
Latent-Lens: Visual Perception in Small Language Models Through Latent Communication

Mo Shahdloo

Microsoft Applied Sciences Group (ASG)

Faegheh Sardari

Microsoft Applied Sciences Group (ASG)

Mathew Salvaris

Microsoft Applied Sciences Group (ASG)

Abstract

Bridging independent vision-language models (VLMs) and small language models (SLMs) in a modular sender–receiver system typically forces them to exchange information as discrete text, which creates a lossy bottleneck for spatial and fine-grained details. This constraint is especially acute for SLMs, whose limited capacity makes them more reliant on the quality of the input signal. We propose **Latent-Lens**, a lightweight codec that bridges a frozen VLM and a frozen text-only SLM directly in latent space. The codec projects three depth-band summaries from the VLM into the SLM’s embedding space and uses rank-4 LoRA on the SLM’s query and value projections, training 1.6M parameters for the default pair (0.2% of the combined model size). Across three runs, Latent-Lens obtains 80.97 ± 7.93 on CLEVR and 65.39 ± 5.43 on GQA, compared with 58.27 ± 5.69 and 61.75 ± 1.03 for text communication with the same LoRA budget. A token-count-matched final-layer control supports the value of cross-depth extraction on CLEVR, while GQA shows a smaller, seed-sensitive gain. On image captioning (Flickr8k), Latent-Lens improves over the same baseline by **+18.5** BLEU-4 (**+18.9** when both agents are frozen). All four combinations of two VLM and two SLM families support the interface, establishing compatibility within the evaluated small-model range.

1 Introduction

Small language models (SLMs) are increasingly deployed on-device and at the edge, where compute and memory budgets can preclude accommodating a monolithic vision-language model, leaving these systems without visual perception. Tightly coupled vision-language models (VLMs) such as LLaVA [Liu et al., 2023] and Qwen-VL [Bai et al., 2023] fuse visual and linguistic representations within a single end-to-end trained system. While effective, this tight coupling is inflexible. Upgrading the vision encoder or the language model requires retraining the entire system, and scaling such VLMs down to fit on-device budgets often degrades visual capabilities disproportionately. Multi-agent approaches offer modularity by pairing a small VLM with a small SLM, but have so far relied on text-based communication [Du et al., 2024]. However, text constitutes a lossy channel: spatial layouts and fine-grained visual attributes are difficult to verbalise faithfully, and an SLM cannot recover details that the sender has discarded.

Recent latent-communication methods primarily exchange language-model hidden states or KV caches [Zou et al., 2025, Zheng et al., 2025, Dery et al., 2026], leaving cross-modal protocols comparatively underexplored.

We propose **Latent-Lens**, a lightweight codec that bridges the representation gap between a frozen small VLM and a frozen text-only SLM, allowing them to communicate directly in latent space and

thus bypass the lossy text bottleneck. By extracting multiband latent prefixes, the method preserves the depth at which visual features are encoded across the VLM’s layers.

The codec consists of two functionally distinct components, jointly trained end-to-end. The first is a lightweight MLP domain encoder that extracts and projects hidden states from three depth bands (early, mid, late) of the frozen VLM. The second is an attention decoder, instantiated as LoRA adapters on the frozen SLM’s query and value projections, that teaches the SLM to attend to these projected prefix tokens. We evaluate two SLMs, Qwen2.5-0.5B (896-dim [Alibaba Cloud, 2024]) and TinyLlama-1.1B (2048-dim [Zhang et al., 2024]), with two VLMs, SmolVLM-256M (576-dim [HuggingFace, 2024]) and InternVL2-1B (896-dim [Chen et al., 2024]). Only 1.6–6.6M parameters are trained per pair, representing under 1% of the combined frozen parameters.

We evaluate on fixed splits from CLEVR [Johnson et al., 2017] and GQA [Hudson and Manning, 2019], and on an open-ended captioning task using Flickr8k [Hodosh et al., 2013], comparing against VLM-only, SLM-only, text-communication, and LLaMA-Adapter [Zhang et al., 2023] baselines. Three-run evaluation and a token-matched final-layer control support the CLEVR result; the GQA advantage is smaller and seed-sensitive. Results across four VLM–SLM pairings show compatibility within the evaluated small-model range.

2 Related Work

Text-based inter-model communication. Multi-agent systems built from independently trained components most commonly exchange information through natural language. Du et al. [2024] show that multi-round text debate among LLM instances improves factuality and reasoning. Liang et al. [2024] extend this by encouraging divergent thinking across agents to enhance answer diversity, and ChatDev [Qian et al., 2024] applies text-based conversation to coordinate specialised models for software development. In the visual domain, BeMyEyes [Huang et al., 2025] pairs a small VLM with a large proprietary LLM, where the VLM describes the image in text and the LLM reasons over the description. While modular, all these approaches are limited by the expressiveness of the text channel, which discards spatial layout and fine-grained visual attributes.

Latent representation transfer between models. An alternative to text is to communicate directly through hidden states, motivated by the observation that natural language is structurally misaligned with LLMs’ high-dimensional representations and causes irreversible information compression [Zhou et al., 2025]. In line with this, Yue et al. [2025] show that 28-73% of text-based inter-agent communication is redundant. The idea originated in multi-agent reinforcement learning. DIAL [Foerster et al., 2016] introduced differentiable inter-agent messaging, instantiated by CommNet [Sukhbaatar et al., 2016] as a mean-pooled hidden-state channel. Subsequent work introduced selectivity over the channel via attention-based targeting in TarMAC [Das et al., 2019] and learnable gating in IC3Net [Singh et al., 2019], and structured topologies via relational graphs [Jiang et al., 2019].

More recently, the principle has been applied to transfer representations between frozen language models. LatentMAS [Zou et al., 2025] shares last-layer hidden embeddings through a latent working memory between *identical* LLMs, while ThoughtComm [Zheng et al., 2025] studies identifiable latent exchange. Interlat [Du et al., 2025], C2C [Zhang et al., 2025], and Dery et al. [2026] target heterogeneous LLMs through compression or KV-cache alignment. SDE [Tang et al., 2025] augments text with hidden-state deltas, while Vision Wormhole [Liu et al., 2026] communicates through a receiver VLM’s visual-token interface; our setting instead passes image-conditioned representations from multiple VLM depth bands to a text-only SLM. Within a *single* model, LLaMA-Adapter [Zhang et al., 2023] inserts gated visual residuals at several transformer depths.

Contributions.

- We introduce **Latent-Lens**, a lightweight latent-communication codec that bridges a frozen VLM and a separately pretrained text-only SLM.
- Latent-Lens represents VLM depth explicitly through compact band summaries, in contrast to single-layer hidden-state exchange and layer-wise cache alignment.
- Three-run VQA comparisons, a token-matched final-layer control, and results across four VLM–SLM pairings characterize the method within the evaluated small-model setting.

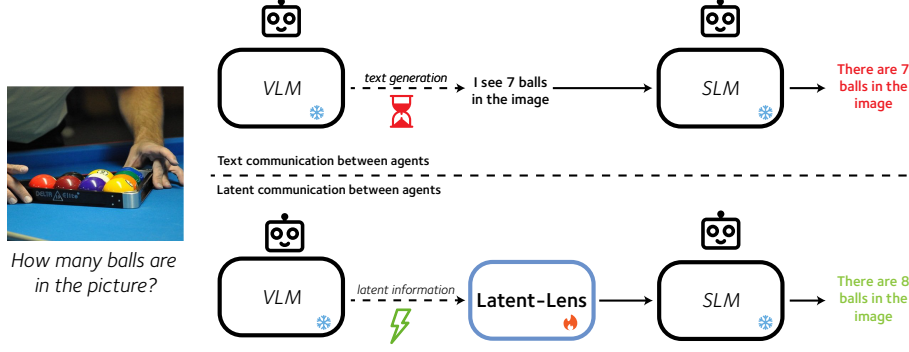


Figure 1: **Text and latent communication.** Text communication decodes the VLM state into discrete tokens before the SLM receives it. Latent-Lens instead projects image-conditioned VLM states directly into the SLM input. Black borders denote frozen components, while the blue border denotes the trained codec.

3 Method

3.1 Problem Setup

Given an image and a question, we want a frozen SLM to generate a free-form answer that leverages visual information from a frozen VLM, without any text exchange between the two models. Concretely, the VLM $\mathcal{V}$ is one of SmolVLM-256M ($L_V = 30$, $d_V = 576$) or InternVL2-1B ($L_V = 24$, $d_V = 896$), and the SLM $\mathcal{S}$ is one of Qwen2.5-0.5B ($L_S = 24$, $d_S = 896$) or TinyLlama-1.1B ($L_S = 22$, $d_S = 2048$). The two models have different tokenizers, vocabularies, and embedding dimensions, so the latent channel must explicitly bridge their representation spaces.

3.2 Multiband Signal Decomposition

A representation selected from one layer exposes only one point in the VLM’s depth hierarchy. To preserve cross-depth structure, we decompose the hidden states into multiple bands, each spanning a contiguous group of transformer layers.

The VLM processes the image and question, producing per-layer hidden states $\{h_\ell\}_{\ell=0}^{L_V-1}$ of the VLM’s transformer stack, where $h_\ell \in \mathbb{R}^{T \times d_V}$ is the activation at layer ℓ and T is the input sequence length. We partition the L_V layers into $B = 3$ bands of equal size and compute the band representation as the mean of last-token hidden states within each band:

$$\mathbf{v}_b = \frac{1}{|\mathcal{L}_b|} \sum_{\ell \in \mathcal{L}_b} h_\ell[:, -1, :] \in \mathbb{R}^{d_V}, \quad b \in \{1, 2, 3\} \quad (1)$$

where $\mathcal{L}_b$ is the set of layer indices in band b . Each band covers a contiguous third of the VLM’s transformer stack: Band 1 (early), Band 2 (mid), and Band 3 (late). For SmolVLM-256M (30 layers) this yields L0-9 / L10-19 / L20-29, and for InternVL2-1B (24 layers), L0-7 / L8-15 / L16-23. Section 4.5 reports descriptive analyses of their linear decodability and representational similarity.

3.3 Domain Transcoding

The three band representations are projected from the VLM’s representation space into the SLM’s embedding space using a shared 2-layer MLP that serves as a lightweight domain transcoder.

$$\tilde{\mathbf{v}}_b = W_2 \text{GELU}(W_1 \mathbf{v}_b + b_1) + b_2, \quad W_1 \in \mathbb{R}^{d_S \times d_V}, \quad W_2 \in \mathbb{R}^{d_S \times d_S} \quad (2)$$

This yields three projected embeddings $\tilde{\mathbf{v}}_1, \tilde{\mathbf{v}}_2, \tilde{\mathbf{v}}_3 \in \mathbb{R}^{d_S}$, one per band. The transcoder has 1.3M trainable parameters.

3.4 Prefix Delivery and SLM Adaptation

The transcoded band embeddings are prepended as soft prefix tokens to the SLM’s input sequence:

$$\mathbf{E}_{\text{input}} = [\tilde{\mathbf{v}}_1; \tilde{\mathbf{v}}_2; \tilde{\mathbf{v}}_3; \mathbf{e}_1, \dots, \mathbf{e}_T] \in \mathbb{R}^{(3+T) \times d_s} \quad (3)$$

where $\mathbf{e}_1, \dots, \mathbf{e}_T$ are the SLM’s text token embeddings. The prefix tokens receive attention mask 1 so the SLM attends to them at every decoding step, but their positions are excluded from the SLM’s next-token prediction (cross-entropy) loss (label = -100).

To adapt the SLM to attend to this novel visual signal, we add LoRA (rank 4, ablated in Section 4.3) to the query and value projections of every attention layer, contributing 0.3M trainable parameters. For the SmolVLM-256M + Qwen2.5-0.5B pair, this gives a total trainable budget of 1.6M parameters (1.3M transcoder + 0.3M LoRA), or 0.2% of the 752M combined frozen parameters. Other VLM-SLM pairs scale similarly with the VLM/SLM hidden dimensions (Section B).

4 Experiments

4.1 Setup

Datasets. We use two free-form VQA benchmarks and one open-ended captioning benchmark. **CLEVR** [Johnson et al., 2017]¹ contains rendered scenes with questions about counting, spatial relations, and attributes; we use 5,000 examples with a fixed 4,000/1,000 split. **GQA** [Hudson and Manning, 2019]² contains compositional questions over natural images; we use a fixed 9,600/2,400 split. These are fixed splits of the selected source pools rather than the canonical benchmark protocols. **Flickr8k** [Hodosh et al., 2013]³ is an open-ended image-captioning task with a 4,000/1,000 split.

Evaluation. For VQA, predictions and references are lowercased, stripped of articles and punctuation, normalized for whitespace, and mapped from number words to digits. We then accept exact match or bidirectional word-set containment. This normalized containment heuristic was validated against 50 hand-labelled examples (100% agreement), but it is not a complete semantic metric. Multiple-choice formatting with random distractors inflated SLM-only GQA accuracy in preliminary experiments, motivating free-form generation. Flickr8k is evaluated with BLEU-4 against the reference captions.

Training. All trained conditions use AdamW with learning rate 2×10^{-4} , cosine annealing, weight decay 10^{-4} , gradient clipping at 1.0, effective batch size 16 (4×4 gradient accumulation), and five epochs. Unless otherwise noted, mean and sample standard deviation are reported over three complete runs.

Baselines. (1) **VLM-only**: SmolVLM-256M [HuggingFace, 2024] generates directly from the image and question. (2) **SLM-only (frozen)**: Qwen2.5-0.5B [Alibaba Cloud, 2024] receives text only. (3) **Text communication**: the same frozen VLM generates a response that is inserted into the SLM prompt, with either a frozen SLM or the same rank-4 Q/V LoRA used by Latent-Lens. This is a single handoff and is not an implementation of BeMyEyes [Huang et al., 2025], whose trained perceiver and multi-turn protocol instead provide conceptual motivation. (4) **SLM-only + LoRA**: the text-only SLM with rank-4 Q/V LoRA [Hu et al., 2022]. (5) **LLaMA-Adapter** [Zhang et al., 2023]: a parameter-matched visual interface that retains the shared VLM→SLM projection and injects gated visual residuals at the upper 12 SLM layers. (6) **Latent-Lens**: our input-prefix codec with the same rank-4 Q/V LoRA. The original text+LoRA and Latent-Lens conditions use identical frozen backbones, examples, optimizer, epochs, decoding budget, and SLM adaptation; Latent-Lens additionally trains its projection MLP.

Text prompts. On VQA, the original sender prompt is “*Question: [question] Answer with a single word or short phrase:*”; the receiver is told “*A vision model looked at the image and responded:*”

¹https://huggingface.co/datasets/MMInstruction/Clevr_CoGenT_ValA

²<https://huggingface.co/datasets/lmms-lab/GQA>

³<https://huggingface.co/datasets/jxie/flickr8k>

Table 1: Main results on CLEVR and GQA (accuracy) and Flickr8k (BLEU-4). VQA entries with $\pm$ report mean $\pm$ sample standard deviation over three runs; point estimates are single-run conditions. $\dagger$ changes only the VLM sender prompt.

Condition	Params	CLEVR (%)	GQA (%)	Flickr8k (BLEU-4)
VLM only	0	58.6	41.8	13.9
SLM only (frozen)	0	1.5	18.0	11.8
Text communication (frozen)	0	11.2	35.2	0.1
SLM only + LoRA	0.3M	42.2	45.6	11.7
Text communication + LoRA	0.3M	58.27 ± 5.69	61.75 ± 1.03	0.5
Evidence prompt + LoRA †	0.3M	69.5	58.23	0.3
LLaMA-Adapter	1.6M	73.57 ± 3.46	61.93 ± 6.14	17.3
Latent-Lens (ours, $B=3$)	1.6M	80.97 ± 7.93	65.39 ± 5.43	19.0

[response]. Based on this, answer the following question with a single word or short phrase.” To test whether the result depends on this limited response, we change only the sender instruction to “Briefly describe the visual evidence relevant to answering the question in one or two concise sentences. Do not answer the question directly.” All other settings remain fixed.

4.2 Main Results

Table 1 reports the main comparison. On CLEVR, Latent-Lens improves over text+LoRA by 22.70 ± 5.41 points and over LLaMA-Adapter by 7.40 ± 7.84 points under paired runs. On GQA, the corresponding gains are 3.64 ± 5.98 and 3.46 ± 4.43 points, and one of the three runs is below each baseline; we therefore treat the GQA improvement as seed-sensitive.

Frozen baselines. The frozen SLM reaches 1.5% / 18.0% / 11.8 BLEU-4 on CLEVR / GQA / Flickr8k, the VLM reaches 58.6% / 41.8% / 13.9, and the frozen text handoff yields 11.2% / 35.2% / 0.1. These VQA text-handoff scores remain below direct VLM generation, showing that the receiver cannot recover visual evidence omitted by the sender.

Stronger text prompt. Changing only the sender instruction raises text+LoRA to 69.5% on CLEVR but reaches 58.23% on GQA. Despite the request for one or two sentences, SmolVLM emits 1.01 words on CLEVR and 1.34 words on GQA on average; the intervention therefore improves the CLEVR handoff without producing dense VQA descriptions. The original Latent-Lens run obtains 89.6% and 71.6% under the otherwise identical setting.

Alternative VLM-LM coupling. LLaMA-Adapter introduces visual context at the upper transformer layers, whereas Latent-Lens injects the prefix at the input so every SLM layer can condition on it. The three-run comparison in Table 1 shows a clear CLEVR difference and overlapping variability on GQA.

Per-benchmark analysis. CLEVR isolates spatial relations, counting, and attribute recognition, and shows the clearest separation between latent and text communication. GQA contains natural images and stronger language priors; its smaller, variable margin places a narrower bound on the evidence for transfer. Flickr8k is an open-ended captioning task that allows the VLM to emit substantially richer text than the terse VQA answers, which we hypothesised would narrow the gap between Latent-Lens and VLM-only generation. The VLM produces 40.6-word descriptions on average against 1.0 and 1.1 words on CLEVR and GQA, and the +5.1 BLEU-4 margin of Latent-Lens over VLM-only is the smallest in our experiments, consistent with this hypothesis.

4.3 Ablations

Number of bands. We ablate $B \in \{1, 2, 3, 5, 7\}$ with $r = 4$ fixed (Figure 2 in the appendix). The $B = 1$ condition averages the last-token states over *all* VLM layers into one token; it is not a single-layer baseline. It scores highest on CLEVR (91.8%) and Flickr8k (19.5 BLEU-4), while

Table 2: Token-count-matched cross-depth control. Both conditions use three prefix positions and the same training protocol. Values are mean $\pm$ sample standard deviation over three runs.

Representation	CLEVR (%)	GQA (%)
Final layer repeated $\times 3$	71.47 $\pm$ 0.92	62.69 $\pm$ 1.62
Three depth bands	80.97 $\pm$ 7.93	65.39 $\pm$ 5.43
Paired difference	+9.50 $\pm$ 8.81	+2.70 $\pm$ 5.56

Table 3: VLM scale comparison across the two VLMs, with the default Qwen2.5-0.5B SLM.

Condition	CLEVR (%)	GQA (%)	Flickr8k (BLEU-4)
SmolVLM-256M only	58.6	41.8	13.9
+ Latent-Lens (ours)	89.6	71.6	19.0
InternVL2-1B only	82.2	58.4	13.9
+ Latent-Lens (ours)	88.3	68.3	14.3

$B = 5$ and $B = 7$ are highest on GQA (76.2% and 76.5%). We therefore treat $B = 3$ as a compact fixed-bandwidth default rather than an optimum, and the preferred granularity as task-dependent.

Cross-depth control. Repeating the true final-layer state into three prefix positions changes only the representation source while matching communication length. The three bands outperform this control on all CLEVR runs. On GQA, the average gain is smaller and is not positive in every run. Cross-depth extraction therefore has clear task-dependent value without establishing that three bands are universally optimal.

LoRA rank. To assess whether additional LoRA capacity beyond the default helps, we ablate $r \in \{4, 8, 16, 32, 64\}$ with $B = 3$ fixed (Figure 3 in the appendix). Rank $r = 4$ yields the highest scores on all three benchmarks (89.6 / 71.6 / 19.0), against 88.0 / 57.1 / 14.9 at $r = 16$ and 86.5 / 70.0 / 14.4 at $r = 64$. Intermediate ranks ($r = 8, 32$) are non-monotonic, with $r = 8$ dropping to 73.0% on CLEVR. Increasing LoRA capacity beyond rank 4 does not improve any of the three metrics universally.

4.4 VLM Scale: SmolVLM-256M vs. InternVL2-1B

To test whether richer VLM hidden states improve downstream SLM performance, we replace SmolVLM-256M (576-dim, 30 layers) with InternVL2-1B [Chen et al., 2024] (896-dim, 24 layers, comprising an InternViT-300M encoder and a Qwen2-0.5B LLM). The SLM remains Qwen2.5-0.5B, and only the projection MLP is adjusted for the VLM’s hidden dimension.

Latent-Lens improves over direct VLM generation with both evaluated senders. InternVL2-1B is substantially stronger in isolation, while the two Latent-Lens configurations reach similar VQA scores. This establishes compatibility with both evaluated VLMs; it does not establish scale invariance.

4.5 Representation Analysis

We measure two descriptive properties of the multiband latent channel: linear answer decodability from each band and representational similarity between bands.

Linear-probe decodability. We fit a logistic-regression probe on each frozen band representation to predict the answer classes observed in the training split. The resulting accuracy measures linear decodability; it is not interpreted as a mutual-information bound.

Higher bands are more linearly predictive, consistent with the depth-abstraction hierarchy [Tenney et al., 2019]. The late-band mean slightly exceeds the final layer on both tasks, while the full codec scores higher than every linear probe.

Table 4: Linear probe accuracy per band on CLEVR and GQA.

Representation	CLEVR (%)	GQA (%)
Band 1 (early, L0-9)	59.6	43.1
Band 2 (mid, L10-19)	67.9	53.2
Band 3 (late, L20-29)	70.1	57.8
Last layer only	68.6	56.5

Table 5: Latent-Lens VQA accuracy for the four evaluated VLM–SLM pairs. These are single-run compatibility checks.

VLM	SLM	CLEVR (%)	GQA (%)
SmolVLM-256M	Qwen2.5-0.5B	89.6	71.6
InternVL2-1B	Qwen2.5-0.5B	88.3	68.3
SmolVLM-256M	TinyLlama-1.1B	78.2	66.3
InternVL2-1B	TinyLlama-1.1B	87.4	72.4

Inter-band similarity. Linear CKA [Kornblith et al., 2019] is lower between Bands 1 and 3 (0.174 on GQA, 0.441 on CLEVR) than between Bands 2 and 3 (0.798 / 0.885). This indicates greater representational dissimilarity across distant depths, but does not by itself establish statistically independent or complementary information.

Effective dimensionality. We describe the spread of the centered representations using effective rank [Roy and Vetterli, 2007], $r_{\text{eff}} = \exp(-\sum_i \hat{\sigma}_i \log \hat{\sigma}_i)$, where $\hat{\sigma}_i = \sigma_i / \sum_j \sigma_j$. Per band, r_{eff} ranges from 187 to 230 on CLEVR and 267 to 364 on GQA out of 576 dimensions; concatenating the bands gives 304 and 473. These values characterize effective dimensionality and are not a channel-capacity estimate.

4.6 Compatibility Across Evaluated Model Pairs

We combine two VLMs and two text-only SLMs with different architectures, tokenizers, vocabularies, and hidden dimensions. Each pair uses its own projection MLP and the same three-token interface.

All four pairs support the interface on both VQA tasks. The spread across pairings also shows that performance depends on the selected backbones; the experiment establishes compatibility within these four small-model configurations. With SmolVLM on Flickr8k, Qwen and TinyLlama obtain 19.0 and 17.5 BLEU-4, respectively, while TinyLlama+LoRA obtains 27.1.

4.7 Latency and Efficiency

Replacing autoregressive VLM text generation with one VLM forward pass reduces end-to-end inference cost under the measured 20-token answer budget. Text communication first generates up to 50 sender tokens, whereas Latent-Lens directly extracts hidden states. On the reported single-24GB-GPU measurement, Latent-Lens takes 781 ± 11 ms and text communication takes $2,137 \pm 9$ ms, a measured $2.7\times$ difference. If image features can be precomputed, online generation uses the SLM and a 3.4KB fp16 prefix per image. Table 9 (appendix) reports an additional cross-attention experiment on ScienceQA [Lu et al., 2022].

5 Limitations

Latent-Lens is a unidirectional sender–receiver interface, not a demonstration of coordination in larger teams or of multi-turn latent dialogue. It requires white-box access to VLM hidden states and a separately trained projection for each VLM–SLM pair, which limits use with closed models and introduces pair-specific training cost. The evaluation covers two small VLMs, two small SLMs, and fixed, noncanonical splits of three datasets; behaviour at larger scales and on other visual tasks

Table 6: Inference latency and GPU memory on a single 24GB GPU (fp16, 20-token generation, averaged over 10 runs).

Condition	Total Params	GPU Memory	Latency
VLM only	256M	1,008 MB	277 ± 1 ms
SLM only (+ LoRA)	495M	1,013 MB	620 ± 36 ms
Text communication	752M	2,113 MB	$2,137 \pm 9$ ms
Latent-Lens	752M	2,477 MB	781 ± 11 ms

remains unknown. The main VQA comparison uses three runs, while the backbone-pairing, most ablation, latency, and captioning results are single-run. The GQA margin is seed-sensitive. VQA scoring uses a normalized containment heuristic, and BLEU-4 captures only part of caption quality. The equal-depth partition and $B=3$ default are design choices whose preferred values vary by task.

6 Broader Impact

Latent communication can make modular visual systems more efficient and can give small text models access to visual evidence without retraining either backbone. The same opacity that avoids a text bottleneck also reduces auditability: a latent prefix is difficult for users or safety filters to inspect, can transmit sensitive image information that is absent from a textual summary, and inherits biases, hallucinations, and failure modes from both backbones and the training data. The current results do not support safety-critical deployment. Practical systems should retain access controls and data-minimization policies for cached prefixes, evaluate demographic and domain-specific failures, and expose a human-readable explanation channel alongside the latent message.

7 Conclusion

Latent-Lens connects a frozen VLM to a frozen text-only SLM using three projected depth-band states and lightweight Q/V LoRA. Across three runs, it improves substantially over matched text communication on CLEVR; GQA shows a smaller, seed-sensitive difference. A token-matched final-layer control supports cross-depth extraction on CLEVR, and four measured VLM–SLM pairings show that the same interface can be retrained across the evaluated small-model families. These results support latent prefixes as a compact visual communication channel while leaving model scale, bidirectional interaction, learned depth selection, and broader task coverage open.

Funding and Competing Interests

Microsoft supported this work through the authors’ employment. The authors declare no competing interests beyond their employment at Microsoft.

References

- Alibaba Cloud. Qwen2.5: A series of foundation models. <https://github.com/QwenLM/Qwen2.5>, 2024.
- Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-VL: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023.
- Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. InternVL: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *CVPR*, 2024.
- Abhishek Das, Théophile Gervet, Joshua Romoff, Dhruv Batra, Devi Parikh, Mike Rabbat, and Joelle Pineau. TarMAC: Targeted multi-agent communication. *ICML*, 2019.

- Lucio M. Dery, Zohar Yahav, Henry Prior, Qixuan Feng, Jiajun Shen, and Arthur Szlam. Latent space communication via K-V cache alignment. *arXiv preprint*, 2026.
- Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. *ICML*, 2024.
- Zhuoyun Du, Haoran Chen, Yushen Li, Jianguo Huang, Shan Zhong, Kaiqi Mao, and Yu Zhou. Enabling agents to communicate entirely in latent space. *arXiv preprint arXiv:2511.09149*, 2025.
- Jakob Foerster, Yannis M Assael, Nando de Freitas, and Shimon Whiteson. Learning to communicate with deep multi-agent reinforcement learning. *NeurIPS*, 2016.
- Micah Hodosh, Peter Young, and Julia Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47:853–899, 2013. doi: 10.1613/jair.3994.
- 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://arxiv.org/abs/2106.09685>.
- Xingyu Huang et al. BeMyEyes: Enabling small vision-language models to see through the eyes of large language models. *arXiv preprint arXiv:2511.19417*, 2025.
- Drew A Hudson and Christopher D Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, 2019.
- HuggingFace. SmoVLM: Small vision language models. <https://huggingface.co/HuggingFaceTB/SmoVLM-256M-Instruct>, 2024.
- Jiechuan Jiang, Ziyang Zheng, Hengyuan Hu, Shu Wang, Zhiheng Ma, and Zongzhang Zhang. Graph convolutional reinforcement learning. In *International Conference on Learning Representations*, 2019. URL <https://arxiv.org/abs/1810.09202>.
- Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*, 2017.
- Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *ICML*, 2019.
- Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. Encouraging divergent thinking in large language models through multi-agent debate. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17889–17904, 2024. doi: 10.18653/v1/2024.emnlp-main.992.
- Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *NeurIPS*, 2023.
- Xiaoze Liu et al. The vision wormhole: Latent-space communication in heterogeneous multi-agent systems. *arXiv preprint arXiv:2602.15382*, 2026.
- Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *NeurIPS*, 2022.
- Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. Chat-Dev: Communicative agents for software development. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pages 15174–15186, 2024. doi: 10.18653/v1/2024.acl-long.810.
- Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. *European Signal Processing Conference (EUSIPCO)*, 2007.

- Amanpreet Singh, Tushar Jain, and Sainbayar Sukhbaatar. Learning when to communicate at scale in multiagent cooperative and competitive tasks. *ICLR*, 2019.
- Sainbayar Sukhbaatar, Arthur Szlam, and Rob Fergus. Learning multiagent communication with backpropagation. *NeurIPS*, 2016.
- Chengxing Tang, Chang Su, Yanfeng Zhou, Zhoujun Liu, Richong Zhang, Yaowei Ma, and Chunyu Ai. Augmenting multi-agent communication with state delta trajectory. *EMNLP*, 2025.
- Ian Tenney, Dipanjan Das, and Ellie Pavlick. BERT rediscovers the classical NLP pipeline. *ACL*, 2019.
- Yanwei Yue et al. Cut the crap: An economical communication pipeline for LLM-based multi-agent systems. *ICLR*, 2025.
- Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. TinyLlama: An open-source small language model. *arXiv preprint arXiv:2401.02385*, 2024.
- Renrui Zhang, Jiaming Han, Chris Liu, Peng Gao, Aojun Zhou, Xiangfei Hu, Shilin Yan, Pan Lu, Hongsheng Li, and Yu Qiao. LLaMA-Adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023.
- Yujun Zhang, Yufei Li, Bingqian Li, Liang Zhang, and Yonghong Cai. Cache-to-cache: Direct semantic communication between LLMs. *arXiv preprint arXiv:2510.03215*, 2025.
- Yujia Zheng, Zhuokai Zhao, Zijian Li, Yaqi Xie, Mingze Gao, Lizhu Zhang, and Kun Zhang. Thought communication in multiagent collaboration. *arXiv preprint arXiv:2505.16476*, 2025.
- Pengcheng Zhou, Yinglun Feng, Halimulati Julaiti, and Zhongliang Yang. Why do AI agents communicate in human language? *arXiv preprint arXiv:2506.02739*, 2025.
- Jiaru Zou, Xiyuan Yang, Ruizhong Qiu, Gaotang Li, Katherine Tieu, Pan Lu, Ke Shen, Hanghang Tong, Yejin Choi, Jingrui He, James Zou, Mengdi Wang, and Ling Yang. Latent collaboration in multi-agent systems. *arXiv preprint arXiv:2506.05813*, 2025.

Supplementary Material

A Model Details

Table 7: Model specifications and band assignments.

Model	Layers	Hidden dim	Params	Band ranges ($B=3$)
SmolVLM-256M (VLM)	30	576	256M	L0-9 / L10-19 / L20-29
InternVL2-1B (VLM)	24	896	900M	L0-7 / L8-15 / L16-23
Qwen2.5-0.5B (SLM)	24	896	500M	-
TinyLlama-1.1B (SLM)	22	2048	1.1B	-

B Trainable Parameter Breakdown

Table 8: Trainable parameters of Latent-Lens for the Qwen2.5-0.5B pair.

Component	Params	Description
Domain encoder	1,300K	2-layer MLP: $576 \rightarrow 896$ (GELU) $\rightarrow 896$
Attention decoder (LoRA)	300K	Rank 4, q/v projections, all 24 layers
Total	1,600K	0.2% of frozen parameters

C Evaluation Details

Free-form answers are compared using `answers_match()`, which applies the following normalization pipeline: (1) lowercase; (2) strip articles (a, an, the); (3) strip punctuation; (4) normalize whitespace; (5) convert number words to digits (e.g., “three” $\rightarrow$ “3”); and (6) accept an exact string match or containment of every word in the shorter answer by the longer answer. This heuristic was validated against 50 hand-labelled CLEVR and GQA examples with 100% agreement.

Early experiments used multiple-choice formatting with random distractors, which inflated SLM-only GQA accuracy from 56.5% to 96% because the model could eliminate implausible distractors without any visual understanding. The switch to free-form generation was essential for measuring actual visual comprehension.

D Prompts and Reproducibility Details

VQA prompts. The VLM sender in the original text condition receives:

```
Question: {question}
Answer with a single word or short phrase:
```

The evidence-prompt control replaces the second line with:

```
Briefly describe the visual evidence relevant to answering the
question in one or two concise sentences. Do not answer the
question directly:
```

The text receiver receives:

```
A vision model looked at the image and responded:
"{vlm_description}"
Based on this, answer the following question with a single word or
short phrase.
{question}
Answer:
```

The Latent-Lens receiver receives the same question instruction as the SLM-only condition, preceded by the projected prefix. For Flickr8k, the sender prompt is `Describe this image.`, and the receiver is asked to generate a caption from the sender response.

Software and compute. Experiments used Python 3.9.19, PyTorch 2.8.0 with CUDA 12.8, Transformers 4.57.6, PEFT 0.17.1, Datasets 4.5.0, NumPy 2.0.2, and NLTK 3.9.2. Each run used one 24GB NVIDIA GPU. The reported VQA variability is the sample standard deviation across three complete training and evaluation runs with the fixed dataset splits; auxiliary ablations, model-pair checks, latency measurements, and Flickr8k results are single-run unless stated otherwise.

E Additional Figures

This section collects the ablation plots referenced from the main text.

F ScienceQA Classification

G Cross-Attention Analysis (ScienceQA)

H Qualitative Examples: Text vs. Latent Communication

Tables 10 to 12 show samples from CLEVR, GQA, and Flickr8k, comparing what each condition receives. In text communication, the VLM generates a response and the SLM reasons or generates from that text. In Latent-Lens, three band summaries (1,728 floats total for SmolVLM) are projected into the SLM.

Ablation: Number of Spectral Bands

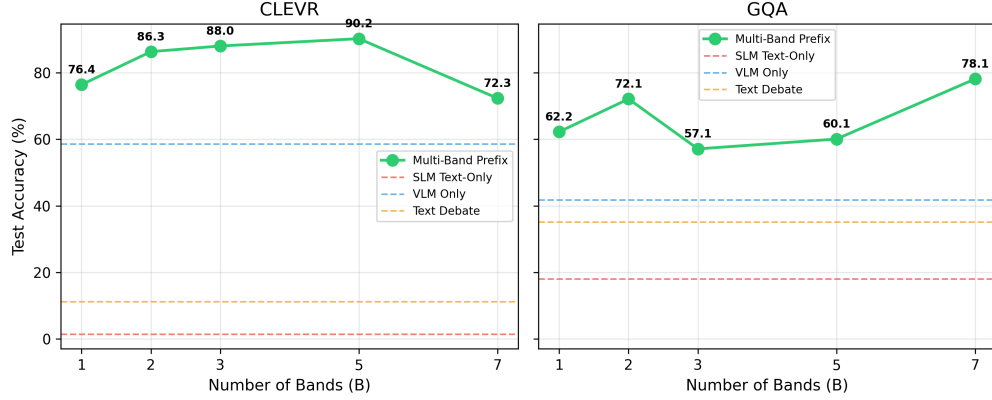


Figure 2: Ablation over the number of spectral bands B . The default $B = 3$ is highlighted in orange. Dashed lines show the SLM+LoRA baseline.

Ablation: LoRA Rank ($B=5$)

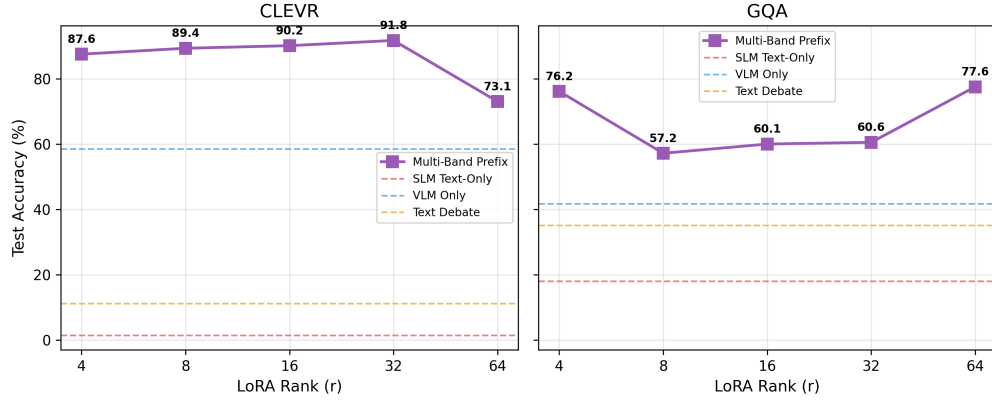


Figure 3: Ablation over LoRA rank r . The default $r = 4$ is highlighted in orange. Dashed lines show the SLM+LoRA baseline.

On VQA, the sender responses are frequently single numbers or words. On Flickr8k, they are much longer narratives and can include unsupported details. These examples illustrate two ways a generated-text handoff can omit or distort task-relevant evidence; they do not establish that every latent feature is useful or faithfully decoded.

In the **Latent-Lens** condition, the SLM prompt is identical to the SLM-only prompt (“Answer the following question...” for VQA, or “Describe this image.” for captioning), but is preceded by three projected prefix tokens from early, middle, and late VLM depth bands. The tokens are not human-readable, which motivates the auditability concerns in Section 6.

Table 9: ScienceQA classification accuracy on the multimodal subset (5-way).

Condition	Params	Accuracy (%)
VLM only	150K	56.2
SLM only	231K	55.0
Concat $\rightarrow$ MLP	445K	64.1
Cross-attention debate	1,039K	65.9

Cross-Attention Weights: All Bands (no select)

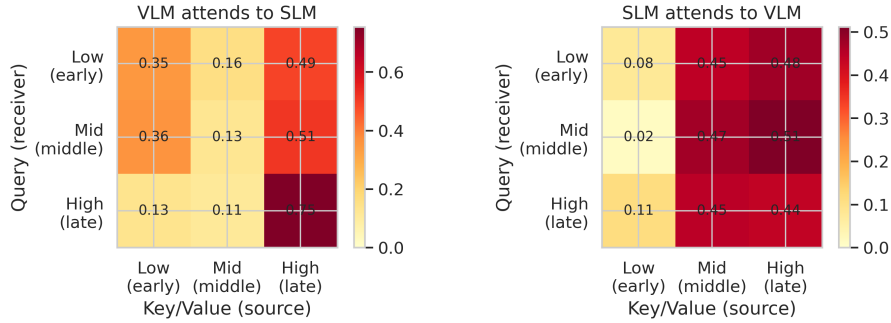


Figure 4: Average cross-attention weights from the ScienceQA classification experiment.

Table 10: CLEVR examples. Gold answers shown for reference.

Question	Gold	VLM resp.	SLM prompt (text communication)
How many objects are either yellow shiny spheres or blue metallic cubes?	2	“1.”	A vision model responded: “1.” Based on this, answer: How many objects...
How many tiny things are metal cylinders or yellow matte spheres?	2	“Metal.”	A vision model responded: “Metal.” Based on this, answer: How many tiny...
How many items are there in the described scene?	8	“7.”	A vision model responded: “7.” Based on this, answer: How many items...
What number of other objects are the same size as the purple rubber cylinder?	7	“4.”	A vision model responded: “4.” Based on this, answer: What number of...

Table 11: GQA examples. Gold answers shown for reference.

Question	Gold	VLM resp.	SLM prompt (text communication)
Does the striped shirt look long sleeved and white?	no	“No.”	A vision model responded: “No.” Based on this, answer: Does the striped...
Does the table in front of the other table look black and round?	yes	“No.”	A vision model responded: “No.” Based on this, answer: Does the table...
Is the toilet paper to the right of the chair resting on a lamp?	no	“Yes.”	A vision model responded: “Yes.” Based on this, answer: Is the toilet paper...
Which kind of animal sits atop the chair?	cat	“Cat”	A vision model responded: “Cat” Based on this, answer: Which kind of animal...

Table 12: Flickr8k examples. Gold is one of the five reference captions; the prompt to the VLM is “Describe this image.”. In text communication, the SLM receives a prompt of the form “Generate a caption for the image based on the context.\n A vision model responded: *<VLM resp>*. Describe this image.\n Caption:”. The VLM’s narrative responses do not match the short, declarative reference style and frequently hallucinate specific details (e.g., jersey number 28 below), explaining the BLEU-4 collapse of text communication on this benchmark.

Gold caption (1 of 5 refs)	VLM response (text-communication channel)
The dogs are in the snow in front of a fence.	“In the center of the image, we can see two dogs are fighting. In the background, we can see the snow, fencing, trees, and the sky.”
A brown and white dog swimming towards some in the pool.	“In the image, a person is playing with a dog in a pool. The person is wearing athletic shorts and a jersey with the number ‘3’ on it. The dog is brown and white, and it is holding a leash in its...”
A man is wearing a Sooners red football shirt and helmet.	“In the image, we can see a person wearing a red sports jersey with the number 28 on it. The person is standing and the background is blurred.”
A group of tourists stand around as a lady puts her hand near the mouth of a statue.	“The image depicts a group of five individuals, all of whom are standing or seated, engaged in a dynamic moment. The setting appears to be outdoors, possibly a park or a public space, given the presence of a fence and a river in the background...”

NeurIPS Paper Checklist

1. Claims

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

Answer: [\[Yes\]](#)

Justification: The abstract and introduction state the evaluated scope, three-run VQA results, seed-sensitive GQA result, and small-model compatibility claim; limitations are detailed in Section 5.

Guidelines:

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

2. Limitations

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

Answer: [\[Yes\]](#)

Justification: Section 5 discusses model and dataset scope, white-box access, pair-specific training, metric limitations, single-run auxiliary results, and task-dependent design choices.

Guidelines:

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

3. Theory assumptions and proofs

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

Answer: [N/A]

Justification: The paper presents an empirical method and descriptive representation analyses, without formal theoretical results or proofs.

Guidelines:

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

4. Experimental result reproducibility

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

Answer: [Yes]

Justification: The architecture, model identifiers, data sources and splits, prompts, optimization settings, evaluation heuristic, software versions, and run aggregation are specified in Sections A and 3 to D.

Guidelines:

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

5. Open access to data and code

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

Answer: [No]

Justification: The datasets and pretrained models are publicly linked, but the experiment code and trained projection checkpoints are not included with this paper.

Guidelines:

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

6. Experimental setting/details

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

Answer: [Yes]

Justification: Sections A and C to D specify the fixed splits, prompts, model dimensions, optimizer, schedule, batch size, epochs, decoding setup, and evaluation.

Guidelines:

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

7. Experiment statistical significance

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

Answer: [Yes]

Justification: Tables 1 and 2 report mean and sample standard deviation over three complete runs for the main VQA comparison and depth control; Section D identifies the variability source and marks auxiliary single-run results.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

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

8. Experiments compute resources

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

Answer: [No]

Justification: Section D reports the single-GPU memory class and software environment, but exact per-run training time and total project compute were not retained.

Guidelines:

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

9. Code of ethics

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

Answer: [Yes]

Justification: The work uses public datasets and pretrained models for empirical research and follows the NeurIPS Code of Ethics.

Guidelines:

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

10. Broader impacts

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

Answer: [Yes]

Justification: Section 6 discusses potential efficiency and accessibility benefits together with auditability, privacy, bias, hallucination, and deployment risks and possible mitigations.

Guidelines:

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

11. Safeguards

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

Answer: [N/A]

Justification: The paper does not release a high-risk generative model or a newly collected dataset; it evaluates lightweight adapters over existing public models.

Guidelines:

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

12. Licenses for existing assets

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

Answer: [No]

Justification: The paper cites and links the datasets and pretrained models, but it does not enumerate the license and terms of use for every existing asset.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.

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

13. **New assets**

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

Answer: [N/A]

Justification: The paper does not introduce or distribute a new dataset, pretrained foundation model, or public software package.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

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

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

Answer: [N/A]

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

Guidelines:

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

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

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

Answer: [N/A]

Justification: The research does not involve human subjects, so IRB review is not applicable.

Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [N/A]

Justification: LLM assistance was limited to writing and editing and did not form an original or non-standard component of the research method.

Guidelines:

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