WHAT DOES FULL ATTENTION ADD? STATE-COMPLEMENTARY VISUAL KV COMPRESSION FOR HYBRID VISION-LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-language models are increasingly adopting hybrid architectures that interleave full-attention layers with recurrent modules. Recurrent modules summarize historical information in fixed-capacity states, while full attention preserves direct access to historical visual tokens through explicit key–value (KV) caches. Existing visual KV compression methods typically assess entry importance through visual contributions to full attention, without explicitly accounting for information already represented in recurrent states. This motivates us to consider the additional information that full attention provides beyond recurrent-state representations when compressing visual KV entries in hybrid architectures. We propose Full-Attention Extra Semantic Information (FA-ESI), a training-free method for active visual KV compression that explicitly models this complementarity. For a target full-attention layer, FA-ESI replays the preceding Gated DeltaNet on the target layer's normalized inputs to construct an auxiliary recurrent state. It estimates full-attention extra information by comparing FA values with the corresponding state readouts. Current attention weights condition this information on the query. Projection onto the subspace of top-ranked prediction candidates followed by alignment scoring yields token-level FA-ESI scores for dynamic visual KV selection during decoding. Across benchmarks spanning multi-image understanding, long-video understanding, and GUI-agent tasks, FA-ESI achieves approximately 95% compression of active visual KV entries while retaining 98.984%–99.243% of Vanilla's performance on average.

1 INTRODUCTION

Hybrid attention architectures are increasingly being adopted in large vision-language models. These architectures interleave recurrent layers, such as Gated Delta Networks (GDNs), with full-attention layers. For example, Qwen3.5 repeats a pattern of three GDN layers followed by one full-attention layer. A similar arrangement appears in Kimi Linear, which interleaves three Kimi Delta Attention (KDA) layers with one global Multi-Head Latent Attention (MLA) layer. Recurrent layers summarize preceding tokens in fixed-size states, while full-attention layers retain explicit key–value (KV) entries that can be accessed by subsequent queries.

For inputs containing multiple images or video frames, the visual KV cache grows with the visual sequence length. Selecting a subset of visual KV entries for subsequent decoding therefore reduces the number of visual KV entries accessed by full attention. The challenge is to decide which entries to retain while preserving the visual evidence needed for the task.

In hybrid architectures, visual information can persist both in compressed recurrent states and in explicit KV entries. These two forms of memory may therefore retain overlapping information about the visual input. For example, in the scenario shown in Figure 1, background information outside the notebook and cube regions may already be captured by the recurrent state, while identifying the yellow notebook and the Professor's Cube, as well as reading "OA-mate" on the notebook's cover, may require additional information from visual KV entries. A visual token may receive substantial attention even when some of the content retrieved through its KV entry is already represented in recurrent memory. In this case, its attention contribution reflects both information represented in

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.

the recurrent state and information that the state does not adequately capture. Attention strength alone therefore does not indicate how much useful content that entry provides beyond the recurrent state. This motivates selecting visual KV entries based on their additional contributions beyond recurrent-state representations.

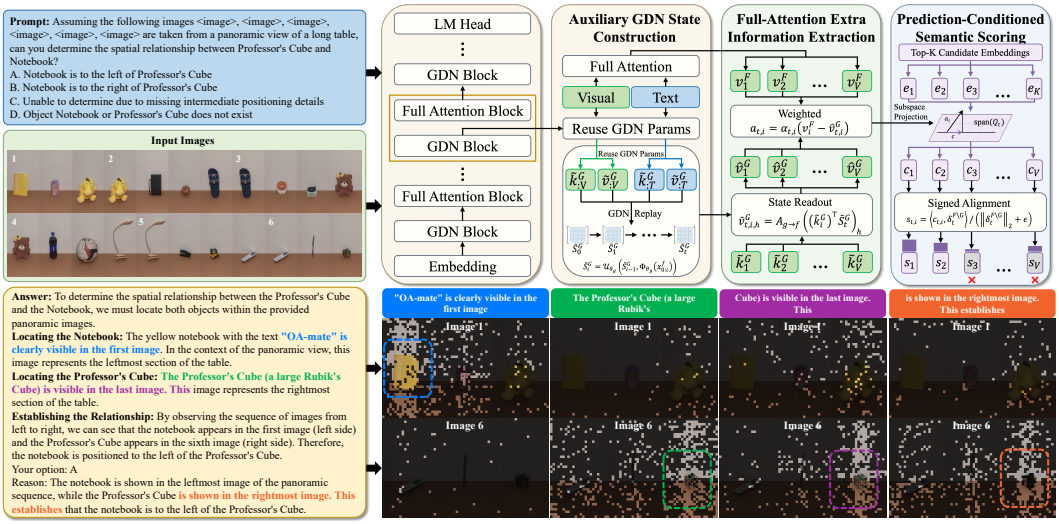


Figure 1: Overview of FA-ESI with a multi-image spatial reasoning example. Stage I constructs an auxiliary GDN state and extracts full-attention extra information; Stage II performs prediction-conditioned semantic scoring to obtain FA-ESI scores. The lower panels show retained visual tokens in Images 1 and 6 at four points during decoding; colored text links each panel to the corresponding generated passage.

To this end, we introduce Full-Attention Extra Semantic Information (FA-ESI), a training-free method for active visual KV compression that explicitly models this complementarity. FA-ESI combines full-attention extra information extraction with prediction-conditioned semantic scoring. For a target full-attention layer, we replay the preceding GDN on the target layer's normalized inputs to construct an auxiliary recurrent state. We estimate full-attention extra information by comparing the FA values with head-concatenated historical-key readouts from this state. Current attention weights, together with the FA output gate and projection, produce query-conditioned full-attention extra information. We then project it onto a subspace spanned by the model's top-ranked prediction candidates and compute each token's FA-ESI score from the alignment of its projected contribution with the aggregate projected contribution. The resulting scores guide the selection of visual KV entries during decoding.

We evaluate FA-ESI across 14 benchmarks covering single-image and multi-image understanding, long-video understanding, multi-step reasoning, and GUI-agent tasks, examining the trade-off between task performance and active visual KV compression during decoding. Across Qwen3.5-2B and Qwen3.5-4B, FA-ESI compresses active visual KV entries by approximately 95% while retaining 98.984%–99.243% of Vanilla's performance on average. It achieves the highest or tied-highest scores among the compared compression methods on 12 of the 14 benchmarks for each model. Across the evaluated cases with output lengths from 512 to 2048 tokens, FA-ESI delivers 1.19–1.22× decoding throughput relative to Vanilla, including scoring, selection, and KV update overhead. Separate end-to-end evaluations show 1.11–1.18× speedups with matched output lengths.

Our contributions are twofold:

1. We introduce FA-ESI, a training-free method for active visual KV compression that estimates full-attention extra semantic information to guide visual KV selection during decoding.
2. Experiments across image understanding and reasoning, video understanding, and GUI-agent tasks show that FA-ESI preserves task performance under substantial active visual

KV compression, with additional efficiency evaluations demonstrating improved decoding throughput and reduced end-to-end latency.

2 Related Work

Recurrent modules summarize historical information in fixed-size states (Yang et al., 2025b; Gu & Dao, 2024; Dao & Gu, 2024). Hybrid architectures combine recurrent state-based attention with full attention, as exemplified by Kimi Linear and the Qwen3.5 family (Kimi Team et al., 2025; Qwen Team, 2026). In controlled studies of hybrid language models, Qiao et al. (2026) find that full-attention layers primarily support long-range retrieval, while efficient-attention modules influence how this capability develops during training.

To reduce visual redundancy, existing methods compress visual tokens or their KV caches at different stages of multimodal inference. Before visual features enter the language model, LLaVA-PruMerge and VisionZip shorten visual sequences through token selection and feature merging (Shang et al., 2025; Yang et al., 2025a). Within the language model, FastV prunes visual tokens according to intermediate-layer attention scores, while SparseVLM combines text-guided importance estimation, layer-adaptive sparsification, and token recycling to reduce the number of visual tokens processed by subsequent layers (Chen et al., 2024; Zhang et al., 2025b).

Recent work extends visual selection to autoregressive decoding to accommodate changing visual requirements during generation. SIVS assesses the semantic impact of visual inputs on the output by projecting visually induced hidden-state changes onto a prediction-conditioned semantic subspace defined by current output candidates, and periodically updates the active visual KV set (Wang et al., 2026). VisionPulse selects visual tokens at each decoding step and adapts the retention budget according to the current visual attention mass (Xu et al., 2026). Our method estimates full-attention extra semantic information by combining auxiliary recurrent-state readouts with prediction-conditioned semantic scoring to guide dynamic visual KV selection.

3 Method

3.1 Problem Setup and Overview

Consider a hybrid vision-language model that interleaves recurrent and full-attention layers. Let f denote the target full-attention (FA) layer and g its nearest preceding Gated DeltaNet (GDN) layer. FA-ESI combines full-attention extra information extraction with prediction-conditioned semantic scoring to guide periodic, fixed-budget visual KV selection during decoding (Figure 1).

Let $\mathcal{H}_t$ denote the history accessible at step t, and let $\mathcal{V} \subseteq \mathcal{H}_t$ be the set of visual-token positions. For FA head h, the attention weight assigned to position i is

$$\alpha^F_{t,i,h} = \frac{\exp\left((q^F_{t,h})^\top k^F_{i,h}/\sqrt{d_k} + m_{t,i}\right)}{\sum_{j\in\mathcal{H}_t}\exp\left((q^F_{t,h})^\top k^F_{j,h}/\sqrt{d_k} + m_{t,j}\right)}, \tag{1}$$

where $q^F_{t,h}$ is the current query, $k^F_{i,h}$ is the key at position i, d_k is the key dimension, and $m_{t,i}$ is the attention mask. With output gate γ_t and output projection W^O, the FA output is

$$o^F_t = W^O\left(\gamma_t \odot \mathrm{Concat}_h\left[\sum_{i\in\mathcal{H}_t}\alpha^F_{t,i,h}v^F_{i,h}\right]\right), \tag{2}$$

where $v^F_{i,h}$ is the corresponding value vector, Concat_h concatenates the head outputs, and $\odot$ denotes element-wise multiplication.

3.2 Stage I: Full-Attention Extra Information Extraction

GDN state replay. Let h^f_i denote the residual stream entering decoder block f, and let $x^f_i = \mathrm{RMSNorm}_f(h^f_i)$ be the input to its FA module. We apply the preceding GDN's frozen projections

and gated-delta update to the recorded x_i^f sequence to construct an independent auxiliary state. The GDN QKV projection, causal convolution and SiLU activation produce auxiliary keys and values, with keys further L2-normalized:

$$\widetilde{S}_i^G = \mathcal{U}_{\theta_g}\left(\widetilde{S}_{i-1}^G, \Phi_{\theta_g}(x_{0:i}^f)\right), \qquad \widetilde{S}_{-1}^G = 0. \tag{3}$$

where Φ_{θ_g} denotes the GDN feature transformation and $\mathcal{U}_{\theta_g}$ its recurrent update. We replay the complete multimodal prompt once during prefill and incrementally update the auxiliary state during decoding.

Full-attention extra information. The delta update can be viewed as an online gradient step on the key–value association loss $\mathcal{L}_i(S) = \frac{1}{2}\|S^\top k_i^G - v_i^G\|_2^2$ (Schlag et al., 2021; Yang et al., 2024). A historical key therefore retrieves a value estimate from the current auxiliary memory. For visual token i, we read the current post-write auxiliary state $\widetilde{S}_t^G$ and map the readouts to the FA query-head layout:

$$\widehat{v}_{t,i,h}^G = \mathcal{A}_{g\to f}\left((\widetilde{k}_i^G)^\top \widetilde{S}_t^G\right)_h. \tag{4}$$

For the evaluated Qwen3.5 models, the mapping $\mathcal{A}_{g\to f}$ concatenates adjacent pairs of 128-dimensional GDN state-head readouts into 256-dimensional vectors, without additional projection or readout normalization. We estimate full-attention extra information relative to this auxiliary recurrent reference as

$$\rho_{t,i,h} = v_{i,h}^F - \widehat{v}_{t,i,h}^G. \tag{5}$$

As the auxiliary state incorporates the decoding prefix, the estimated extra information associated with the same visual token can change across decoding steps.

Query-conditioned full-attention extra information. We weight the estimated extra information by the current attention weights and apply the FA output gate and projection to obtain query-conditioned full-attention extra information:

$$a_{t,i} = W^O\left(\gamma_t \odot \mathrm{Concat}_h\left[\alpha_{t,i,h}^F \rho_{t,i,h}\right]\right). \tag{6}$$

The resulting $a_{t,i}$ lies in the FA output space. With attention weights, the output gate and projection held fixed, let o_t^{sub} denote the output obtained by replacing only visual FA values with the corresponding auxiliary state readouts. Then

$$o_t^F - o_t^{\mathrm{sub}} = \sum_{i\in\mathcal{V}} a_{t,i}. \tag{7}$$

3.3 Stage II: Prediction-Conditioned Semantic Scoring

To assess the semantic relevance of the query-conditioned full-attention extra information $a_{t,i}$, we project it onto the subspace spanned by the output embeddings of the current prediction candidates. Let z_t be the vocabulary logits produced by the current main forward pass, and select the K highest-scoring candidate tokens:

$$\mathcal{C}_t = \mathrm{TopK}(z_t, K). \tag{8}$$

Let e_y denote the output embedding of vocabulary token y. The selected candidates form the matrix

$$M_t = [e_{y_1}, e_{y_2}, \ldots, e_{y_K}], \qquad y_k \in \mathcal{C}_t. \tag{9}$$

We obtain an orthonormal basis for the candidate subspace through QR decomposition and define the prediction-conditioned projector:

$$M_t = Q_t R_t, \qquad Q_t^\top Q_t = I, \qquad P_t = Q_t Q_t^\top. \tag{10}$$

We project each visual token's query-conditioned extra information onto this semantic subspace and aggregate the projected contributions:

$$c_{t,i} = P_t a_{t,i}, \qquad \delta_t^{F\setminus G} = \sum_{i\in\mathcal{V}} c_{t,i}. \tag{11}$$

The projection $c_{t,i} = P_t a_{t,i}$ retains the component of $a_{t,i}$ along the current candidate output directions. Finally, we define the Full-Attention Extra Semantic Information (FA-ESI) score as

$$s_{t,i} = \frac{\left\langle c_{t,i}, \delta_t^{F\setminus G} \right\rangle}{\left\| \delta_t^{F\setminus G} \right\|_2 + \epsilon}, \tag{12}$$

where $\epsilon > 0$ is a small constant for numerical stability. The score measures the signed alignment of each token's projected contribution with the aggregate contribution in the prediction-conditioned semantic subspace. Visual KV entries with higher FA-ESI scores are preferentially retained.

3.4 Periodic Fixed-Budget Visual KV Selection

FA-ESI computes the initial scores after prefill and refreshes them every T decoding steps. At each refresh, we select the $B = \max(1, \lceil rN_v \rceil)$ highest-scoring visual tokens, where N_v is the number of candidate visual tokens and r is the retention ratio. The selected indices are shared across all FA layers, whose corresponding original KV entries remain active from the next decoding step until the next refresh. All visual KV entries remain cached for subsequent rescoring, allowing previously inactive tokens to be selected again. Text and generated-token KV entries are retained, and the native GDN states continue their normal updates. Selection uses only the prompt and already processed generated tokens, preserving autoregressive causality.

4 Experiments

4.1 Experimental Setup

We conduct the main experiments on Qwen3.5-2B and Qwen3.5-4B (Qwen Team, 2026), with additional evaluations on Qwen3.5-0.8B, 9B, 27B, and 35B-A3B. The main experiments cover 14 benchmarks spanning image understanding and reasoning, video understanding, and GUI-agent tasks (Table 1). Dataset descriptions and references are provided in Appendix A. We further evaluate FA-ESI on NVIDIA Nemotron Nano V2 VL (NVIDIA et al., 2025) to assess its applicability to a hybrid architecture that combines Mamba-2 with full attention. We compare FA-ESI against Vanilla, which uses all visual KV entries in full attention, and six compression methods: SIVS (Wang et al., 2026), FastV (Chen et al., 2024), LOOK-M (Wan et al., 2024), SparseVLM (Zhang et al., 2025b), AirCache (Huang et al., 2025), and VisPruner (Zhang et al., 2025a). Baselines follow their original papers or released implementations, using the best-performing hyperparameter configurations reported in the respective papers.

The main experiments use a visual KV retention ratio of $r = 0.05$, a score refresh interval of $T = 10$, and $K = 50$ prediction candidates. The target FA layers are 15 for Qwen3.5-2B and 19 for Qwen3.5-4B, with zero-based indexing.

4.2 Main Results

FA-ESI maintains performance on multi-image understanding, long-video understanding, and GUI-agent tasks under substantial active visual KV compression (Table 1). At approximately 95% compression of active visual KV entries, it retains 99.243% and 98.984% of Vanilla's performance on Qwen3.5-2B and Qwen3.5-4B, respectively, averaged across 14 benchmarks. FA-ESI achieves both a higher average compression ratio and higher average relative performance than FastV, LOOK-M, SparseVLM, and VisPruner on both models. AirCache achieves higher average compression ratios on both models, as does SIVS on Qwen3.5-4B, but each retains less of Vanilla's performance on average than FA-ESI under the evaluated configurations. FA-ESI also achieves the highest scores among the evaluated compression methods on all four GUI-agent benchmarks, while remaining close to Vanilla on multi-image and long-video understanding tasks. Figure 2 further illustrates FA-ESI's favorable accuracy–compression trade-off relative to the evaluated baselines on TextVQA, with higher accuracy than SIVS in the highlighted high-compression range for both models.

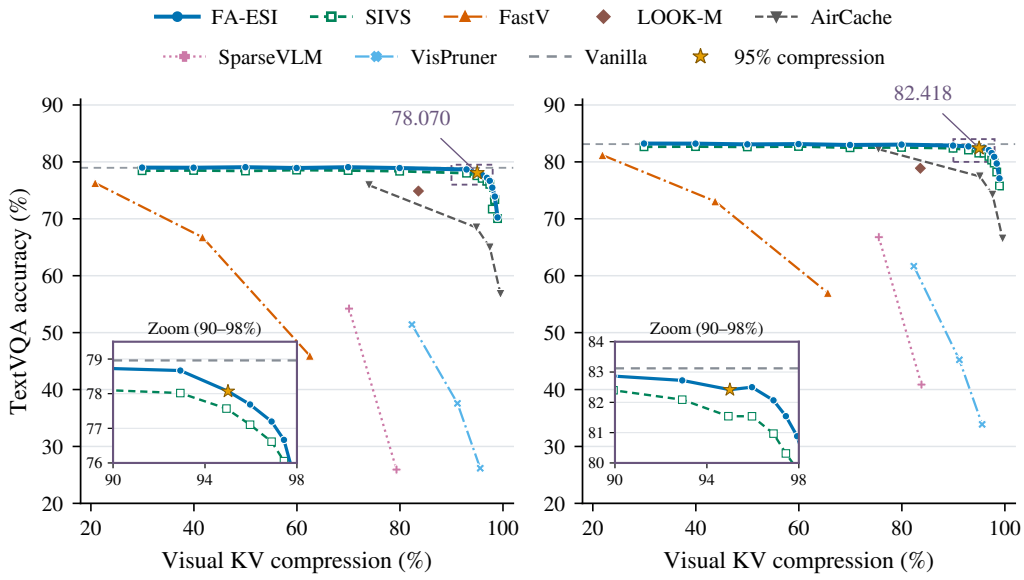
On Qwen3.5-0.8B, 9B, 27B, and 35B-A3B (Table 2), FA-ESI retains 95.436%–99.432% of Vanilla's performance at approximately 95% compression, averaged over the benchmarks reported for each

Table 1: Comparison of FA-ESI with other visual KV compression methods on Qwen3.5-2B and Qwen3.5-4B. Rel. is the mean performance relative to Vanilla across benchmarks; Comp. denotes the percentage reduction in visual KV entries used in the main full-attention computation. Bold indicates the highest values among compression methods, including ties.

Qwen3.5-2B								
Benchmark	Vanilla	FastV	LOOK-M	SparseVLM	AirCache	VisPruner	SIVS	**Ours**
Image understanding and reasoning								
MMBench	71.749	55.031	70.511	69.195	71.440	62.074	71.517	**71.827**
TextVQA	78.960	45.822	61.310	25.384	57.122	25.242	76.230	**78.070**
MileBench	53.349	48.243	47.116	40.409	47.300	36.818	51.788	**52.936**
MuirBench	42.192	34.923	41.154	29.385	41.462	30.846	40.808	**41.769**
MMRB	59.550	48.410	51.410	41.020	46.290	38.850	56.280	**58.130**
MIRBench	41.118	36.748	40.097	35.749	39.756	29.502	**40.893**	40.832
Video understanding								
Video-MME	50.056	45.954	49.738	45.287	50.000	42.604	50.112	**50.150**
LongVideoBench	46.597	43.231	46.298	44.577	46.223	40.239	**46.672**	46.372
MMBench-Video	1.14	0.88	1.04	0.65	0.89	0.66	1.02	**1.10**
MVBench	55.263	47.684	55.158	46.842	54.526	45.158	54.947	**55.895**
GUI-agent tasks								
MMBench-GUI	74.572	33.871	20.004	3.732	10.416	1.308	74.239	**74.627**
ScreenSpot-Pro	44.529	25.617	3.605	0.506	0.886	0.190	39.405	**43.896**
UI-Vision	26.665	8.706	11.772	0.986	1.825	0.347	24.001	**26.410**
AndroidCtrl.	41.905	38.830	24.001	21.186	19.279	15.750	40.750	**42.008**
Rel.	100.000%	75.351%	76.552%	58.437%	69.850%	53.553%	96.318%	**99.243%**
Comp.	–	62.551%	90.958%	78.560%	**99.566%**	94.727%	92.081%	94.973%
Qwen3.5-4B								
Benchmark	Vanilla	FastV	LOOK-M	SparseVLM	AirCache	VisPruner	SIVS	**Ours**
Image understanding and reasoning								
MMBench	81.037	67.260	79.799	77.090	**81.037**	73.607	80.805	80.805
TextVQA	83.122	56.982	67.894	39.104	68.214	32.160	81.868	**82.418**
MileBench	48.552	45.276	42.933	36.873	41.884	31.875	47.517	**47.907**
MuirBench	54.115	52.538	53.654	44.308	54.038	46.000	53.654	**54.885**
MMRB	68.008	56.370	59.620	43.906	53.747	42.181	63.585	**64.580**
MIRBench	46.300	42.940	46.311	38.823	45.477	32.005	46.882	**47.255**
Video understanding								
Video-MME	56.446	52.975	**56.322**	52.552	56.301	47.755	56.209	56.025
LongVideoBench	51.309	48.168	50.935	48.018	50.935	45.999	50.935	**51.085**
MMBench-Video	1.61	1.41	**1.60**	1.20	1.54	0.96	**1.60**	**1.60**
MVBench	61.158	57.158	61.158	52.842	61.158	50.737	61.158	**61.474**
GUI-agent tasks								
MMBench-GUI	69.555	31.597	50.308	1.534	42.153	1.030	69.999	**70.111**
ScreenSpot-Pro	59.140	35.421	25.553	0.696	6.831	0.190	56.610	**58.634**
UI-Vision	32.615	19.949	23.946	0.931	17.558	0.274	31.046	**31.283**
AndroidCtrl.	50.778	44.979	33.809	21.004	29.268	17.268	45.797	**48.352**
Rel.	100.000%	81.525%	86.385%	60.259%	80.267%	54.630%	97.853%	**98.984%**
Comp.	–	65.616%	90.758%	84.009%	**99.550%**	94.729%	95.073%	94.972%

model. These results show that FA-ESI maintains most of Vanilla's performance under substantial active visual KV compression across the evaluated model scales.

We further evaluate FA-ESI on Nemotron Nano V2 VL, which combines Mamba-2 with full attention. As shown in Table 3, FA-ESI retains 96.085% of Vanilla's performance on average across the four benchmarks at approximately 95% compression of active visual KV entries. These results demonstrate that FA-ESI can extend to hybrid vision-language models that combine full attention with recurrent modules other than GDN.



(a) Qwen3.5-2B (b) Qwen3.5-4B

Figure 2: TextVQA accuracy versus visual KV compression on (a) Qwen3.5-2B and (b) Qwen3.5-4B. Stars mark approximately 95% compression.

Table 2: Extended evaluation of FA-ESI on Qwen3.5 models. Rel. reports Ours as a percentage of Vanilla for each benchmark; Comp. denotes the percentage reduction in visual KV entries used in the main full-attention computation.

	Qwen3.5-0.8B				**Qwen3.5-9B**			
Benchmark	Vanilla	Ours	Rel. (%)	Comp. (%)	Vanilla	Ours	Rel. (%)	Comp. (%)
MMBench	56.502	56.115	99.315	94.793	85.913	85.913	100.000	94.672
TextVQA	69.534	68.886	99.068	94.934	85.650	85.250	99.533	94.934
MuirBench	28.923	30.423	105.186	94.986	65.192	65.000	99.705	94.938
MMRB	40.55	38.77	95.610	94.698	68.840	66.520	96.630	94.986
MVBench	51.684	51.789	100.203	94.984	65.368	65.053	99.518	94.983
Video-MME	45.667	45.815	100.324	95.000	66.333	66.111	99.665	94.991
MMBench-GUI	33.190	31.875	96.038	94.993	82.223	78.114	95.003	94.992
UI-Vision	12.046	9.363	77.727	94.994	36.625	33.146	90.501	94.990
AndroidCtrl.	40.036	34.211	85.451	94.981	55.138	52.543	95.294	94.981
	Qwen3.5-27B				**Qwen3.5-35B-A3B**			
Benchmark	Vanilla	Ours	Rel. (%)	Comp. (%)	Vanilla	Ours	Rel. (%)	Comp. (%)
MMBench	89.319	89.474	100.174	94.917	85.913	86.223	100.361	94.908
TextVQA	86.676	85.736	98.916	94.934	86.462	85.642	99.052	94.934
MVBench	69.895	69.895	100.000	94.986	68.842	68.632	99.695	94.985
MMBench-GUI	88.671	86.868	97.967	94.992	81.205	66.042	81.328	94.991
Video-MME	73.444	73.519	100.102	94.991	61.473	61.556	100.135	94.998

Table 3: Results on Nemotron Nano V2 VL at approximately 95% compression of active visual KV entries.

Method	MMBench	TextVQA	MVBench	MMBench-GUI
Vanilla	84.056	84.896	69.263	15.500
FA-ESI	83.669	84.796	69.158	13.186

4.3 Ablation Studies

To assess the scoring design, we compare FA-ESI with variants based on full-attention extra information (FA-EI), query-conditioned full-attention extra information (Q-FA-EI), and auxiliary GDN readouts (GDN-readout), alongside attention-based and random selection. Table 4 shows that FA-ESI outperforms these alternatives on both models, supporting the combination of extra-information estimation and prediction-conditioned semantic scoring. Shared selection also outperforms per-layer selection; consistent with Figure 3(a), this gap may arise from less reliable token selection at shallow layers.

Table 4: Component ablations and selection strategies.

Benchmark	**FA-ESI**	FA-EI	Q-FA-EI	Attention	Random	GDN-readout	Per-layer
TextVQA 2B	**78.070**	53.806	73.020	76.590	53.380	50.772	74.266
TextVQA 4B	**82.418**	63.994	77.978	81.626	60.944	60.828	81.082

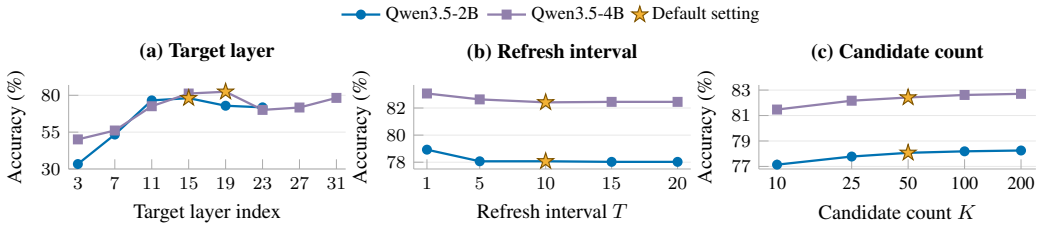


Figure 3: Effect of (a) the target layer, (b) the refresh interval T, and (c) the prediction candidate count K on TextVQA accuracy. Stars mark the main-experiment settings.

Figure 3 shows that intermediate-to-deeper target layers outperform shallow layers, while accuracy remains stable for $T = 5$–20 and gains from increasing K diminish beyond 50.

4.4 Mechanism Analysis

FA-ESI's strong performance on multi-image understanding, long-video understanding, and GUI-agent tasks demonstrates its ability to retain task-relevant visual tokens from long visual sequences. As shown in Figure 4, across three ImageNeedleInAHaystack examples with different sequence lengths and target-image positions, both Qwen3.5-2B and Qwen3.5-4B assign their highest scores to tokens within the target image. With only 5% of all visual tokens retained, most of the password region is preserved, even in the longest sequence with a late target. These results suggest that the FA-ESI score helps prioritize task-relevant visual tokens by combining estimated full-attention extra information with prediction-conditioned relevance.

4.5 Efficiency Analysis

We evaluate decoding throughput and end-to-end latency on Qwen3.5-2B and Qwen3.5-4B using an NVIDIA GeForce RTX 3090 GPU, retaining 5% of visual KV entries. As shown in Figure 5, FA-ESI achieves 1.19–1.22× the decoding throughput of Vanilla at fixed output lengths, including scoring, selection, and KV update overhead. With Vanilla and FA-ESI generating the same number of output tokens in each comparison, FA-ESI achieves end-to-end speedups of 1.11–1.18× across the evaluated settings.

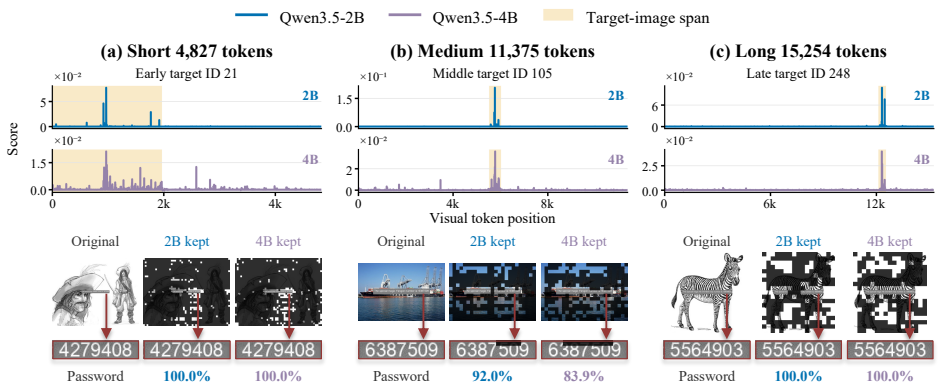


Figure 4: FA-ESI scores and retained visual regions on ImageNeedleInAHaystack at a 5% visual-token budget. Percentages indicate the fraction of visual tokens within the password region that are retained.

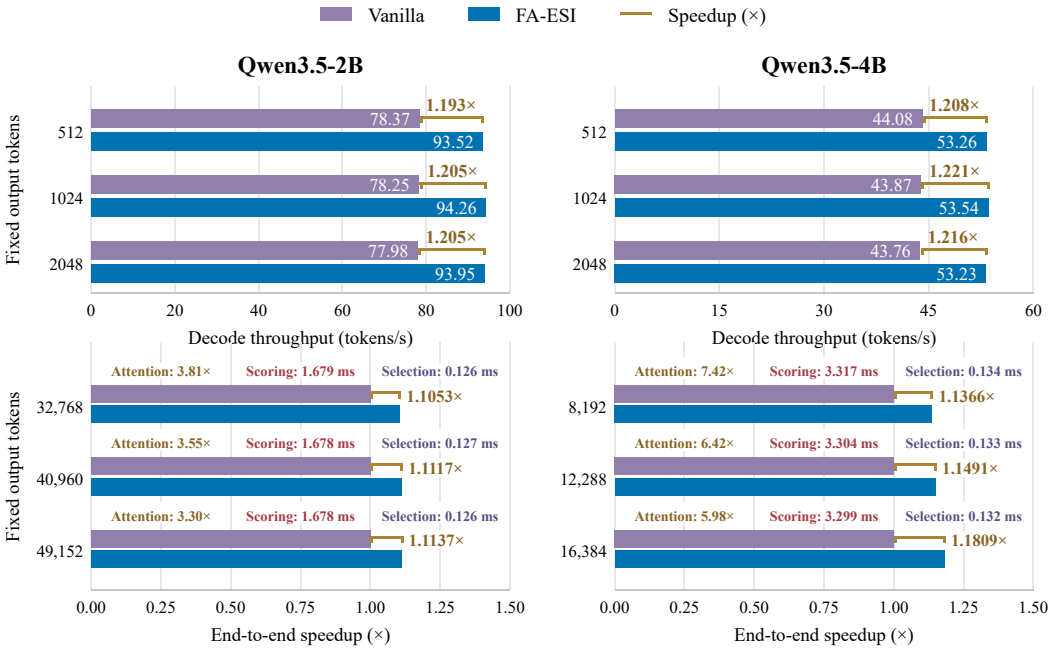


Figure 5: Decoding throughput (top) and end-to-end speedup (bottom) on Qwen3.5-2B and Qwen3.5-4B. Both methods use the same fixed output length in each comparison. Speedups are relative to Vanilla. Scoring and selection times are measured per refresh (ms).

5 Conclusion

We presented FA-ESI, a training-free method for active visual KV compression that estimates full-attention extra semantic information to guide visual KV selection. By combining full-attention extra information extraction with prediction-conditioned semantic scoring, FA-ESI scores and selects visual KV entries during decoding. In the main evaluation across 14 benchmarks, FA-ESI retains 98.984%–99.243% of Vanilla's performance on average at approximately 95% compression of active visual KV entries. Extended evaluations demonstrate its applicability across model scales and hybrid architectures with different recurrent modules. Efficiency evaluations show 1.19–1.22× decoding throughput and 1.11–1.18× end-to-end speedups relative to Vanilla, with matched output lengths in each comparison. These results support using full-attention extra semantic information to select active visual KV entries in hybrid architectures.

AI USE STATEMENT

We used generative AI tools to assist with manuscript translation, language polishing, and formatting; identify relevant literature; draft and revise portions of the manuscript based on author-provided methods and experimental results; discuss methodological explanations and interpret experimental results; prepare figures and tables; and write and debug implementation code. The authors take responsibility for the final content, including all text, claims, code, and results.

ETHICS STATEMENT

This work investigates visual KV compression in hybrid vision-language models. We have not identified any new ethical concerns arising from this work.

REPRODUCIBILITY STATEMENT

Section 3 describes full-attention extra information extraction, prediction-conditioned semantic scoring, and visual KV selection in FA-ESI. Section 4.1 specifies the evaluated models, benchmarks, baselines, and key hyperparameters, including the visual KV retention ratio, score refresh interval, target layers, and number of prediction candidates. Section 4.5 reports the hardware used for the efficiency evaluation. We provide an anonymized implementation of FA-ESI, including inference configurations, a runnable example, and verification scripts, in the supplementary material.

REFERENCES

Liang Chen, Haozhe Zhao, Tianyu Liu, Shuai Bai, Junyang Lin, Chang Zhou, and Baobao Chang. An Image is Worth 1/2 Tokens After Layer 2: Plug-and-Play Inference Acceleration for Large Vision-Language Models. In *European Conference on Computer Vision*, 2024. URL https://arxiv.org/abs/2403.06764.

Ziming Cheng, Binrui Xu, Lisheng Gong, Zuhe Song, Tianshuo Zhou, Shiqi Zhong, Siyu Ren, Mingxiang Chen, Xiangchao Meng, Yuxin Zhang, Yanlin Li, Lei Ren, Wei Chen, Zhiyuan Huang, Mingjie Zhan, Xiaojie Wang, and Fangxiang Feng. Evaluating MLLMs with Multimodal Multi-image Reasoning Benchmark. *arXiv preprint arXiv:2506.04280*, 2025. URL https://arxiv.org/abs/2506.04280.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized Models and Efficient Algorithms Through Structured State Space Duality. In *International Conference on Machine Learning*, 2024. URL https://arxiv.org/abs/2405.21060.

Hang Du, Jiayang Zhang, Guoshun Nan, Wendi Deng, Zhenyan Chen, Chenyang Zhang, Wang Xiao, Shan Huang, Yuqi Pan, Tao Qi, and Sicong Leng. From Easy to Hard: The MIR Benchmark for Progressive Interleaved Multi-Image Reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025. URL https://arxiv.org/abs/2509.17040.

Xinyu Fang, Kangrui Mao, Haodong Duan, Xiangyu Zhao, Yining Li, Dahua Lin, and Kai Chen. MMBench-Video: A Long-Form Multi-Shot Benchmark for Holistic Video Understanding. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2406.14515.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, Ran He, and Xing Sun. Video-MME: The First-Ever Comprehensive Evaluation Benchmark of Multi-modal LLMs in Video Analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24108–24118, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Fu_Video-MME_The_First-Ever_Comprehensive_Evaluation_Benchmark_of_Multi-modal_LLMs_in_CVPR_2025_paper.html.

Albert Gu and Tri Dao. Mamba: Linear-Time Sequence Modeling with Selective State Spaces. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=tEYskw1VY2.

Kai Huang, Hao Zou, Bochen Wang, Ye Xi, Zhen Xie, and Hao Wang. AirCache: Activating Inter-modal Relevancy KV Cache Compression for Efficient Large Vision-Language Model Inference. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 23958–23967, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Huang_AirCache_Activating_Inter-modal_Relevancy_KV_Cache_Compression_for_Efficient_Large_ICCV_2025_paper.html.

Kimi Team, Yu Zhang, Zongyu Lin, Xingcheng Yao, Jiaxi Hu, Fanqing Meng, Chengyin Liu, Xin Men, Songlin Yang, Zhiyuan Li, Wentao Li, Enzhe Lu, Weizhou Liu, Yanru Chen, Weixin Xu, Longhui Yu, Yejie Wang, Yu Fan, Longguang Zhong, Enming Yuan, Dehao Zhang, Yizhi Zhang, T. Y. Liu, Haiming Wang, Shengjun Fang, Weiran He, Shaowei Liu, Yiwei Li, Jianlin Su, Jiezhong Qiu, Bo Pang, Junjie Yan, Zhejun Jiang, Weixiao Huang, Bohong Yin, Jiacheng You, Chu Wei, Zhengtao Wang, Chao Hong, Yutian Chen, Guanduo Chen, Yucheng Wang, Huabin Zheng, Feng Wang, Yibo Liu, Mengnan Dong, Zheng Zhang, Siyuan Pan, Wenhao Wu, Yuhao Wu, Longyu Guan, Jiawen Tao, Guohong Fu, Xinran Xu, Yuzhi Wang, Guokun Lai, Yuxin Wu, Xinyu Zhou, Zhilin Yang, and Yulun Du. Kimi Linear: An Expressive, Efficient Attention Architecture. *arXiv preprint arXiv:2510.26692*, 2025. URL https://arxiv.org/abs/2510.26692.

Kaixin Li, Ziyang Meng, Hongzhan Lin, Ziyang Luo, Yuchen Tian, Jing Ma, Zhiyong Huang, and Tat-Seng Chua. ScreenSpot-Pro: GUI Grounding for Professional High-Resolution Computer Use. *arXiv preprint arXiv:2504.07981*, 2025. URL https://arxiv.org/abs/2504.07981.

Kunchang Li, Yali Wang, Yinan He, Yizhuo Li, Yi Wang, Yi Liu, Zun Wang, Jilan Xu, Guo Chen, Ping Luo, Limin Wang, and Yu Qiao. MVBench: A Comprehensive Multi-modal Video Understanding Benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024a. URL https://arxiv.org/abs/2311.17005.

Wei Li, William Bishop, Alice Li, Chris Rawles, Folawiyo Campbell-Ajala, Divya Tyamagundlu, and Oriana Riva. On the Effects of Data Scale on UI Control Agents. In *Advances in Neural Information Processing Systems*, 2024b. URL https://arxiv.org/abs/2406.03679.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. MMBench: Is Your Multi-modal Model an All-around Player? In *European Conference on Computer Vision*, pp. 216–233, 2024. URL https://doi.org/10.1007/978-3-031-72658-3_13.

Shravan Nayak, Xiangru Jian, Kevin Qinghong Lin, Juan A. Rodriguez, Montek Kalsi, Rabiul Awal, Nicolas Chapados, M. Tamer Özsu, Aishwarya Agrawal, David Vazquez, Christopher Pal, Perouz Taslakian, Spandana Gella, and Sai Rajeswar. UI-Vision: A Desktop-centric GUI Benchmark for Visual Perception and Interaction. In *International Conference on Machine Learning*, 2025. URL https://arxiv.org/abs/2503.15661.

NVIDIA, Amala Sanjay Deshmukh, Kateryna Chumachenko, Tuomas Rintamaki, Matthieu Le, Tyler Poon, Danial Mohseni Taheri, Ilia Karmanov, Guilin Liu, Jarno Seppanen, Guo Chen, Karan Sapra, Zhiding Yu, Adi Renduchintala, Charles Wang, Peter Jin, Arushi Goel, Mike Ranzinger, Lukas Voegtle, Philipp Fischer, Timo Roman, Wei Ping, Boxin Wang, Zhuolin Yang, Nayeon Lee, Shaokun Zhang, Fuxiao Liu, Zhiqi Li, Di Zhang, Greg Heinrich, Hongxu Yin, Song Han, Pavlo Molchanov, Parth Mannan, Yao Xu, Jane Polak Scowcroft, Tom Balough, Subhashree Radhakrishnan, Paris Zhang, Sean Cha, Ratnesh Kumar, Zaid Pervaiz Bhat, Jian Zhang, Darragh Hanley, Pritam Biswas, Jesse Oliver, Kevin Vasques, Roger Waleffe, Duncan Riach, Oluwatobi Olabiyi, Ameya Sunil Mahabaleshwarkar, Bilal Kartal, Pritam Gundecha, Khanh Nguyen, Alexandre Milesi, Eugene Khvedchenia, Ran Zilberstein, Ofri Masad, Natan Bagrov, Nave Assaf, Tomer Asida, Daniel Afrimi, Amit Zuker, Netanel Haber, Zhiyu Cheng, Jingyu Xin, Di Wu, Nik Spirin, Maryam Moosaei, Roman Ageev, Vanshil Atul Shah, Yuting Wu, Daniel Korzekwa, Unnikrishnan Kizhakkemadam Sreekumar, Wanli Jiang, Padmavathy Subramanian, Alejandra Rico, Sandip Bhaskar, Saeid Motiian, Kedi Wu, Annie Surla, Chia-Chih Chen, Hayden Wolff, Matthew

Feinberg, Melissa Corpuz, Marek Wawrzos, Eileen Long, Aastha Jhunjhunwala, Paul Hendricks, Farzan Memarian, Benika Hall, Xin-Yu Wang, David Mosallanezhad, Soumye Singhal, Luis Vega, Katherine Cheung, Krzysztof Pawelec, Michael Evans, Katherine Luna, Jie Lou, Erick Galinkin, Akshay Hazare, Kaustubh Purandare, Ann Guan, Anna Warno, Chen Cui, Yoshi Suhara, Shibani Likhite, Seph Mard, Meredith Price, Laya Sleiman, Saori Kaji, Udi Karpas, Kari Briski, Joey Conway, Michael Lightstone, Jan Kautz, Mohammad Shoeybi, Mostofa Patwary, Jonathen Cohen, Oleksii Kuchaiev, Andrew Tao, and Bryan Catanzaro. NVIDIA Nemotron Nano V2 VL. *arXiv preprint arXiv:2511.03929*, 2025. URL https://arxiv.org/abs/2511.03929.

Ziqing Qiao, Yinuo Xu, Chaojun Xiao, Zhou Su, Zihan Zhou, Yingfa Chen, Xiaoyue Xu, Xu Han, and Zhiyuan Liu. Rethinking the Role of Efficient Attention in Hybrid Architectures. *arXiv preprint arXiv:2606.15378*, 2026. URL https://arxiv.org/abs/2606.15378.

Qwen Team. Qwen3.5-4B Model Card. Hugging Face, 2026. URL https://huggingface.co/Qwen/Qwen3.5-4B.

Imanol Schlag, Kazuki Irie, and Jürgen Schmidhuber. Linear Transformers Are Secretly Fast Weight Programmers. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 9355–9366. PMLR, 2021. URL https://proceedings.mlr.press/v139/schlag21a.html.

Yuzhang Shang, Mu Cai, Bingxin Xu, Yong Jae Lee, and Yan Yan. LLaVA-PruMerge: Adaptive Token Reduction for Efficient Large Multimodal Models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22857–22867, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Shang_LLaVA-PruMerge_Adaptive_Token_Reduction_for_Efficient_Large_Multimodal_Models_ICCV_2025_paper.html.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA Models That Can Read. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. URL https://arxiv.org/abs/1904.08920.

Dingjie Song, Shunian Chen, Guiming Hardy Chen, Fei Yu, Xiang Wan, and Benyou Wang. MileBench: Benchmarking MLLMs in Long Context. In *Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=Uhwze2LEwq.

Zhongwei Wan, Ziang Wu, Che Liu, Jinfa Huang, Zhihong Zhu, Peng Jin, Longyue Wang, and Li Yuan. LOOK-M: Look-Once Optimization in KV Cache for Efficient Multimodal Long-Context Inference. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 4065–4078, 2024. URL https://aclanthology.org/2024.findings-emnlp.235/.

Fei Wang, Xingyu Fu, James Y. Huang, Zekun Li, Qin Liu, Xiaogeng Liu, Mingyu Derek Ma, Nan Xu, Wenxuan Zhou, Kai Zhang, Tianyi Lorena Yan, Wenjie Jacky Mo, Hsiang-Hui Liu, Pan Lu, Chunyuan Li, Chaowei Xiao, Kai-Wei Chang, Dan Roth, Sheng Zhang, Hoifung Poon, and Muhao Chen. MuirBench: A Comprehensive Benchmark for Robust Multi-image Understanding. In *International Conference on Learning Representations*, 2025a. URL https://arxiv.org/abs/2406.09411.

Xuan Wang, Yilin Liu, Fangxiang Feng, Caixia Yuan, Huixing Jiang, and Xiaojie Wang. Semantic Impact–Driven Visual Scheduling in Vision-Language Models. In *International Conference on Machine Learning*, 2026. URL https://icml.cc/virtual/2026/poster/61633.

Xuehui Wang, Zhenyu Wu, JingJing Xie, Zichen Ding, Bowen Yang, Zehao Li, Zhaoyang Liu, Qingyun Li, Xuan Dong, Zhe Chen, Weiyun Wang, Xiangyu Zhao, Jixuan Chen, Haodong Duan, Tianbao Xie, Chenyu Yang, Shiqian Su, Yue Yu, Yuan Huang, Yiqian Liu, Xiao Zhang, Yanting Zhang, Xiangyu Yue, Weijie Su, Xizhou Zhu, Wei Shen, Jifeng Dai, and Wenhai Wang. MMBench-GUI: Hierarchical Multi-Platform Evaluation Framework for GUI Agents. *arXiv preprint arXiv:2507.19478*, 2025b. URL https://arxiv.org/abs/2507.19478.

Haoning Wu, Dongxu Li, Bei Chen, and Junnan Li. LongVideoBench: A Benchmark for Long-context Interleaved Video-Language Understanding. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2407.15754.

Hengbo Xu, Shengjie Jin, Yanbiao Ma, and Zhiwu Lu. VisionPulse: Dynamic Visual Sparsity for Efficient Multimodal Reasoning. In *International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2605.31457.

Senqiao Yang, Yukang Chen, Zhuotao Tian, Chengyao Wang, Jingyao Li, Bei Yu, and Jiaya Jia. VisionZip: Longer is Better but Not Necessary in Vision Language Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025a. URL https://arxiv.org/abs/2412.04467.

Songlin Yang, Bailin Wang, Yu Zhang, Yikang Shen, and Yoon Kim. Parallelizing Linear Transformers with the Delta Rule over Sequence Length. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2406.06484.

Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated Delta Networks: Improving Mamba2 with Delta Rule. In *International Conference on Learning Representations*, 2025b. URL https://arxiv.org/abs/2412.06464.

Qizhe Zhang, Aosong Cheng, Ming Lu, Renrui Zhang, Zhiyong Zhuo, Jiajun Cao, Shaobo Guo, Qi She, and Shanghang Zhang. Beyond Text-Visual Attention: Exploiting Visual Cues for Effective Token Pruning in VLMs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20857–20867, 2025a. URL https://openaccess.thecvf.com/content/ICCV2025/html/Zhang_Beyond_Text-Visual_Attention_Exploiting_Visual_Cues_for_Effective_Token_Pruning_ICCV_2025_paper.html.

Yuan Zhang, Chun-Kai Fan, Junpeng Ma, Wenzhao Zheng, Tao Huang, Kuan Cheng, Denis Gudovskiy, Tomoyuki Okuno, Yohei Nakata, Kurt Keutzer, and Shanghang Zhang. SparseVLM: Visual Token Sparsification for Efficient Vision-Language Model Inference. In *International Conference on Machine Learning*, 2025b. URL https://arxiv.org/abs/2410.04417.

A Evaluation benchmarks

A.1 Image understanding and reasoning

MMBench. MMBench (Liu et al., 2024) assesses multimodal perception and reasoning through multiple-choice questions covering a range of visual abilities.

TextVQA. TextVQA (Singh et al., 2019) evaluates question answering that requires reading text in images and reasoning about it in the visual context.

MileBench. MileBench (Song et al., 2024) evaluates multimodal long-context understanding through diagnostic and realistic tasks involving multiple images and text.

MuirBench. MuirBench (Wang et al., 2025a) evaluates multi-image understanding across diverse tasks and image relationships, including scene understanding, ordering, and multiview relations.

MMRB. MMRB (Cheng et al., 2025) evaluates structured reasoning across multiple images, covering spatial, temporal, and semantic relationships.

MIRBench. MIRBench (Du et al., 2025) evaluates joint reasoning over multiple images and interleaved text, requiring models to associate image regions with text and connect information across images.

A.2 VIDEO UNDERSTANDING

Video-MME. Video-MME (Fu et al., 2025) evaluates video understanding across diverse domains and short, medium, and long video durations.

LongVideoBench. LongVideoBench (Wu et al., 2024) evaluates long-video understanding through multiple-choice questions that require retrieving and reasoning about relevant details in video-language contexts.

MMBench-Video. MMBench-Video (Fang et al., 2024) uses free-form questions about long-form, multi-shot videos to assess holistic understanding and temporal reasoning.

MVBench. MVBench (Li et al., 2024a) comprises multiple-choice video tasks that assess temporal understanding beyond what can be inferred from a single frame.

A.3 GUI-AGENT TASKS

MMBench-GUI. MMBench-GUI (Wang et al., 2025b) evaluates GUI agents across desktop, mobile, and web platforms, covering content understanding, element grounding, task automation, and task collaboration.

ScreenSpot-Pro. ScreenSpot-Pro (Li et al., 2025) evaluates GUI grounding in high-resolution screenshots of professional applications, where targets can be small and interfaces visually complex.

UI-Vision. UI-Vision (Nayak et al., 2025) evaluates desktop GUI perception and interaction through element grounding, layout grounding, and action prediction.

AndroidControl. AndroidControl (Li et al., 2024b) provides demonstrations of everyday tasks in Android applications, with both high-level task instructions and low-level step instructions for evaluating UI control agents.

B ADDITIONAL RESULTS

B.1 EFFECT OF REMOVING VISUAL KV ENTRIES

Removing all visual KV entries eliminates direct access to historical visual tokens through full attention, while visual information encoded in the recurrent GDN states remains available. Table 5 shows that performance remains close to the full-cache baseline on some benchmarks but drops sharply on others, particularly GUI grounding. By retaining approximately 5% of visual KV entries, FA-ESI maintains approximately 99% of the full-cache performance on average across the 14 main benchmarks (Table 1). Together, these results highlight the value of preserving a small, task-relevant visual KV subset to complement recurrent-state representations under high compression.

Table 5: Performance with and without visual KV entries. "Full KV" retains the complete visual KV cache; "No visual KV" removes all visual KV entries from full-attention layers while preserving the recurrent GDN states.

	Qwen3.5-2B		Qwen3.5-4B	
Benchmark	Full KV	No visual KV	Full KV	No visual KV
MMBench	71.749	38.390	81.037	2.786
TextVQA	78.960	47.686	83.122	55.358
MileBench	53.349	47.391	48.552	41.395
MuirBench	42.192	41.923	54.115	53.769
Video-MME	50.056	49.869	56.446	56.468
LongVideoBench	46.597	46.597	51.309	51.309
MVBench	55.263	55.263	61.158	61.158
MMBench-GUI	74.572	5.181	69.555	14.825
ScreenSpot-Pro	44.529	0.443	59.140	0.633
MMRB	59.550	46.560	–	–

C VISUAL KV SELECTION DURING DECODING

We visualize the image regions corresponding to visual KV entries retained by FA-ESI at different decoding steps (Figures 6–8). Each example includes the input images, the generated response, and the corresponding selection masks. Colored text links the visualizations to the associated passages in the response.

Prompt: Is the dog or is the human in <image> bigger? Assume the car in <image> is the same size as in <image>.
A. The human is bigger.
B. The dog is bigger.
C. The dog is as big as the human.



Answer: To determine which is bigger, we need to compare the sizes of the subjects in the images based on the provided assumption.
- **Assumption**: The car in the second image is the same size as the car in the third image.
- **Comparison**: We can compare the apparent size of the dog in the second image to the human in the third image relative to their respective cars.
  - In the second image, the dog appears to be roughly the same height as the car's roofline.
  - In the third image, the human appears to be significantly shorter than the car's roofline.
- **Conclusion**: Since the cars are the same size, the dog must be taller than the human. Therefore, the dog is bigger.
Your option: B. The dog is bigger.
Reason: By assuming the cars are the same size, we can compare the subjects' heights relative to the cars. The dog is shown to be as tall as the car, while the human is much shorter than the car, indicating the dog is bigger.

roofline. - In the third image

the human appears to be significantly shorter than the car's

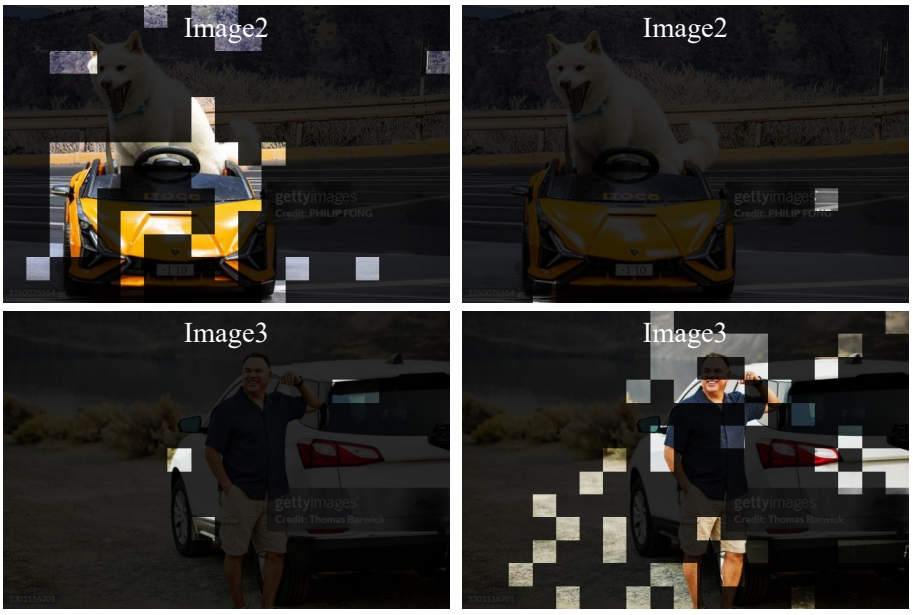


Figure 6: Visual KV selection during decoding for a cross-image size comparison.

Prompt: Assuming the following images <image>, <image>, <image>, <image>, <image>, <image> are taken from a panoramic view of a long table, can you determine the spatial relationship between Stuffed bear with a yellow bow and bottle of Pepsi?
A. Stuffed bear with a yellow bow is to the left of bottle of Pepsi
B. Stuffed bear with a yellow bow is to the right of bottle of Pepsi
C. Unable to determine due to missing intermediate positioning details
D. Object Stuffed bear with a yellow bow or bottle of Pepsi does not exist

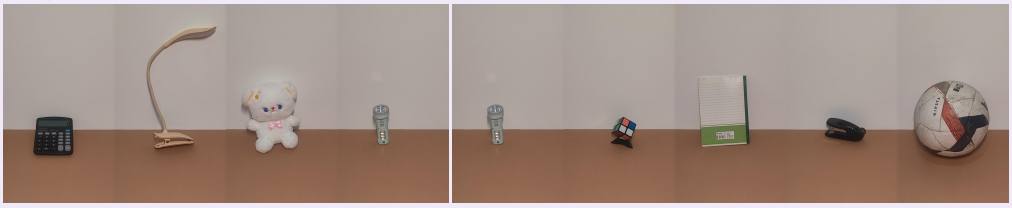


Answer: To determine the spatial relationship between the "Stuffed bear with a yellow bow" and the "bottle of Pepsi", we need to locate both objects within the provided panoramic images.
- The "bottle of Pepsi" is clearly visible in the first image, positioned on the far left side of the table.
- The "Stuffed bear with a yellow bow" is visible in the sixth image, positioned on the far right side of the table.
Since the images are taken from a panoramic view of a long table, we can infer the relative positions of objects across the different images. The bottle of Pepsi is on the left end of the table, and the stuffed bear is on the right end. Therefore, the stuffed bear is to the right of the bottle of Pepsi.
Your option: B
Reason: The bottle of Pepsi is located on the far left of the table (Image 1), while the stuffed bear with a yellow bow is located on the far right of the table (Image 6). In a panoramic view of a single long table, an object on the right is spatially to the right of an object on the left.



Figure 7: Visual KV selection during decoding for the spatial relationship between a stuffed bear and a bottle of Pepsi.

Prompt: Assuming the following images <image>, <image> are taken from a panoramic view of a long table, can you determine the spatial relationship between calculator and Soccer ball?
A. calculator is to the left of Soccer ball
B. calculator is to the right of Soccer ball
C. Unable to determine due to missing intermediate positioning details
D. Object calculator or Soccer ball does not exist



Answer: Based on the provided panoramic images, we can determine the spatial relationship between the calculator and the soccer ball by observing their positions from left to right.
- In the first image, the calculator is located on the far left side of the table.
- In the second image, the Soccer ball is located on the far right side of the table.
Since the calculator is on the left and the soccer ball is on the right, the calculator is to the left of the soccer ball.
Your option: A
Reason: The calculator is positioned on the left side of the table in the first image, while the soccer ball is on the right side in the second image. Therefore, the calculator is to the left of the soccer ball.

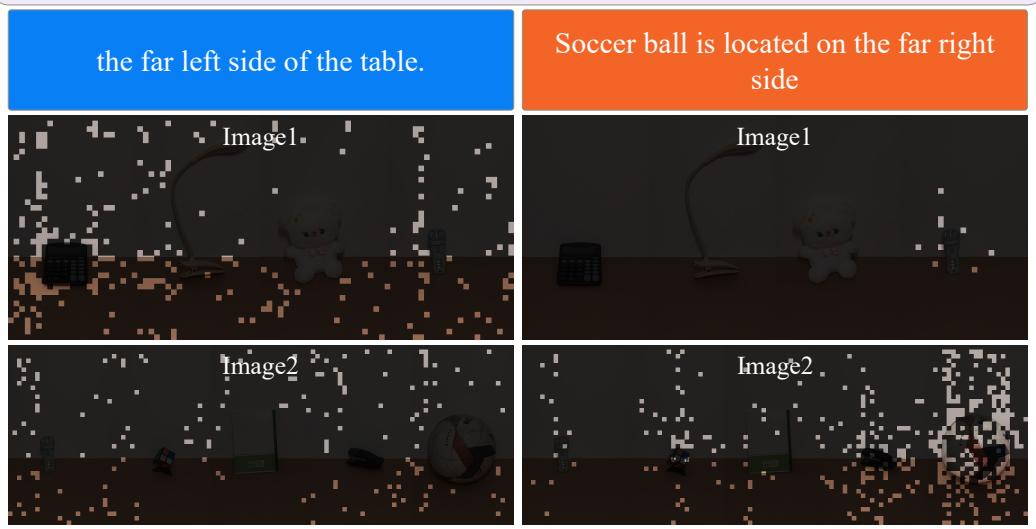


Figure 8: Visual KV selection during decoding for the spatial relationship between a calculator and a soccer ball.