 27B, respectively. The largest mean improvement occurs at 27B, where all six benchmarks improve, led by MIRB (+4.08), Q-Bench2 (+3.30), and Mantis-Eval (+3.23). These results show the benefit of preserving spatially varying relation information during fusion and support the token-aligned design of RDSF.

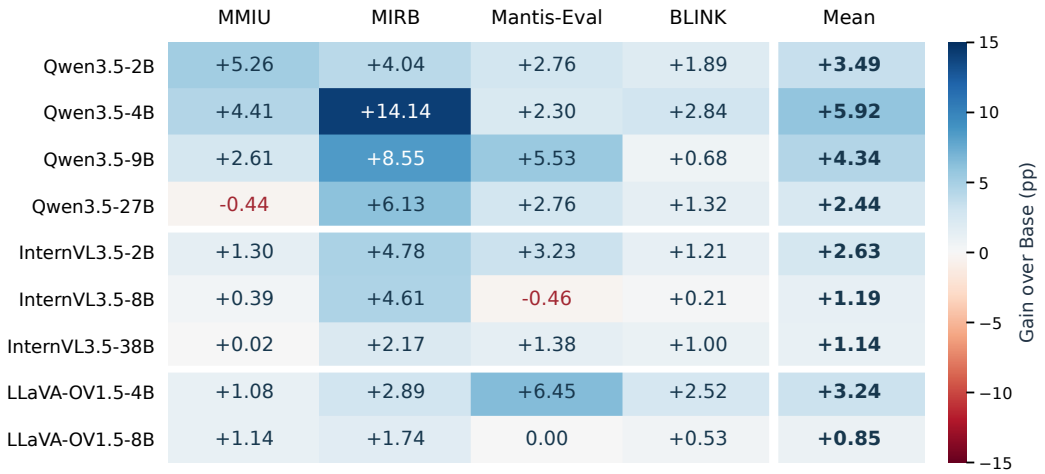

Figure 6: **Benchmark-level gains across backbone architectures.** Values are RDSF minus Base in percentage points; Mean is the unweighted average over the four benchmarks. The color scale is centered at zero. Scores use shared answer-format normalization for Base and RDSF.

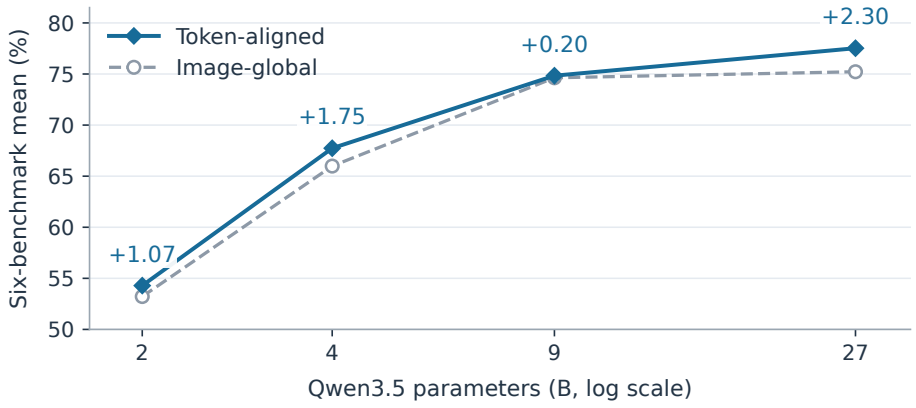

Figure 7: **Fusion granularity across Qwen3.5 model scales.** Both variants use the matched ablation settings in Appendix C.3. Scores are unweighted six-benchmark means; labels show Token minus Global in percentage points, computed before rounding.

D.3 Relation Initialization and Cross-Image Context

Table 16 reports matched initialization and context comparisons on Qwen3.5-9B, complementing Table 3(a). In the initialization comparison, structural pretraining improves the six-benchmark mean by 1.04 percentage points under the matched QA settings in Appendix C.3.

Table 16: **Benchmark-level initialization and context ablations on Qwen3.5-9B.** Scores are percentages; Mean (6) is the unweighted six-benchmark average. Bold indicates the higher score within each comparison. Shading denotes structured initialization in (a) and cross-image context in (b).

(a) Relation-mapping initialization.

Backbone	Initialization	MMIU	MIRB	NLVR2	Mantis-Eval	BLINK	Q-Bench2	Mean (6)
Qwen3.5-9B	Random initialization	61.31	63.51	**90.46**	79.26	65.65	80.70	73.48
	Pretrained initialization	**61.50**	**64.34**	88.80	**82.95**	**67.33**	**82.20**	**74.52**

(b) Cross-image context.

Backbone	Context	MMIU	MIRB	NLVR2	Mantis-Eval	BLINK	Q-Bench2	Mean (6)
Qwen3.5-9B	Cross-image (RDSF)	**61.58**	**65.28**	**90.80**	**80.65**	**67.02**	**81.70**	**74.50**
	Within-image	58.00	62.98	89.31	74.19	62.97	75.80	70.54

D.4 Adaptation Strategy

Table 17 provides the benchmark-level results summarized in Table 3(a). LoRA uses the final checkpoint of the two-stage training schedule described in Appendix C.3. Base and RDSF use the Qwen3.5-9B configurations from Table 1.

Table 17: **Adaptation strategies on Qwen3.5-9B.** Scores are percentages; Mean (6) is the unweighted six-benchmark mean. Bold marks the highest score in each column; shading denotes RDSF.

Method	MMIU	MIRB	NLVR2	Mantis-Eval	BLINK	Q-Bench2	Mean (6)
Base	58.97	56.73	81.94	75.12	66.33	77.90	69.50
Two-stage LoRA	**61.97**	63.62	90.46	76.04	66.18	80.70	73.16
RDSF	61.58	**65.28**	**90.80**	**80.65**	**67.02**	**81.70**	**74.50**