GEOKV: GEOMETRY-AWARE KV CACHE COMPRESSION VIA GLOBAL STRUCTURAL CONTRIBUTION AND KEY RECTIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-Language Models (VLMs) have demonstrated remarkable capabilities, yet the rapidly growing memory consumption of Key-Value (KV) caches poses a major challenge for efficient inference. Existing KV cache compression methods predominantly characterize token importance from a query-dependent functional perspective, such as attention or local relevance, while overlooking the global structural role of cached representations. Consequently, tokens with low current-query relevance may still be important for preserving underrepresented directions and the global geometry of the cached-key space. To address this limitation, we propose **GeoKV**, a global-structure-aware framework for KV cache compression. Specifically, we introduce **Global Structural Contribution (GSC)**, a query-independent criterion derived from the marginal gain of a regularized log-determinant objective, and perform greedy selection to preserve structurally valuable tokens. We further introduce a lightweight geometry-aware key rectification module based on a low-rank feature-space transformation to mitigate compression-induced geometric distortion. Extensive experiments on VLMs demonstrate that GeoKV effectively reduces KV cache size while maintaining model performance. Comprehensive analyses further show that structural contribution captures complementary information beyond query-dependent functional relevance.

1 INTRODUCTION

Large Language Models (LLMs) and Vision-Language Models (VLMs) have demonstrated remarkable capabilities across a broad range of applications. However, deploying these models in long-context scenarios remains challenging due to the rapidly increasing memory consumption of Key-Value (KV) caches (Behnam et al., 2025; Qin et al., 2025; Li et al., 2025). During autoregressive inference, KV caches store historical contextual representations to avoid redundant computation, but their memory footprint grows linearly with sequence length, becoming one of the primary bottlenecks for efficient deployment (Lee et al., 2025).

KV cache compression alleviates this memory overhead by removing or compressing states that have limited influence on model computation (Cai et al., 2024; Shutova et al., 2025). Existing approaches include token eviction, token merging, quantization, and low-rank approximation (Li et al., 2024; Shutova et al., 2025). A central challenge underlying these methods is therefore to determine which cached states can be discarded with minimal information loss (Li et al., 2024; Cai et al., 2024). Most existing methods characterize token importance from a local or query-dependent perspective (Tang et al., 2024; Devoto et al., 2024). For example, attention-based methods prioritize tokens receiving large attention weights from the current query (Zhang et al., 2023; Li et al., 2024), query-aware approaches select states according to their immediate relevance (Huang et al., 2025), and similarity-based methods identify locally redundant representations through pairwise relationships (Jha et al., 2025). These methods primarily focus on identifying the most useful cached states for the current query or computation by evaluating the impact of an individual cached state on the current attention

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.

computation, which we refer to as **functional relevance** (Zhang et al., 2023; Li et al., 2024; Tu et al., 2025; Huang et al., 2025).

However, functional relevance (Zhang et al., 2023; Li et al., 2024; Tu et al., 2025; Huang et al., 2025) does not fully characterize the role of a cached state within the entire cache. The cached keys jointly define a structured space whose global geometry cannot be fully characterized by the importance of individual tokens (Wang et al., 2024a). While current-query attention measures how strongly a cached key interacts with the present query (Zhang et al., 2023; Li et al., 2024), the cached-key space also contains query-independent information about how its representations are collectively organized (Devoto et al., 2024). We refer to this notion as **structural contribution**, namely the extent to which a cached state contributes to preserving the global geometry of the cached-key space. These two notions therefore capture different aspects of cache importance: functional relevance asks whether a state is useful *for the current query*, whereas structural contribution asks whether it is important *for maintaining the global geometry of the cached-key space*. Consequently, a state with low current-query relevance may still preserve an important geometric direction, while a highly attended state may be redundant with other cached states (Cai et al., 2026).

We empirically examine this distinction on real multimodal inputs. Using LLaVA-OneVision-7B (Liu et al., 2023) as our primary testbed, Fig. 1(a) shows that current-query attention and a leave-one-out (LOO) diagnostic of structural sensitivity exhibit substantial misalignment across the evaluated images. The mean Spearman correlation is -0.232 ± 0.135, with 95% of the samples showing negative correlation.

Table 1: Cross-model Spearman correlation between attention and LOO sensitivity on MSCOCO.

Model	Corr.	Std.	Neg. Corr. (%)
InternVL2-8B	-0.18	0.12	88.0
Qwen2-VL-7B	-0.21	0.14	91.0

Importantly, this phenomenon is not an artifact of a specific architecture. As detailed in Table 1, this severe misalignment consistently generalizes VLMs, including InternVL2-8B (Chen et al., 2024b) and Qwen2-VL-7B (Wang et al., 2024b), where over 88% of samples exhibit negative correlations. This pervasive divergence indicates that query-dependent functional relevance and structural sensitivity are inherently decoupled. This discrepancy also affects the behavior of the retained cache beyond the current query (Tang et al., 2025). As shown in Fig. 1(b), we measure the functional attention change as the query moves away from the current position. The cache selected by current-query attention exhibits a distinct decay pattern from the structurally selected cache, revealing that query-dependent relevance does not necessarily preserve the same functional behavior for future queries. Finally, Fig. 1(c) jointly compares geometric distortion and future-query functional drift. Top-LOO and Top-Attn exhibit substantially different trade-offs between preserving the global geometry of the cached-key space and maintaining future-query functionality. Together, these cross-model observations indicate that query-dependent functional relevance and query-independent structural contribution capture complementary properties of the cached-key space.

In the end, we propose **GeoKV**, a geometry-aware KV cache compression framework that explicitly models the global geometry of the cached-key space. GeoKV introduces **Global Structural Contribution (GSC)**, a query-independent selection criterion that quantifies the conditional marginal contribution of each cached key to the global geometry of the selected cache. By accounting for the geometry already covered by retained states, GSC favors complementary representations while suppressing structurally redundant tokens. Since token selection inevitably introduces **compression-induced geometric distortion**, GeoKV further employs a lightweight **Geometry-aware Key Rectification** mechanism based on a compact low-rank feature-space transformation, calibrated offline to compensate for systematic distortion during inference. Our contributions are summarized as follows:

(1) We propose GeoKV, a geometry-aware KV cache compression framework that combines global-structure-aware KV selection with lightweight geometric rectification, offering a query-independent complement to existing query-dependent token selection criteria.

(2) We introduce **Global Structural Contribution (GSC)**, a query-independent criterion that quantifies the conditional marginal contribution of each KV state to the global geometry of the cached-key space, enabling geometry-aware KV cache selection.

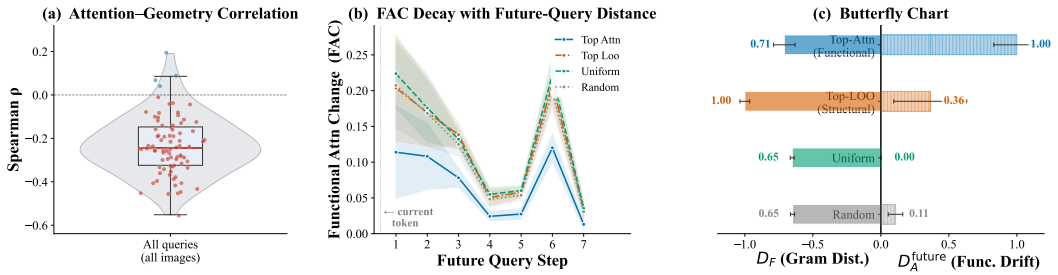


Figure 1: Motivation for geometry-aware KV cache compression. (a) Current-query attention and leave-one-out (LOO) structural sensitivity show substantial misalignment across multimodal inputs. (b) Their selected caches exhibit distinct functional attention change patterns as the query moves into the future, revealing limited generality of query-dependent relevance. (c) The butterfly chart compares geometric distortion D_F with future-query functional drift D_F^{future}, highlighting the different trade-offs between geometric preservation and future-query functionality. These observations motivate a query-independent criterion that captures the global structural contribution of cached keys.

(3) We develop a **Geometry-aware Key Rectification** mechanism based on a compact low-rank transformation, which compensates for compression-induced structural distortion with lightweight runtime overhead.

2 Related Work

2.1 Token Eviction for KV Cache Compression

To reduce autoregressive memory costs, KV compression methods retain tokens using attention sinks (Xiao et al., 2024), heavy-hitters (Zhang et al., 2023), or adaptive relevance (Li et al., 2024). Recently, VLM-specific approaches have similarly pruned visual tokens via modality-specific attention patterns (Chen et al., 2024a; Tu et al., 2025; Yang et al., 2025; Huang et al., 2025; Yang et al., 2026). Despite their efficiency, these methods universally characterize token importance through local or query-dependent functional signals (Zhang et al., 2023; Li et al., 2024; Qin et al., 2025; Huang et al., 2025). By prioritizing immediate utility for the current query, they inherently neglect a token's global structural contribution to the cached representation space.

2.2 Low-Rank KV Cache Approximation

Representation-level methods compress KV caches by reducing feature dimensions via low-rank projection, offline calibration, or quantization (Chang et al., 2025; Yan et al., 2025; Kang et al., 2024). However, these approaches replace the cache with continuous approximations rather than selecting structurally critical discrete tokens. In contrast, GeoKV retains a discrete token subset based on its marginal contribution to the global cached-key geometry. Our low-rank key rectification is thus highly complementary: instead of transforming the entire cache, it applies a lightweight projection exclusively to the retained discrete tokens to mitigate eviction-induced geometric distortion.

3 Method

3.1 Problem Formulation

Let $K^{(h)} \in \mathbb{R}^{N\times d}$ denote the post-RoPE key cache of KV head h, and let $V^{(h)} \in \mathbb{R}^{N\times d_v}$ denote the corresponding value cache. Here N is the number of cached tokens and d is the key dimension. All geometric quantities below are computed on these post-RoPE keys, i.e., the representations consumed by attention. Given a target retention length $N' < N$, KV cache compression seeks a subset $\mathcal{S} \subseteq \{1,\dots,N\}$ with $|\mathcal{S}| = N'$ such that the resulting attention computation preserves as much useful information as possible.

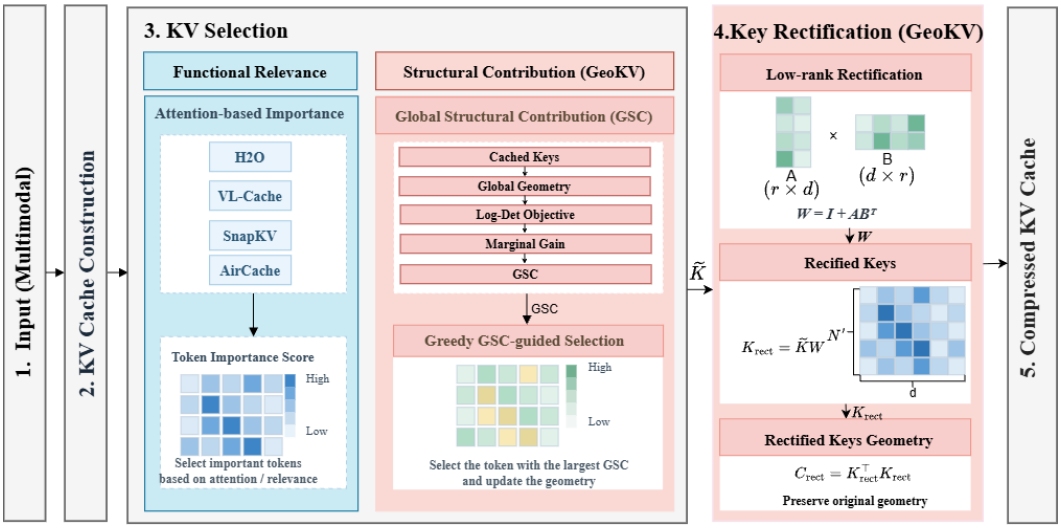


Figure 2: Overview of GeoKV. The blue branch represents existing *functional-relevance*-based KV selection methods, which rely on query-dependent attention or relevance signals. The pink branch represents our GeoKV, which selects tokens according to *global structural contribution* (GSC) and subsequently applies low-rank key rectification to preserve the global geometry of the cached-key space.

To construct $\mathcal{S}$, GeoKV explicitly preserves the global geometry of the cached keys through two complementary stages. First, we select tokens based on their *global structural contribution*, quantified by the marginal gain of a regularized log-determinant objective. Second, we apply a lightweight low-rank feature-space transformation to rectify the geometric distortion induced by token removal.

3.2 Global Structural Contribution

3.2.1 Feature-space Geometry

To explicitly capture the global structure of the cached keys, let $K_{\mathcal{S}}^{(h)} = \left[k_i^{(h)}\right]_{i\in\mathcal{S}} \in \mathbb{R}^{|\mathcal{S}|\times d}$ denote the keys indexed by $\mathcal{S}$ for head h. We summarize their global feature-space geometry using the uncentered second-moment matrix

$$C_{\mathcal{S}}^{(h)} = K_{\mathcal{S}}^{(h)\top} K_{\mathcal{S}}^{(h)} \in \mathbb{R}^{d\times d}. \tag{1}$$

This representation characterizes how the selected keys distribute energy across feature directions without explicitly constructing the $N \times N$ token-level Gram matrix. More importantly, adding a candidate key $k_i^{(h)}$ induces a rank-one update:

$$C_{\mathcal{S}\cup\{i\}}^{(h)} = C_{\mathcal{S}}^{(h)} + k_i^{(h)} k_i^{(h)\top}. \tag{2}$$

This property enables us to quantify the structural contribution of a candidate through an analytically tractable marginal gain.

3.2.2 Regularized Log-determinant Objective

To measure the geometric coverage of a selected subset, we define

$$\mathcal{F}(\mathcal{S}) = \frac{1}{H_{\text{KV}}} \sum_{h=1}^{H_{\text{KV}}} \log\det\left(C_{\mathcal{S}}^{(h)} + \lambda I\right), \tag{3}$$

where H_{KV} denotes the number of KV heads and $\lambda > 0$ is a ridge parameter.

The log-determinant is particularly suitable for measuring global representation coverage. Let $\{\gamma_j\}_{j=1}^{d}$ denote the eigenvalues of $C_{\mathcal{S}}^{(h)}$. Then

$$\log\det\left(C_{\mathcal{S}}^{(h)}+\lambda I\right)=\sum_{j=1}^{d}\log(\gamma_j+\lambda). \tag{4}$$

Therefore, $\mathcal{F}(\mathcal{S})$ jointly accounts for the distribution of energy across feature directions rather than evaluating candidates only through pairwise similarity. The ridge term also makes the objective well-defined when the selected keys span only a low-dimensional subspace.

An important consequence is that the marginal gain of i is weighted by the inverse matrix $\left(C_{\mathcal{S}}^{(h)}+\lambda I\right)^{-1}$. Directions that are already strongly represented in $\mathcal{S}$ receive smaller marginal gains, whereas directions that are weakly represented receive larger gains. Thus, the objective naturally favors candidates that provide complementary feature-space coverage while suppressing structurally redundant candidates.

3.2.3 Marginal Structural Contribution

For $i\notin\mathcal{S}$, the matrix determinant lemma gives

$$\mathcal{F}(\mathcal{S}\cup\{i\})-\mathcal{F}(\mathcal{S})=\frac{1}{H_{\mathrm{KV}}}\sum_{h=1}^{H_{\mathrm{KV}}}\log\left(1+k_i^{(h)\top}\left(C_{\mathcal{S}}^{(h)}+\lambda I\right)^{-1}k_i^{(h)}\right). \tag{5}$$

We define this marginal gain as the **Global Structural Contribution (GSC)**:

$$\mathrm{GSC}_i(\mathcal{S})=\frac{1}{H_{\mathrm{KV}}}\sum_{h=1}^{H_{\mathrm{KV}}}\log\left(1+k_i^{(h)\top}M_{\mathcal{S}}^{(h)}k_i^{(h)}\right), \tag{6}$$

where

$$M_{\mathcal{S}}^{(h)}=\left(C_{\mathcal{S}}^{(h)}+\lambda I\right)^{-1}. \tag{7}$$

$\mathrm{GSC}_i(\mathcal{S})$ measures the *conditional* structural contribution of i. A candidate aligned with directions already well represented by $\mathcal{S}$ produces a small marginal gain, whereas a candidate contributing an underrepresented direction produces a larger gain. Importantly, $\mathrm{GSC}_i(\mathcal{S})$ depends only on the cached-key geometry and is independent of the current attention query. It therefore provides a query-independent criterion for evaluating the structural role of cached states.

3.3 GSC-guided KV Selection

Given N', GeoKV greedily constructs a shared token subset across all H_{KV} heads. Let $\mathcal{S}_t$ denote the selected subset after t iterations. For each h, we maintain

$$M_t^{(h)}=\left(C_{\mathcal{S}_t}^{(h)}+\lambda I\right)^{-1}. \tag{8}$$

For every $i\notin\mathcal{S}_t$, its conditional structural contribution is computed as

$$\mathrm{GSC}_{i,t}=\frac{1}{H_{\mathrm{KV}}}\sum_{h=1}^{H_{\mathrm{KV}}}\log\left(1+k_i^{(h)\top}M_t^{(h)}k_i^{(h)}\right). \tag{9}$$

We then select the candidate with the largest aggregated marginal gain:

$$i^*=\arg\max_{i\notin\mathcal{S}_t}\mathrm{GSC}_{i,t},\qquad \mathcal{S}_{t+1}=\mathcal{S}_t\cup\{i^*\}. \tag{10}$$

Since $f(\mathcal{S})=\mathcal{F}(\mathcal{S})-\mathcal{F}(\emptyset)$ is monotone submodular, the greedy procedure achieves the classical $(1-1/e)$ approximation guarantee under $|\mathcal{S}|\le N'$ (Nemhauser et al., 1978); see Appendix B. The procedure terminates when $|\mathcal{S}_t|=N'$. For Grouped-Query Attention (GQA), we use the same token index set across all KV heads. Each h independently evaluates the geometric contribution of i, and the resulting scores are averaged to obtain $\mathrm{GSC}_{i,t}$. This preserves the key-value correspondence across H_{KV} heads and avoids inconsistent token subsets. After selection, the compressed key and value caches are

$$\widetilde{K}^{(h)}=K_{\mathcal{S}_{N'}}^{(h)},\qquad \widetilde{V}^{(h)}=V_{\mathcal{S}_{N'}}^{(h)}. \tag{11}$$

3.3.1 EFFICIENT MATRIX UPDATES

Directly recomputing $M_t^{(h)}$ after every greedy selection would introduce unnecessary computational cost. Since adding the selected key produces a rank-one update,

$$C_{\mathcal{S}_{t+1}}^{(h)} = C_{\mathcal{S}_t}^{(h)} + k_{i^*}^{(h)} k_{i^*}^{(h)\top}, \tag{12}$$

the inverse can be updated exactly using the Sherman–Morrison formula:

$$M_{t+1}^{(h)} = M_t^{(h)} - \frac{M_t^{(h)} k_{i^*}^{(h)} k_{i^*}^{(h)\top} M_t^{(h)}}{1 + k_{i^*}^{(h)\top} M_t^{(h)} k_{i^*}^{(h)}}. \tag{13}$$

Consequently,

$$M_{t+1}^{(h)} = \left(C_{\mathcal{S}_{t+1}}^{(h)} + \lambda I \right)^{-1} \tag{14}$$

up to numerical precision, without requiring a new matrix inversion at each iteration. The quadratic forms in Eq. 9 can be evaluated in batched form across the remaining tokens. Across N' iterations, evaluating Eq. 9 requires $O\left(H_{\mathrm{KV}} N N' d^2\right)$ arithmetic operations in the straightforward batched implementation, while the Sherman–Morrison (Jiao, 2025) updates require an additional $O\left(H_{\mathrm{KV}} N' d^2\right)$. Importantly, GeoKV operates on compact $d \times d$ feature-space matrices and never constructs the $N \times N$ token-level Gram matrix. The practical selection overhead is evaluated experimentally in Sec. 4.4.

3.4 GEOMETRY-AWARE KEY RECTIFICATION

3.4.1 COMPRESSION-INDUCED GEOMETRIC DISTORTION

Although GSC selects tokens according to their global structural contribution, reducing the cache from N to N' inevitably changes the geometry of the retained representation set. To quantify this distortion, for layer l and KV head h, let

$$C^{(l,h)} = K^{(l,h)\top} K^{(l,h)}, \qquad \widetilde{C}^{(l,h)} = \widetilde{K}^{(l,h)\top} \widetilde{K}^{(l,h)} \tag{15}$$

denote the second-moment matrices of the original and compressed caches, respectively. We define the normalized feature-space Frobenius distortion (Bhattarai et al., 2025) as

$$D_F = \frac{1}{H_{\mathrm{KV}}} \sum_{h=1}^{H_{\mathrm{KV}}} \frac{\left\| \widetilde{C}^{(l,h)} - C^{(l,h)} \right\|_F}{\left\| C^{(l,h)} \right\|_F}. \tag{16}$$

Lower D_F indicates that the compressed cache better preserves the global second-order geometry of the original key space, where this geometry is given by the overall distribution of representation energy across feature directions. Unless otherwise specified, D_F is measured on post-RoPE key representations.

3.4.2 LOW-RANK GEOMETRY-AWARE RECTIFICATION

To reduce the geometric distortion introduced by token removal, GeoKV learns a feature-space transformation that maps the compressed key geometry toward the original geometry. For layer l and KV head h, we seek a transformation $W_{\mathrm{rect}}^{(l,h)}$ by minimizing

$$\min_{W_{\mathrm{rect}}^{(l,h)}} \left\| W_{\mathrm{rect}}^{(l,h)\top} \widetilde{C}^{(l,h)} W_{\mathrm{rect}}^{(l,h)} - C^{(l,h)} \right\|_F^2. \tag{17}$$

A dense $d \times d$ transformation would introduce substantial additional parameters and computation. We therefore parameterize $W_{\mathrm{rect}}^{(l,h)}$ as an identity mapping plus a low-rank residual:

$$W_{\mathrm{rect}}^{(l,h)} = I + A^{(l,h)} B^{(l,h)\top}, \qquad A^{(l,h)}, B^{(l,h)} \in \mathbb{R}^{d \times r}, \quad r \ll d. \tag{18}$$

The identity component ensures that the learned transformation starts from the unmodified compressed representation, while the low-rank residual captures systematic geometric deviations introduced by compression. The rectified keys are computed as

$$K_{\mathrm{rect}}^{(l,h)} = \widetilde{K}^{(l,h)} W_{\mathrm{rect}}^{(l,h)} = \widetilde{K}^{(l,h)} + \left(\widetilde{K}^{(l,h)} A^{(l,h)} \right) B^{(l,h)\top}. \tag{19}$$

Thus, the dense feature transformation is never explicitly materialized. The additional computation is $O(N' d r)$, which is substantially smaller than the $O(N' d^2)$ cost of applying a dense transformation when $r \ll d$.

3.4.3 OFFLINE CALIBRATION

The rectifier is calibrated offline and remains fixed during inference. Let $\mathcal{D}_{\text{cal}} = \{X_1, \ldots, X_M\}$ denote a calibration set containing M representative inputs. For each $X_x \in \mathcal{D}_{\text{cal}}$, we first obtain the original KV cache and then apply the same GSC-guided selection procedure to construct the compressed cache. The resulting paired matrices $\left(C_x^{(l,h)}, \widetilde{C}_x^{(l,h)}\right)$ provide supervision for the rectifier. The layer- and head-specific parameters are optimized offline according to

$$\min_{\{A^{(l,h)}, B^{(l,h)}\}} \sum_{x \in \mathcal{D}_{\text{cal}}} \sum_{l,h} \left\| W_{\text{rect}}^{(l,h)\top} \widetilde{C}_x^{(l,h)} W_{\text{rect}}^{(l,h)} - C_x^{(l,h)} \right\|_F^2, \tag{20}$$

where

$$W_{\text{rect}}^{(l,h)} = I + A^{(l,h)} B^{(l,h)\top}. \tag{21}$$

After calibration, $A^{(l,h)}, B^{(l,h)}$ are frozen and reused for subsequent inputs. Therefore, no per-input optimization or gradient-based adaptation is required during inference. The number of additional parameters introduced by the rectifier is

$$P_{\text{rect}} = 2LH_{\text{KV}}dr, \tag{22}$$

where L is the number of transformer layers. Since $r \ll d$, the additional parameter and memory overhead remains small relative to the backbone. The empirical effects of r and M are evaluated in Sec. 4.4 and the Appendix D. The rectifier operates only in the feature space and therefore preserves the selected token indices, sequence ordering, and causal structure. Applied to post-RoPE keys, it modifies the representation geometry without altering the retained token positions.

4 EXPERIMENT

4.1 EXPERIMENTAL SETUP

We evaluate GeoKV on ChatQA (Liu et al., 2024), InfoVQA (Mathew et al., 2022), DocVQA (Mathew et al., 2021), and TextVQA (Singh et al., 2019) using representative models from the InternVL2 (Chen et al., 2024b), LLaVA (Liu et al., 2023), and Qwen2-VL (Wang et al., 2024b) families. KV caches are compressed at a fixed retention ratio $\rho = N'/N$, with the main comparison conducted at $\rho = 0.20$ under matched retention budgets. Baselines include representative relevance-based methods (H2O (Zhang et al., 2023), SnapKV (Li et al., 2024), VL-Cache (Tu et al., 2025), AirCache (Huang et al., 2025)) and sampling controls, where Random uniformly samples N' tokens at random and Uniform selects N' tokens at evenly spaced positions, with Full Cache as the uncompressed reference. All methods follow the same inference protocol, while the GeoKV rectifier is calibrated offline and fixed during inference, using $r = 4$ by default.

4.2 MAIN RESULTS

Table 2 compares GeoKV with representative KV-cache compression methods under the same retention ratio $\rho = 0.20$. GeoKV provides a favorable performance trade-off under the same retention budget, consistently remaining close to the full-cache model across different VLMs and multimodal benchmarks. Figure 3 details the performance of Qwen2-VL-7B (Wang et al., 2024b) across four different VLM benchmarks under varying KV cache retention ratios. Experimental results demonstrate that as the KV cache retention ratio decreases, GeoKV consistently outperforms all competing methods, exhibiting slower performance degradation and better robustness in low-retention regimes. These results indicate that incorporating global structural information into token selection can better preserve multimodal task performance under a fixed compression budget. Additional results across retention ratios are provided in the Appendix F.

4.3 EFFICIENCY ANALYSIS

Table 3 summarizes runtime efficiency on a single NVIDIA A100 GPU. At $\rho = 0.20$, GeoKV incurs a 2.0 ms token-selection overhead, which remains comparable to the other compressed methods. Its decoding latency is also comparable at 30.6 ms per step, while the low-rank rectifier introduces

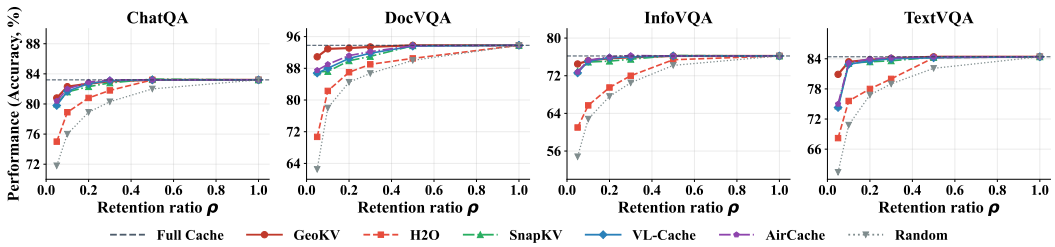


Figure 3: Accuracy (%) of KV-cache compression methods across four VLM benchmarks (ChatQA, DocVQA, InfoVQA, TextVQA) on Qwen2-VL-7B, as a function of the KV-cache retention ratio ρ.

Table 2: Main comparison results across diverse multimodal benchmarks under KV cache retention ratio $\rho = 0.20$. Best results under compression are highlighted in **bold**.

Category	Method	InternVL2-8B				LLaVA-OneVision-7B			
		ChatQA	InfoVQA	DocVQA	TextVQA	ChatQA	InfoVQA	DocVQA	TextVQA
Functional Relevance	H2O	80.7	71.0	82.2	71.8	78.0	60.8	77.3	71.5
	SnapKV	80.9	71.4	90.3	74.8	79.5	64.7	85.0	74.0
	VL-Cache	81.1	72.0	89.5	76.6	79.6	65.1	84.8	74.6
	AirCache	81.7	71.7	89.7	76.3	79.3	65.0	85.1	74.2
Reference	Random	76.8	66.2	77.5	68.4	72.9	57.1	72.6	67.0
	Full Cache	82.2	73.0	91.0	77.7	80.3	66.1	87.0	76.0
Structural Contribution	**GeoKV (Ours)**	**82.0**	**72.8**	**90.3**	**77.1**	**80.0**	**65.8**	**85.8**	**75.5**

only $O(N'dr)$ additional feature-space computation. Overall, GeoKV retains the runtime benefits of KV-cache compression, reducing E2E latency from 4767.7 ms for the full cache to 4493.0 ms at $\rho = 0.20$.

Table 3: Runtime efficiency comparison on an NVIDIA A100 GPU using LLaVA-OneVision-7B with batch size 32, 740 input tokens, and 128 generated tokens. All compressed methods use $\rho = 0.20$.

Method	ρ	**KV Mem** (MB)	**Comp.** (ms)	**TTFT** (ms)	**Dec/Step** (ms)	**E2E** (ms)	**Throughput** (tps)
Full Cache	1.00	1295.0	-	571.7	32.8	4767.7	30.5
Uniform	0.20	259.0	0.2	571.9	30.5	4487.9	32.8
H2O	0.20	259.0	1.4	573.1	30.6	4491.8	32.7
SnapKV	0.20	259.0	2.3	574.0	30.7	4495.6	32.6
VL-Cache	0.20	259.0	2.8	574.6	30.8	4499.3	32.5
GeoKV (Ours)	0.20	259.0	2.0	573.6	30.6	4493.0	32.7

4.4 Structural Analysis and Ablation Studies

To evaluate the structural properties of KV compression, we measure the global geometric distortion (D_F) of post-RoPE key representations and the Future Attention Coverage (FAC) of retained tokens on MSCOCO (Lin et al., 2014) images. Furthermore, we ablate the core components of GeoKV to validate their individual contributions.

Global Geometry and Future Query Coverage. Table 4 shows GeoKV best preserves global geometry and cross-query coverage among non-oracle methods. At $\rho = 0.30$, GeoKV achieves $D_F = 0.295$ and FAC=0.136, significantly outperforming relevance-based baselines like SnapKV (Li et al., 2024) and spatial heuristics. Notably, GeoKV closely approaches the Top-LOO oracle, confirming that GSC-guided selection retains a robust representational basis highly relevant for unseen heterogeneous queries. Full results are in the Appendix E.

Effect of GSC-guided Selection. Table 5 compares GSC against baselines and the LOO oracle. To isolate the selection effect, we ablate key rectification. At $\rho = 0.30$, GSC-only strictly outperforms

Table 4: Structural preservation and cross-query coverage at $\rho = 0.30$. Lower D_F and higher FAC are better. Values are mean over MSCOCO images.

Metric	SnapKV	VL-Cache	AirCache	Uniform	GeoKV (Ours)	Top-LOO
$D_F \downarrow$	0.314	0.308	0.348	0.312	**0.295**	*0.235*
FAC $\uparrow$	0.112	0.118	0.086	0.131	**0.136**	*0.149*

H2O in both accuracy and distortion under similar overhead. Applying rectification (GeoKV) further boosts accuracy to 77.1%. Conversely, per-query refitting yields negligible gains (+0.3%) despite a 2.5× overhead (5.0 ms), confirming structural geometry is query-agnostic. Furthermore, Appendix C shows that GSC remains strictly superior to functional baselines even when both employ identical rectification.

Table 5: Comparison of selection criteria at a retention budget of $\rho = 0.30$. GSC outperforms functional baselines even without rectification, while per-query refitting offers diminishing returns.

Method	Task Acc. (%)	$D_F \downarrow$	Overhead (ms) $\downarrow$	Deployable
GeoKV (GSC + Rect.)	**77.1**	**0.295**	**2.0**	**Yes**
GSC-only (w/o Rect.)	76.8	0.320	1.9	Yes
GSC (per query)	77.4	0.293	5.0	Yes
H2O	74.5	0.345	2.2	Yes
Random	71.2	0.311	0.2	Yes
Top-LOO (Oracle)	*77.3*	*0.235*	*7200.0*	*No*

Effect of Key Rectification. Table 6 evaluates the rectification module and its rank r. Without rectification ($r = 0$), average VQA accuracy decreases to 76.5% with geometric distortion ($D_F = 0.358$). Applying feature-space rectification steadily mitigates this distortion, with performance saturating at $r = 4$. Increasing to $r = 16$ yields only marginal gains (+0.1%) while quadrupling parameter and computation costs. These results validate our default choice of $r = 4$ as an optimal trade-off between geometric restoration and efficiency.

Table 6: Ablation of geometry-aware key rectification and the effect of rank r on LLaVA-OneVision-7B at $\rho = 0.20$. The setting $r = 0$ denotes GeoKV without rectification. Relative parameter and computation costs are normalized to our default setting ($r = 4$).

Rank (r)	Avg. VQA Acc. (%) $\uparrow$	$D_F \downarrow$	Rel. Params.	Rel. Compute
0 (w/o Rect.)	76.5	0.358	0.00×	0.00×
1	76.6	0.356	0.25×	0.25×
2	76.7	0.353	0.50×	0.50×
4 (Default)	**76.8**	**0.350**	**1.00×**	**1.00×**
8	76.9	0.348	2.00×	2.00×
16	76.9	0.347	4.00×	4.00×

5 Conclusion

In this work, we proposed **GeoKV**, a geometry-aware framework for KV cache compression. We showed that query-dependent functional relevance does not fully capture the structural contribution of cached keys, and introduced **Global Structural Contribution (GSC)** for query-independent, geometry-aware cache selection. We further proposed a lightweight low-rank key rectification module to mitigate compression-induced geometric distortion. Experiments across multiple vision-language models and multimodal benchmarks demonstrate that GeoKV can substantially reduce KV cache size while preserving model performance with limited runtime overhead.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript editing, language refinement, literature checking, reference formatting, and improving the clarity of mathematical exposition. No experimental results were generated by AI tools. All technical content, methodological decisions, experimental results, and conclusions were independently verified by the authors, who take full responsibility for the final manuscript.

ETHICS STATEMENT

This work focuses on KV cache compression for existing vision-language models, and does not involve new human-subject data collection or sensitive personal data. Our experiments use publicly available models, datasets, and benchmarks. Since KV cache compression preserves the underlying capabilities and limitations of the original models, the resulting compressed models may inherit their existing biases and potential misuse risks. We therefore encourage responsible use in accordance with the licenses and usage policies of the corresponding models and datasets.

REPRODUCIBILITY STATEMENT

We provide the complete formulation and optimization procedure of GeoKV in Section 3, including the Global Structural Contribution (GSC) criterion and the geometry-aware key rectification mechanism. Section 4 and the Appendix report the calibration data, model configurations, compression settings (e.g., retention ratios), evaluation benchmarks, and baseline methods used in our experiments. These details are provided to facilitate reproducible evaluation and comparison.

REFERENCES

Payman Behnam, Yaosheng Fu, Ritchie Zhao, Po-An Tsai, Zhiding Yu, and Alexey Tumanov. Rocketkv: Accelerating long-context llm inference via two-stage kv cache compression. *arXiv preprint arXiv:2502.14051*, 2025.

Prajjwal Bhattarai, Mohammad Amjad, Dmytro Zhylko, and Tuka Alhanai. Knowledge distillation through geometry-aware representational alignment. *arXiv preprint arXiv:2509.25253*, 2025.

Zefan Cai, Yichi Zhang, Bofei Gao, Yuliang Liu, Yucheng Li, Tianyu Liu, Keming Lu, Wayne Xiong, Yue Dong, Junjie Hu, et al. Pyramidkv: Dynamic kv cache compression based on pyramidal information funneling. *arXiv preprint arXiv:2406.02069*, 2024.

Zefan Cai, Wen Xiao, Hanshi Sun, Yikai Zhang, Ke Wan, Yucheng Li, Yeyang Zhou, Li-Wen Chang, Jiuxiang Gu, Zhen Dong, et al. R-kv: Redundancy-aware kv cache compression for reasoning models. *Advances in neural information processing systems*, 38:60980–61005, 2026.

Chi-Chih Chang, Wei-Cheng Lin, Chien-Yu Lin, Chong-Yan Chen, Yu-Fang Hu, Pei-Shuo Wang, Ning-Chi Huang, Luis Ceze, Mohamed Abdelfattah, and Kai-Chiang Wu. Palu: Kv-cache compression with low-rank projection. In *International Conference on Learning Representations*, volume 2025, pp. 50222–50249, 2025.

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*, pp. 19–35. Springer, 2024a.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 24185–24198, 2024b.

Alessio Devoto, Yu Zhao, Simone Scardapane, and Pasquale Minervini. A simple and effective l_2 norm-based strategy for kv cache compression. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 18476–18499, 2024.

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 *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 23958–23967. IEEE, 2025.

Gopi Krishna Jha, Sameh Gobriel, Liubov Talamanova, and Nilesh Jain. Kvcrush: Key value cache size-reduction using similarity in head-behaviour. *arXiv preprint arXiv:2503.00022*, 2025.

Xiangmin Jiao. Unified smw-like identities of low-rank updates for generalized inverses and pseudoinverses. *SIAM Journal on Matrix Analysis and Applications*, 46(3):1917–1929, 2025.

Hao Kang, Qingru Zhang, Souvik Kundu, Geonhwa Jeong, Zaoxing Liu, Tushar Krishna, and Tuo Zhao. Gear: An efficient kv cache compression recipe for near-lossless generative inference of llm. *arXiv preprint arXiv:2403.05527*, 2024.

Jaeseong Lee, Seung-won Hwang, Aurick Qiao, Daniel Campos, Zhewei Yao, and Yuxiong He. Tale: Token-adaptive low-rank kvcache approximation with reconstruction elimination. *Transactions of the Association for Computational Linguistics*, 13:1298–1318, 2025.

Junyan Li, Yang Zhang, Muhammad Yusuf Hassan, Talha Chafekar, Tianle Cai, Zhile Ren, Pengsheng Guo, Foroozan Karimzadeh, Chong Wang, and Chuang Gan. Commvq: Commutative vector quantization for kv cache compression. *arXiv preprint arXiv:2506.18879*, 2025.

Yuhong Li, Yingbing Huang, Bowen Yang, Bharat Venkitesh, Acyr Locatelli, Hanchen Ye, Tianle Cai, Patrick Lewis, and Deming Chen. Snapkv: Llm knows what you are looking for before generation. *Advances in Neural Information Processing Systems*, 37:22947–22970, 2024.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. pp. 740–755. Springer, 2014.

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

Zihan Liu, Wei Ping, Rajarshi Roy, Peng Xu, Chankyu Lee, Mohammad Shoeybi, and Bryan Catanzaro. Chatqa: Surpassing gpt-4 on conversational qa and rag. *Advances in Neural Information Processing Systems*, 37:15416–15459, 2024.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 2199–2208. IEEE, 2021.

Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and CV Jawahar. Infographicvqa. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2582–2591. IEEE, 2022.

George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming*, 14(1):265–294, 1978.

Shengling Qin, Hao Yu, Chenxin Wu, Zheng Li, Yizhong Cao, Zhengyang Zhuge, Yuxin Zhou, Wentao Yao, Yi Zhang, Zhengheng Wang, et al. Vlcache: Computing 2% vision tokens and reusing 98% for vision-language inference. *arXiv preprint arXiv:2512.12977*, 2025.

Alina Shutova, Vladimir Malinovskii, Vage Egiazarian, Denis Kuznedelev, Denis Mazur, Nikita Surkov, Ivan Ermakov, and Dan Alistarh. Cache me if you must: Adaptive key-value quantization for large language models. *arXiv preprint arXiv:2501.19392*, 2025.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards vqa models that can read. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8309–8318. IEEE, 2019.

Hanlin Tang, Yang Lin, Jing Lin, Qingsen Han, Danning Ke, Shikuan Hong, Yiwu Yao, and Gongyi Wang. Razorattention: Efficient kv cache compression through retrieval heads. In *International Conference on Learning Representations*, volume 2025, pp. 16632–16646, 2025.

Jiaming Tang, Yilong Zhao, Kan Zhu, Guangxuan Xiao, Baris Kasikci, and Song Han. Quest: Query-aware sparsity for efficient long-context llm inference. *arXiv preprint arXiv:2406.10774*, 2024.

Dezhan Tu, Danylo Vashchilenko, Yuzhe Lu, and Panpan Xu. Vl-cache: Sparsity and modality-aware kv cache compression for vision-language model inference acceleration. In *International Conference on Learning Representations*, volume 2025, pp. 219–239, 2025.

Luning Wang, Shiyao Li, Xuefei Ning, Zhihang Yuan, Shengen Yan, Guohao Dai, and Yu Wang. Cskv: Training-efficient channel shrinking for kv cache in long-context scenarios. *arXiv preprint arXiv:2409.10593*, 2024a.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024b.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *International Conference on Learning Representations*, volume 2024, pp. 21875–21895, 2024.

Xianglong Yan, Zhiteng Li, Tianao Zhang, Haotong Qin, Linghe Kong, Yulun Zhang, and Xiaokang Yang. Recalkv: Low-rank kv cache compression via head reordering and offline calibration. *arXiv preprint arXiv:2505.24357*, 2025.

Cheng Yang, Yang Sui, Jinqi Xiao, Lingyi Huang, Yu Gong, Chendi Li, Jinghua Yan, Yu Bai, Ponnuswamy Sadayappan, Xia Hu, et al. Topv: Compatible token pruning with inference time optimization for fast and low-memory multimodal vision language model. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19803–19813. IEEE, 2025.

Yaoxin Yang, Peng Ye, Xudong Tan, Chongjun Tu, Maosen Zhao, Jia Hao, and Tao Chen. Revisiting multimodal kv cache compression: A frequency-domain-guided outlier-kv-aware approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 39550–39560, 2026.

Zhenyu Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark Barrett, et al. H2o: Heavy-hitter oracle for efficient generative inference of large language models. *Advances in neural information processing systems*, 36:34661–34710, 2023.

A Additional Implementation Details

GSC-guided Selection. The regularized log-determinant objective in Eq. 3 uses a ridge parameter $\lambda > 0$ to ensure numerical stability when the selected keys span a low-dimensional subspace. Unless otherwise specified, we set $\lambda = 0.01$ for all experiments. The same value is used throughout the greedy selection procedure and remains fixed across all retention ratios.

Rectifier Calibration. The geometry-aware key rectification module is calibrated offline prior to inference. We construct the calibration set $\mathcal{D}_{\text{cal}}$ by randomly sampling M multimodal instances from the MSCOCO training set. For each calibration instance, we first construct the original KV cache and then apply the same GSC-guided selection procedure used during inference to obtain the compressed cache. The resulting pairs of original and compressed feature-space matrices are used to optimize the layer- and head-specific low-rank rectifiers.

The low-rank factors $A^{(l,h)}$ and $B^{(l,h)}$ are initialized from a standard normal distribution and scaled by 10^{-4}, so that the initial transformation remains close to the identity mapping. Optimization uses AdamW with a learning rate of 10^{-3}, weight decay of 0.01, and 500 optimization steps. After calibration, all rectification parameters are frozen and reused during inference. Unless otherwise specified, we use $M = 128$ calibration samples and rectification rank $r = 4$.

Efficiency Measurement. Runtime experiments are conducted on a single NVIDIA A100 GPU using LLaVA-OneVision-7B. We report KV-cache memory, compression overhead, TTFT, per-step decoding latency, end-to-end (E2E) latency, and decoding throughput. All compressed methods use the same retention ratio and inference configuration. E2E latency is measured directly from the beginning of inference to the completion of generation.

B Submodularity and Greedy Approximation Guarantee

For a normalized objective

$$f(\mathcal{S}) = \mathcal{F}(\mathcal{S}) - \mathcal{F}(\emptyset), \tag{23}$$

we have $f(\emptyset) = 0$. For each KV head, adding a candidate token induces the positive semidefinite rank-one update

$$C^{(h)}_{\mathcal{S}\cup\{i\}} = C^{(h)}_{\mathcal{S}} + k^{(h)}_i k^{(h)\top}_i. \tag{24}$$

For $\mathcal{A} \subseteq \mathcal{B}$ and $i \notin \mathcal{B}$, the marginal gain is

$$\begin{aligned} \Delta^{(h)}_i(\mathcal{S}) &= \log\det\left(C^{(h)}_{\mathcal{S}\cup\{i\}} + \lambda I\right) - \log\det\left(C^{(h)}_{\mathcal{S}} + \lambda I\right) \\ &= \log\left(1 + k^{(h)\top}_i \left(C^{(h)}_{\mathcal{S}} + \lambda I\right)^{-1} k^{(h)}_i\right). \end{aligned} \tag{25}$$

Because $C^{(h)}_{\mathcal{A}} \preceq C^{(h)}_{\mathcal{B}}$, we have

$$(C^{(h)}_{\mathcal{A}} + \lambda I)^{-1} \succeq (C^{(h)}_{\mathcal{B}} + \lambda I)^{-1},$$

and therefore

$$\Delta^{(h)}_i(\mathcal{A}) \geq \Delta^{(h)}_i(\mathcal{B}). \tag{26}$$

Hence the head-wise log-determinant objective is monotone submodular. Averaging over KV heads preserves both properties, so $f(\mathcal{S})$ is also monotone submodular.

Under the cardinality constraint $|\mathcal{S}| \leq N'$, the standard greedy algorithm therefore satisfies

$$f(\mathcal{S}_{\text{greedy}}) \geq \left(1 - \frac{1}{e}\right) f(\mathcal{S}^*), \tag{27}$$

where $\mathcal{S}^*$ is the optimal subset of size at most N' (**?**). Equivalently,

$$\mathcal{F}(\mathcal{S}_{\text{greedy}}) - \mathcal{F}(\emptyset) \geq \left(1 - \frac{1}{e}\right) \left[\mathcal{F}(\mathcal{S}^*) - \mathcal{F}(\emptyset)\right]. \tag{28}$$

C Component-wise Ablation

C.1 Rectification Compatibility with Functional Selection

To examine whether the performance gain of GeoKV is attributable primarily to the rectification module, we evaluate whether existing functional-relevance-based criteria can achieve similar results when equipped with our geometry-aware key rectifier. Specifically, we apply the low-rank rectification procedure to the strongest functional baseline, SnapKV.

Experimental Fairness. To ensure a strictly fair comparison, we do not simply transfer the rectifier parameters calibrated on GeoKV to SnapKV. Instead, we independently calibrate a new rectifier specifically for SnapKV. This is crucial because different token selection criteria (e.g., current-query attention vs. GSC) yield entirely different post-compression feature-space distributions. To isolate the effect of the selection criterion, the SnapKV rectifier is calibrated directly on the feature-space matrices compressed by SnapKV, using the exact same hyperparameter configuration as GeoKV: identical rank ($r = 4$), calibration set size ($M = 128$), optimizer (AdamW), optimization budget (500 steps), and initialization (standard normal scaled by 10^{-4}).

Table 7: Compatibility of geometry-aware rectification with different token selection criteria on LLaVA-OneVision-7B at $\rho = 0.20$. SnapKV's rectifier is independently calibrated on its own compressed representations under identical hyperparameter settings.

Method	**Avg. VQA Acc. (%)** ↑	D_F ↓
SnapKV (w/o Rect.)	75.8	0.375
SnapKV + Rectification	76.1	0.362
GSC-only (w/o Rect.)	76.5	0.358
GeoKV (GSC + Rect.)	**76.8**	**0.350**

As shown in Table 7, while SnapKV benefits from independent geometry-aware rectification (mitigating some structural distortion), its overall performance remains substantially lower than GeoKV. This confirms that query-dependent functional selection permanently discards critical structural information that cannot be fully recovered by subsequent feature-space transformations. Consequently, GSC-guided selection remains strictly superior to functional baselines even when both employ identical rectification setups.

C.2 Combining Functional Relevance and Structural Contribution

Although functional relevance and structural contribution characterize different aspects of cached states, they can theoretically be combined through a two-stage selection procedure. We evaluate an attention-screening variant that first constructs an intermediate candidate pool (e.g., top $2 \times N'$ tokens) using current-query attention, and then applies GSC within this pool to select the final N' tokens. This design preserves the query-dependent screening stage while allowing global geometric diversification at the final selection stage.

Table 8: Comparison of functional, structural, and two-stage selection strategies on LLaVA-OneVision-7B at $\rho = 0.20$.

Selection Strategy	**Avg. VQA Acc. (%)** ↑	D_F ↓
Attention-only (SnapKV)	75.8	0.375
Attention → GSC (Two-stage)	76.1	0.365
GSC-only (Ours)	**76.5**	**0.358**

Table 8 demonstrates that restricting GSC to an attention-filtered candidate pool limits its ability to preserve the global structural geometry, leading to higher distortion and lower accuracy compared to pure GSC selection. This indicates that tokens with minimal current-query relevance often play pivotal roles in maintaining the feature-space scaffold, which justifies our choice of applying GSC globally rather than sequentially after functional screening.

D SENSITIVITY TO CALIBRATION SET SIZE

The rectifier is calibrated offline using a set $\mathcal{D}_{\text{cal}}$ containing M multimodal instances. Table 9 evaluates the sensitivity to M on LLaVA-OneVision-7B at $\rho = 0.20$ with rank $r = 4$.

Table 9: Effect of calibration set size M on geometric distortion and average VQA accuracy on LLaVA-OneVision-7B at $\rho = 0.20$ with $r = 4$.

Calibration Size (M)	$D_F \downarrow$	**Avg. VQA Acc. (%)** $\uparrow$
32	0.361	75.1
64	0.355	76.2
128 (Default)	**0.350**	**76.8**
256	0.349	76.9
512	0.348	76.9

The results show that the rectifier benefits from a relatively small calibration set. Performance improves as M increases from 32 to 128 and changes only marginally beyond $M = 128$, motivating our default choice of $M = 128$.

E STRUCTURAL ANALYSIS ACROSS RETENTION RATIOS

We provide additional structural analyses across a broader range of retention ratios. The geometric distortion D_F measures the deviation between the compressed and original feature-space second-moment matrices, while FAC measures the attention coverage of unseen future queries. Lower D_F and higher FAC are preferred.

E.1 GEOMETRIC DISTORTION

Table 10 reports the geometric distortion across retention ratios. The values are consistent with the structural analysis reported in the main paper.

E.2 FUTURE ATTENTION COVERAGE

Table 11 reports Future Attention Coverage (FAC) across retention ratios. Higher FAC indicates better coverage of unseen future queries.

F DETAILED TASK PERFORMANCE ACROSS RETENTION RATIOS

As a supplement to the experimental analysis in the main text, Table 12 provides the detailed numerical results of average task accuracy across varying KV cache retention ratios (ρ) for two representative models: LLaVA-OneVision-7B and InternVL2-8B.

Consistent with our visual and structural analyses, GeoKV demonstrates superior robustness across different architectures. Notably, in extreme compression regimes (e.g., $\rho \leq 0.10$), functional baselines such as H2O and SnapKV suffer from severe performance degradation due to the irreversible loss of global structural tokens. In contrast, GeoKV effectively mitigates this issue through GSC-guided selection and geometry-aware rectification, maintaining a significantly higher performance bound that consistently approaches the full-cache baseline as the retention budget increases.

Table 10: Geometric distortion D_F across retention ratios. Lower is better.

Method	**0.05**	**0.10**	**0.20**	**0.30**	**0.40**	**0.50**
Full Cache	0.000	0.000	0.000	0.000	0.000	0.000
Top-Attn	0.612	0.498	0.402	0.340	0.291	0.248
H2O	0.618	0.503	0.405	0.345	0.295	0.251
SnapKV	0.589	0.471	0.375	0.314	0.268	0.227
PyramidKV	0.601	0.485	0.392	0.331	0.283	0.240
FastV	0.585	0.466	0.371	0.312	0.265	0.224
VL-Cache	0.582	0.462	0.367	0.308	0.262	0.221
TopV	0.605	0.488	0.395	0.336	0.286	0.243
AirCache	0.615	0.501	0.408	0.348	0.298	0.253
FlashCache	0.588	0.469	0.373	0.315	0.270	0.228
GeoKV (Ours)	**0.568**	**0.445**	**0.350**	**0.295**	**0.252**	**0.213**
Uniform	0.591	0.473	0.378	0.312	0.265	0.225
Random	0.589	0.471	0.376	0.311	0.264	0.224
Top-LOO (Oracle)	*0.435*	*0.341*	*0.272*	*0.235*	*0.198*	*0.165*

Table 11: Future Attention Coverage (FAC) across retention ratios. Higher is better. Values are mean over MSCOCO images.

Method	**0.05**	**0.10**	**0.20**	**0.30**	**0.40**	**0.50**
Full Cache	1.000	1.000	1.000	1.000	1.000	1.000
Top-Attn	0.021	0.038	0.054	0.069	0.085	0.102
H2O	0.023	0.041	0.058	0.074	0.091	0.109
SnapKV	0.035	0.062	0.088	0.112	0.138	0.165
PyramidKV	0.028	0.052	0.075	0.096	0.118	0.142
FastV	0.033	0.059	0.084	0.107	0.132	0.158
VL-Cache	0.038	0.066	0.093	0.118	0.144	0.172
TopV	0.026	0.048	0.070	0.091	0.113	0.135
AirCache	0.025	0.046	0.066	0.086	0.106	0.128
FlashCache	0.040	0.069	0.097	0.123	0.150	0.179
GeoKV (Ours)	**0.045**	**0.077**	**0.108**	**0.136**	**0.165**	**0.195**
Uniform	0.043	0.074	0.104	0.131	0.160	0.190
Random	0.039	0.067	0.095	0.120	0.147	0.175
Top-LOO (Oracle)	*0.051*	*0.086*	*0.121*	*0.149*	*0.181*	*0.210*

Table 12: Detailed average task accuracy (%) of LLaVA-OneVision-7B and InternVL2-8B under different KV cache retention ratios (ρ). Best results under each compression ratio are highlighted in bold.

Method	**LLaVA-OneVision-7B**					**InternVL2-8B**				
	0.05	**0.10**	**0.20**	**0.30**	**0.50**	**0.05**	**0.10**	**0.20**	**0.30**	**0.50**
Random	51.2	58.4	67.4	71.2	74.8	58.3	65.1	72.2	75.4	78.5
H2O	57.5	65.2	71.9	74.5	76.2	64.2	70.5	76.4	78.2	79.8
SnapKV	64.1	70.8	75.8	76.8	77.2	70.3	75.2	79.4	80.2	80.8
VL-Cache	64.5	71.2	76.0	76.9	77.3	70.8	75.5	79.8	80.5	80.9
GeoKV (Ours)	**69.8**	**74.1**	**76.8**	**77.1**	**77.4**	**74.5**	**78.2**	**80.6**	**80.9**	**81.0**
Full Cache			77.4					81.0		

G Additional Efficiency Results

Table 13 reports the runtime behavior of GeoKV across different retention ratios on the same NVIDIA A100 setup as in the main paper. As expected, KV-cache memory decreases approximately linearly with the retention ratio, while the one-time compression overhead remains nearly unchanged because the GSC selection procedure is executed once during cache construction.

Table 13: GeoKV efficiency across retention ratios on LLaVA-OneVision-7B.

Method	ρ	**KV Mem (MB)**	**Comp. (ms)**	**TTFT (ms)**	**Dec/Step (ms)**	**E2E (ms)**	**Throughput (tps)**
GeoKV	0.50	647.5	2.0	573.7	31.4	4595.8	31.8
GeoKV	0.30	388.5	2.0	573.7	30.9	4526.3	32.4
GeoKV	0.10	129.5	2.0	573.7	30.3	4456.8	33.0
GeoKV	0.05	64.8	2.0	573.7	30.2	4439.4	33.1