SHARESVD: SHARING KEY SUBSPACES FOR LOW-RANK GENERATIVE TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Singular value decomposition provides a practical low-rank factorization of attention projections, reducing parameter count and computation while supporting compact latent KV caches. Existing methods typically factorize Query, Key, and Value projections independently, ignoring both their unequal sensitivity to rank reduction and cross-head subspace redundancy. Our systematic study across vision-language models (VLMs) reveals a consistent asymmetry: Value is the most fragile projection, whereas Key is the most robust. Motivated by this observation, we introduce ShareSVD, which adaptively groups geometrically compatible Key heads and shares a low-dimensional input basis within each group while retaining head-specific output factors. This sharing reduces the parameters required for Key factorization, enabling more low-rank directions to be retained in the more sensitive Query and Value projections. At a retained QKV ratio of 0.3, ShareSVD improves average accuracy over the competitive baseline by 8.57% across three VLMs and two benchmarks, while reducing low-rank KV-cache memory by 14.7%. Consistent perplexity improvements on dense and MoE Qwen3 models further demonstrate that ShareSVD generalizes beyond multimodal architectures. Code is available at `https://anonymous.4open.science/r/ShareSVD-F0E0`.

1 INTRODUCTION

Large language models (LLMs) and vision-language models (VLMs) have demonstrated remarkable capabilities in text generation (Bai et al., 2025; Kim et al., 2025; Hwang et al., 2026), visual question answering (Jang & Kim, 2026; An et al., 2026; Hong et al., 2026), and multimodal reasoning (Lee et al., 2024; Li et al., 2025; Jin et al., 2026). Yet, their inference cost scales rapidly with model size and sequence length. Linear projections for Query (Q), Key (K), and Value (V) contribute substantial compute and memory overhead during generation. Reducing these projections without degrading generation quality is thus critical for deployment.

Low-rank matrix factorization offers a promising route toward this goal. Given a large projection matrix, truncated Singular Value Decomposition (SVD) represents it as the product of a narrow input-side factor and a compact output-side factor, with their intermediate rank controlling the degree of compression. A lower retained rank increases compression but discards more singular directions, potentially increasing approximation error. Early neural-network compression methods directly applied truncated SVD to pretrained weight matrices or tensors (Xue et al., 2013; Denton et al., 2014). Recent Transformer-specific approaches improve approximation quality through activation-aware scaling (Yuan et al., 2023) and truncation-aware optimization (Wang et al., 2025b). PaLU (Chang et al., 2025) further demonstrates that the intermediate coordinates of factorized K/V projections can serve as compact KV-cache states. More recently, WSVD (Wang et al., 2026) applies per-head SVD to attention projections, using task-aware importance to assign heterogeneous ranks across heads.

These methods share a common paradigm: Q, K, and V are factorized separately, and each projection is optimized primarily for its own reconstruction. This implicitly assumes that low-rank capacity is equally valuable across all three projections. We re-examine this assumption through a systematic single-projection sensitivity study. As shown in Figure 1, reducing the retained rank of one projection while keeping the other two intact reveals a striking projection-level asymmetry: V is

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

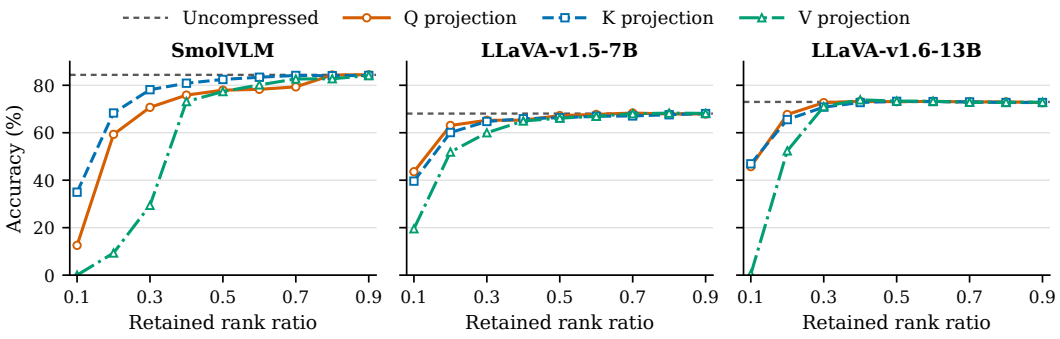


Figure 1: Sensitivity of individual attention projections on ScienceQA-IMG.

consistently the most sensitive, Q is intermediate, and K is remarkably robust. This result suggests that treating all three projections in the same manner can remove valuable directions from sensitive projections while leaving compressible structure in K underexploited.

Exploiting K redundancy naturally points toward parameter sharing across attention heads. While structural cross-head sharing, such as Grouped-Query Attention (Ainslie et al., 2023) and post-training head-grouping heuristics (Chen et al., 2024; Jin et al., 2025), reduces KV overhead, such techniques typically operate at the coarse granularity of entire attention heads and require substantial retraining. This suggests a finer alternative within SVD-based compression: exploiting common input subspaces without forcing complete Key heads to share the same projection.

Motivated by this opportunity and the observed QKV sensitivity asymmetry, we present ShareSVD, an SVD-based compression framework that exploits shareable input subspaces across K heads. ShareSVD organizes geometrically compatible Key heads into adaptive groups. Within each group, the heads share a low-dimensional input basis while retaining head-specific output factors. By avoiding a separate input-side factor for every member head, this representation reduces the parameters required for K factorization. The released parameter capacity allows more directions to be retained in the more sensitive Q and V projections. Because sharing is introduced within the low-rank factors rather than through an attention-wide architectural conversion, ShareSVD can be constructed from pretrained weights and adapted using a small calibration set and short local refinement. Our main contributions are summarized as follows:

- We identify a projection-level compression asymmetry in pretrained attention: Value is consistently the most sensitive under aggressive rank reduction, while Key provides comparatively robust capacity that can be reclaimed.
- We introduce ShareSVD, combining geometry-guided K head sharing with rank reallocation to reinvest capacity released by shared K bases into sensitive Q/V projections.
- Evaluations across three VLMs, two multimodal benchmarks, and dense and MoE Qwen3 models show clear gains in rank-scarce regimes, competitive performance at higher retention ratios, and reduced low-rank KV-cache memory overhead.

2 Related Work

SVD-based model compression. Early neural-network compression methods applied SVD-based low-rank approximation to weight matrices (Xue et al., 2013) and convolutional filters (Denton et al., 2014). For LLMs and VLMs, subsequent work has moved beyond weight-only truncation by incorporating calibration data into the approximation process. Weighted low-rank factorization (Hsu et al., 2022) optimizes a data-dependent reconstruction objective, while ASVD (Yuan et al., 2023) uses activation statistics to rescale weights before decomposition. SVD-LLM (Wang et al., 2025b) further introduces truncation-aware data whitening to better align weight approximation with activation reconstruction, and SVD-LLM V2 (Wang et al., 2025a) improves singular-component selection. Recent methods allocate low-rank capacity at finer granularities: WSVD (Wang et al., 2026) applies per-head SVD to attention projections and assigns heterogeneous ranks based on task importance.

Beyond weight compression, all such low-rank factorizations of Key/Value projections naturally enable a compact KV cache. Formally, for a Key or Value projection $W_t \in \mathbb{R}^{d\times d_t}$ ($t \in \{K, V\}$), a factorization $W_t \approx A_t B_t$ with $A_t \in \mathbb{R}^{d\times r_t}$ and $B_t \in \mathbb{R}^{r_t\times d_t}$ ($r_t < d_t$) allows caching the latent tensor $C_t = XA_t \in \mathbb{R}^{L\times r_t}$ instead of the full state $XW_t \approx C_t B_t \in \mathbb{R}^{L\times d_t}$ for an input sequence $X \in \mathbb{R}^{L\times d}$. This reduces the cache footprint from Ld_t to Lr_t elements per factorized unit, whether the factorization operates on the full projection matrix or on individual heads. Our work builds on this line of methods by jointly considering projection sensitivity and cross-head subspace similarity.

Attention sharing. Analyses of multi-head attention (MHA) show that individual heads exhibit functional redundancy (Voita et al., 2019; Michel et al., 2019). Architectural variants such as Multi-Query Attention (Shazeer, 2019) and Grouped-Query Attention (Ainslie et al., 2023) share Key and Value heads across Query heads. For pretrained MHA models, alignment-based conversion (Jin et al., 2025) and progressive head fusion in DHA (Chen et al., 2024) restructure head-sharing patterns, while MHA2MLA (Ji et al., 2025) converts MHA into latent attention structures (DeepSeek-AI, 2024). However, these structural transformations typically require expensive continued training on millions to billions of tokens to recover performance. In contrast, ShareSVD introduces structural sharing within the low-rank input factors of compatible Key heads. By sharing the input factor A_t and latent state C_t across heads while retaining head-specific output factors B_t, ShareSVD maintains head expressiveness while drastically reducing cache memory, requiring only a compact calibration set (e.g., 256 samples) and rapid local refinement.

3 METHOD

3.1 PROBLEM FORMULATION

We consider SVD-based low-rank compression of the Query (Q), Key (K), and Value (V) projections in an N-layer Transformer with hidden dimension d, h attention heads, and head dimension $d_h = d/h$. For each projection type $t \in \{Q, K, V\}$ in layer ℓ, the dense weight matrix $W_t^{(\ell)} \in \mathbb{R}^{d\times d}$ is partitioned along the output dimension into h head blocks $W_t^{(\ell,i)} \in \mathbb{R}^{d\times d_h}$ for $i \in [h] = \{1, \dots, h\}$. Let Θ_{QKV} collect all dense QKV parameters, and let $|\cdot|$ denote the number of scalar parameters in a parameter set. Given a target retention ratio $\rho \in (0, 1]$, the overall QKV parameter budget is

$$\mathcal{P}_{\mathrm{QKV}}(\rho) = \rho|\Theta_{\mathrm{QKV}}|. \tag{1}$$

Let $f_{\widehat{\Theta}}$ denote the compressed model, and let $\widehat{\Theta}_{\mathrm{QKV}}$ collect its compressed QKV parameters. Given the target-task data distribution $\mathcal{D}$ and task loss $\mathcal{L}_{\mathrm{task}}$, we formulate QKV compression as

$$\begin{aligned} \min_{\widehat{\Theta}} \quad & \mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\mathcal{L}_{\mathrm{task}}(f_{\widehat{\Theta}}(x), y)\right] \\ \text{s.t.} \quad & |\widehat{\Theta}_{\mathrm{QKV}}| \le \mathcal{P}_{\mathrm{QKV}}(\rho). \end{aligned} \tag{2}$$

3.2 METHOD OVERVIEW

As illustrated in Figure 2, ShareSVD first identifies geometrically compatible Key-head groups using activation-aware subspace distances (Section 3.3), and then applies groupwise joint SVD to construct and score shared-subspace candidates (Section 3.4). It represents the heads in each Key group using a shared input basis and head-specific output factors, quantifies the resulting parameter savings (Section 3.5), reallocates the released capacity across Q/K/V under the same budget (Section 3.6), and locally refines the resulting factors (Section 3.7).

3.3 GEOMETRY-GUIDED ADAPTIVE KEY GROUPING

To identify Key heads that are compatible for input-factor sharing while preserving head-specific outputs, we compare input subspaces in an activation-aware geometry. Let $\mathcal{D}_{\mathrm{cal}}$ denote the calibration set, and let $\overline{X}_\ell \in \mathbb{R}^{T_\ell\times d}$ stack the T_ℓ token activations collected at the input of layer ℓ. We compute an upper-triangular Cholesky factor $S_\ell \in \mathbb{R}^{d\times d}$ and transform each projection-head weight according to

$$\overline{X}_\ell^\top \overline{X}_\ell + \varepsilon_\ell I = S_\ell^\top S_\ell, \qquad \widetilde{W}_t^{(\ell,i)} = S_\ell W_t^{(\ell,i)}. \tag{3}$$

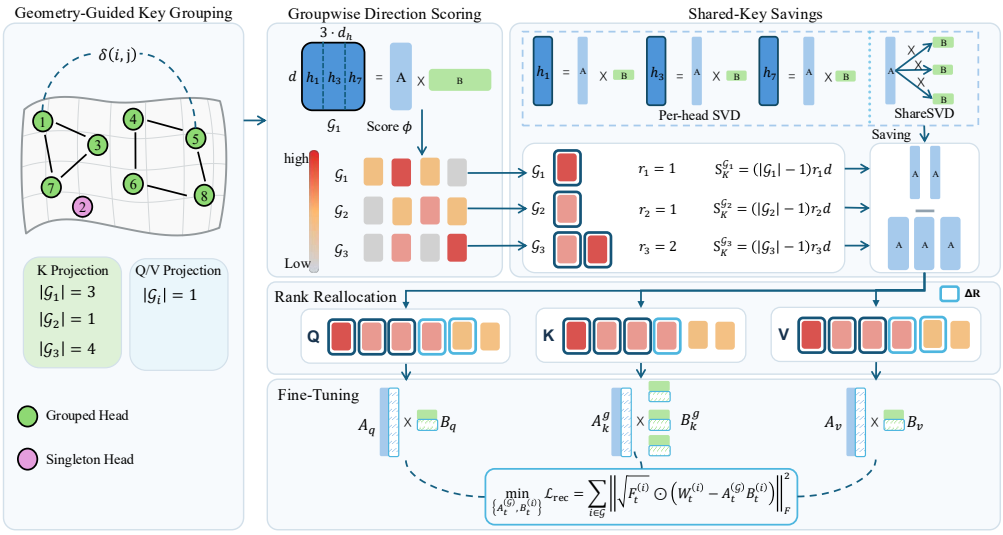


Figure 2: Overview of ShareSVD framework.

Here $\varepsilon_\ell \geq 0$ is a numerical stabilization term applied when needed. The full Cholesky factor preserves cross-channel activation correlations, so discrepancies in the transformed weights reflect reconstruction error on the calibration outputs. From a per-head SVD of $\widetilde{W}_K^{(\ell,i)}$, let $\Psi^{(\ell,i)} \in \mathbb{R}^{d \times r_{\text{probe}}}$ contain its leading $r_{\text{probe}} \leq d_h$ left singular vectors. We measure the distance between Key heads i and k using the principal angles between their probe subspaces:

$$\delta_\ell(i,k) = \left[\sum_{m=1}^{r_{\text{probe}}} \arccos^2\Big(\sigma_m\Big(\big(\Psi^{(\ell,i)}\big)^\top \Psi^{(\ell,k)} \Big) \Big) \right]^{1/2}, \tag{4}$$

where $\sigma_m(\cdot)$ denotes the m-th singular value. A smaller $\delta_\ell(i,k)$ indicates greater overlap between the dominant input subspaces of the two heads. Let $\overline{\delta}_\ell$ be the mean pairwise distance in layer ℓ, and let $\overline{\delta}_{\text{all}}$ be the mean over all Key-head pairs and layers. We set the layer-adaptive threshold to $\tau_\ell = \max(\overline{\delta}_\ell, \overline{\delta}_{\text{all}})$, using the model-wide mean as a common reference while adapting to the subspace variation within each layer. Complete-linkage agglomeration partitions the h Key heads into disjoint groups $\Pi_{\ell,K}$ subject to

$$\max_{i,k \in \mathcal{G}} \delta_\ell(i,k) \leq \tau_\ell, \qquad |\mathcal{G}| \leq g_{\max}, \qquad \mathcal{G} \in \Pi_{\ell,K}. \tag{5}$$

This constraint requires every pair of heads within a group to satisfy the distance threshold and limits each group to at most $g_{\max}$ heads. Heads without a compatible multi-head group remain singletons. For example, the partition illustrated in Figure 2 is $\Pi_{\ell,K} = \{\{1,3,7\},\{2\},\{4,5,6,8\}\}$, where the first and third sets are shared Key groups and $\{2\}$ is a singleton. Query and Value remain singleton partitions, $\Pi_{\ell,Q} = \Pi_{\ell,V} = \{\{1\},\ldots,\{h\}\}$.

3.4 Groupwise Direction Scoring

After forming the projection-specific groups, ShareSVD evaluates the task utility of each candidate direction relative to its parameter cost and selects directions globally under a common QKV budget. We concatenate the transformed head weights along the output dimension:

$$\widetilde{W}_t^{(\ell,\mathcal{G})} = \Big[\widetilde{W}_t^{(\ell,i)}\Big]_{i \in \mathcal{G}} = U_t^{(\ell,\mathcal{G})} \operatorname{diag}\Big(\boldsymbol{\sigma}_t^{(\ell,\mathcal{G})}\Big) \Big(V_t^{(\ell,\mathcal{G})}\Big)^\top. \tag{6}$$

Each singular triplet defines one candidate direction for the corresponding projection group. Let $\sigma_{t,j}^{(\ell,\mathcal{G})}$ denote the j-th entry of $\boldsymbol{\sigma}_t^{(\ell,\mathcal{G})}$. Retaining this direction requires one d-dimensional input vector shared within $\mathcal{G}$ and one d_h-dimensional output row for each member head, giving a parameter

cost of $d + |\mathcal{G}|d_h$. We measure its task utility per parameter using

$$\phi_{t,j}^{(\ell,\mathcal{G})} = \frac{\left|\sigma_{t,j}^{(\ell,\mathcal{G})}\right|}{d + |\mathcal{G}|d_h} \left\{ \mathbb{E}_{z\sim\mathcal{D}_{\text{cal}}} \left[\left(\frac{\partial \mathcal{L}_{\text{task}}(z)}{\partial \sigma_{t,j}^{(\ell,\mathcal{G})}} \right)^2 \right] \right\}^{1/2}. \tag{7}$$

The singular value captures the scale of the direction, while the gradient term estimates its influence on the target task. Normalizing by its parameter cost provides a common priority for comparing candidate directions across projections, layers, and groups.

3.5 Shared-Key Savings

The compatible Key groups allow ShareSVD to replace separate per-head input factors with a single group-shared factor while preserving a distinct output factor for every head. For a Key group $\mathcal{G}$ with rank r, each member head is represented as

$$W_K^{(\ell,i)} \approx A_K^{(\ell,\mathcal{G})} B_K^{(\ell,i)}, \qquad i \in \mathcal{G}, \tag{8}$$

where $A_K^{(\ell,\mathcal{G})} \in \mathbb{R}^{d\times r}$ is shared across the group and $B_K^{(\ell,i)} \in \mathbb{R}^{r\times d_h}$ remains head-specific. We choose to share A_K rather than B_K for two practical reasons. Since $d \gg d_h$, A_K contains substantially more parameters than B_K, so sharing it yields greater parameter savings at the same rank and group size. Moreover, low-rank KV caching stores the latent state $X_\ell A_K$. Sharing only B_K would leave A_K head-specific and thus require a separate latent cache for each head. For a layer input X_ℓ, the shared latent representation is computed once and reused by all member heads:

$$C_K^{(\ell,\mathcal{G})} = X_\ell A_K^{(\ell,\mathcal{G})}, \qquad K^{(\ell,i)} = C_K^{(\ell,\mathcal{G})} B_K^{(\ell,i)}. \tag{9}$$

Let $g = |\mathcal{G}|$. Independently factorizing the g heads at rank r requires $gr(d + d_h)$ parameters, whereas the shared representation requires only $r(d + gd_h)$ parameters. The resulting saving is $gr(d+d_h) - r(d+gd_h) = (g-1)rd$. To quantify this saving at target ratio ρ, we define a reference that independently compresses each head using the same total parameter budget. Its full-model direction count is

$$R_t^{\text{ref}}(\rho) = \left\lfloor \frac{\rho \sum_{\ell=1}^{N} |W_t^{(\ell)}|}{d + d_h} \right\rfloor, \qquad t \in \{Q, K, V\}. \tag{10}$$

For Key, we select the top $R_K^{\text{ref}}(\rho)$ candidate directions globally across all layers and adaptive groups according to ϕ. The number of selected directions within each group defines its reference rank $r_{K,\text{ref}}^{(\ell,\mathcal{G})}(\rho)$. The total capacity released by Key sharing is

$$\mathcal{S}_K(\rho) = d \sum_{\ell=1}^{N} \sum_{\mathcal{G}\in\Pi_{\ell,K}} (|\mathcal{G}| - 1)\, r_{K,\text{ref}}^{(\ell,\mathcal{G})}(\rho). \tag{11}$$

3.6 Rank Reallocation

We reallocate the capacity released by Key sharing across QKV using a simple ratio-dependent rule. Specifically, we set $\Gamma_K(\rho) = \rho\mathcal{S}_K(\rho)$ and $\Gamma_Q(\rho) = \Gamma_V(\rho) = \frac{1-\rho}{2}\mathcal{S}_K(\rho)$. Although Value is more sensitive to rank reduction, increasing its rank also enlarges the latent KV cache, whereas increasing Query rank does not. We therefore split the non-Key allocation equally between Query and Value as a simple trade-off between preserving sensitive projections and retaining the cache savings from Key sharing. Under aggressive compression, Query and Value receive a larger combined share, while more capacity is allocated back to Key as ρ increases.

Let $\overline{\kappa}_t$ denote the average parameter cost of a direction in projection t. Query and Value contain only singleton groups, so $\overline{\kappa}_Q = \overline{\kappa}_V = d + d_h$. For Key, letting $\overline{g}_K$ denote the direction-weighted mean group size gives $\overline{\kappa}_K \approx d + \overline{g}_K d_h$. The allocated capacity therefore increases the retained ranks by approximately

$$\Delta R_t(\rho) \approx \left\lfloor \frac{\Gamma_t(\rho)}{\overline{\kappa}_t} \right\rfloor, \qquad R_t^{\text{new}}(\rho) = R_t^{\text{ref}}(\rho) + \Delta R_t(\rho). \tag{12}$$

Final candidates are selected by ϕ using their realized costs subject to the QKV budget in Equation 1. Let $\mathcal{J}_t^{(\ell,\mathcal{G})} \subseteq \{1,\dots,r_{\text{full}}\}$ denote the indices of the directions selected for projection group $(t,\mathcal{G})$, where $r_{\text{full}} = \min(d, |\mathcal{G}|d_h)$ is the number of candidate directions produced by the groupwise SVD in Equation 6. Let $V_t^{(\ell,i)} \in \mathbb{R}^{d_h \times r_{\text{full}}}$ denote the block of rows in $V_t^{(\ell,\mathcal{G})}$ associated with head i. The selected input and output factors are initialized as

$$A_t^{(\ell,\mathcal{G})} = S_\ell^{-1}\left(U_t^{(\ell,\mathcal{G})}\right)_{:,\mathcal{J}_t^{(\ell,\mathcal{G})}}, \qquad B_t^{(\ell,i)} = \operatorname{diag}\Big(\boldsymbol{\sigma}_t^{(\ell,\mathcal{G})}\left[\mathcal{J}_t^{(\ell,\mathcal{G})}\right]\Big)\left(V_t^{(\ell,i)}\right)^\top_{:,\mathcal{J}_t^{(\ell,\mathcal{G})}}. \tag{13}$$

For Key, $A_K^{(\ell,\mathcal{G})}$ is shared across the group, while each head retains its own $B_K^{(\ell,i)}$. For Query and Value, the same construction applies to their singleton groups. Appendix A shows that each head-specific output factor is least-squares optimal conditional on the selected input subspace under the Cholesky-weighted Frobenius metric, and that retaining all nonzero joint singular directions yields exact recovery.

3.7 Fisher-Weighted Local Fine-Tuning

Once the compressed structure is fixed, low-rank truncation leaves reconstruction errors between initialized factors and original projection weights. To reduce these errors without changing the selected structure, we locally fine-tune the low-rank factors. Following WSVD (Wang et al., 2026), we minimize the Fisher-weighted reconstruction objective

$$\min_{\{A_t^{(\ell,\mathcal{G})},\, B_t^{(\ell,i)}\}} \mathcal{L}_{\text{rec}}^{(\ell)} = \sum_{t\in\{Q,K,V\}} \sum_{\mathcal{G}\in\Pi_{\ell,t}} \sum_{i\in\mathcal{G}} \left\| \sqrt{F_t^{(\ell,i)}} \odot \left(W_t^{(\ell,i)} - A_t^{(\ell,\mathcal{G})} B_t^{(\ell,i)}\right)\right\|_F^2. \tag{14}$$

Here $F_{t,pq}^{(\ell,i)} = \mathbb{E}_{z\sim\mathcal{D}_{\text{cal}}}[(\partial\mathcal{L}_{\text{task}}(z)/\partial W_{t,pq}^{(\ell,i)})^2]$ is the empirical Fisher estimate for entry (p,q) of head i. The square root and $\odot$ are applied elementwise. The Fisher weights leave the original projection weights as the reconstruction targets while assigning larger penalties to approximation errors at more task-sensitive entries. Within each group, the shared input factor $A_t^{(\ell,\mathcal{G})}$ and the head-specific output factors $\{B_t^{(\ell,i)}\}_{i\in\mathcal{G}}$ are optimized jointly.

4 Experiments

4.1 Setup

We compare ShareSVD with ASVD (Yuan et al., 2023), SVD-LLM (Wang et al., 2025b), and WSVD (Wang et al., 2026) on SmolVLM-Instruct (Marafioti et al., 2025), LLaVA-v1.5-7B (Liu et al., 2023a), and LLaVA-v1.6-Vicuna-13B (Liu et al., 2023b), using the official implementation and default hyperparameters provided by each baseline. Following the WSVD setup (Wang et al., 2026), we randomly sample 256 examples from the ScienceQA training split for calibration and use the same samples for 100 local fine-tuning iterations of ShareSVD. The resulting compressed models are evaluated on ScienceQA-IMG (Lu et al., 2022) and SEED-Bench-IMG (Li et al., 2023). All methods use the same parameter-ratio definitions and target QKV budgets, with $\rho \in \{0.3, 0.4, \dots, 0.9\}$. We report the mean performance across three random seeds for each method. The associated error bars and $\pm$ values represent the sample standard deviation. For adaptive grouping in ShareSVD, we set the probe rank to $r_{\text{probe}} = \min(d_h, 32)$ and the maximum group size to $g_{\max} = 8$. All experiments are conducted on NVIDIA H20 GPUs.

4.2 Main Results across Two Benchmarks

Table 1 reports accuracy at each retained QKV ratio, where Avg_L averages $\rho \le 0.5$ and Avg_A averages all evaluated ratios. The results show that ShareSVD achieves the best Avg_L and Avg_A for every model on both benchmarks, with the largest gains under aggressive compression. At $\rho = 0.3$, ShareSVD outperforms all baselines across the six model-dataset pairs and improves average accuracy over the best-performing baseline, WSVD, by 8.57%. Averaged over the evaluated ratios with $\rho \le 0.5$, its improvement over WSVD is 4.82%. The stronger low-rank results support our design: group-shared Key input bases reduce the input-factor parameter cost and allow more

Table 1: Accuracy (%) on ScienceQA-IMG and SEED-Bench-IMG. Bold marks the best compressed result for each model and retention ratio. FP16 is shown as an uncompressed reference.

Model	Method	$\rho = .3$	$\rho = .4$	$\rho = .5$	$\rho = .6$	$\rho = .7$	$\rho = .8$	$\rho = .9$	Avg_L	Avg_A
ScienceQA-IMG ↑										
SmolVLM	ASVD	0.05	0.05	0.00	0.05	0.25	3.62	29.20	0.03	4.75
	SVD-LLM	0.35	0.25	0.79	0.55	3.72	17.25	40.31	0.46	9.03
	WSVD	31.52	53.15	69.81	74.53	80.23	83.04	84.40	51.49	68.10
	ShareSVD	**47.23**	**66.19**	**73.49**	**80.10**	**82.47**	**83.57**	**84.48**	**62.30**	**73.93**
	FP16				84.38				84.38	84.38
LLaVA-v1.5-7B	ASVD	4.91	12.15	13.93	33.42	46.11	48.88	49.03	10.33	29.78
	SVD-LLM	41.99	51.56	55.13	57.26	61.58	63.56	65.15	49.56	56.60
	WSVD	56.19	62.45	64.37	66.29	67.18	**67.63**	67.81	61.00	64.56
	ShareSVD	**59.31**	**63.86**	**65.56**	**66.67**	**67.28**	66.96	**67.94**	**62.91**	**65.37**
	FP16				68.07				68.07	68.07
LLaVA-v1.6-13B	ASVD	58.40	68.42	69.51	72.29	71.34	71.00	71.34	65.44	68.90
	SVD-LLM	66.24	67.72	70.00	70.90	71.39	72.09	72.63	67.99	70.14
	WSVD	60.29	71.53	**73.54**	**73.18**	**72.91**	72.76	72.83	68.45	71.01
	ShareSVD	**68.62**	**72.45**	73.49	73.08	72.83	**72.78**	**73.08**	**71.52**	**72.33**
	FP16				72.98				72.98	72.98
SEED-Bench-IMG ↑										
SmolVLM	ASVD	0.15	0.06	0.08	0.16	0.39	1.48	18.03	0.10	2.91
	SVD-LLM	0.41	0.22	0.33	1.22	4.53	15.96	32.69	0.32	7.91
	WSVD	20.12	48.47	61.24	65.50	66.82	**68.48**	**68.80**	43.27	57.06
	ShareSVD	**38.90**	**59.31**	**64.30**	**66.89**	**67.85**	68.20	68.47	**54.17**	**61.99**
	FP16				68.47				68.47	68.47
LLaVA-v1.5-7B	ASVD	6.67	11.90	19.05	31.75	46.44	52.71	53.95	12.54	31.78
	SVD-LLM	46.01	50.41	55.42	54.63	55.25	57.48	57.93	50.61	53.88
	WSVD	55.78	61.38	61.63	**60.95**	60.02	60.13	60.05	59.60	59.99
	ShareSVD	**59.44**	**61.57**	**62.09**	60.26	**60.38**	**60.36**	**60.12**	**61.04**	**60.60**
	FP16				60.18				60.18	60.18
LLaVA-v1.6-13B	ASVD	63.24	67.72	69.02	69.70	69.93	70.25	70.85	66.66	68.67
	SVD-LLM	67.03	68.66	69.55	69.85	70.22	70.59	70.80	68.41	69.53
	WSVD	67.80	70.43	70.91	**71.16**	**71.29**	**71.26**	**71.35**	69.71	70.60
	ShareSVD	**69.59**	**70.86**	**71.08**	71.15	71.25	71.25	71.28	**70.51**	**70.92**
	FP16				71.30				71.30	71.30

directions to be retained in the more sensitive Query and Value projections. At $\rho = 0.9$, the average difference between ShareSVD and WSVD is only 0.02%. At this retention ratio, both methods already approach the accuracy of the uncompressed FP16 reference, leaving limited room for further improvement and making the gains less pronounced than under aggressive compression.

Although calibration and local fine-tuning use only ScienceQA training data, ShareSVD achieves the highest Avg_L for every model on SEED-Bench-IMG, exceeding WSVD by 4.38% on average and demonstrating cross-dataset generalization.

4.3 Why Share Key?

The projection sensitivity results in Figure 1 motivate our choice to share Key because it is more robust to rank reduction than Query and Value. We validate this design choice by comparing Share-Q, Share-K, and Share-V in the aggressive low-rank regime.

Accuracy. At $\rho = 0.3$, Figure 3 shows that Share-K achieves the highest mean accuracy among the three single-projection sharing variants on every model. Its cross-model average reaches 58.39%, compared with 52.42% for Share-Q, 45.64% for Share-V, and 49.33% for WSVD. This ordering agrees with the projection sensitivity in Figure 1: sharing the robust Key projection is consistently safer than sharing the more sensitive Value projection.

Latent-cache allocation. We measure the CUDA memory allocated for the materialized BF16 low-rank KV cache at batch size $B = 1$ and sequence length $L = 2048$. As shown in Figure 4, Share-K reduces the allocation relative to WSVD by 14.7% on average across the three models. Share-Q instead increases the allocation over WSVD by 21.6% on average. Share-V requires 1.9% less allocation than Share-K on average but loses substantially more accuracy.

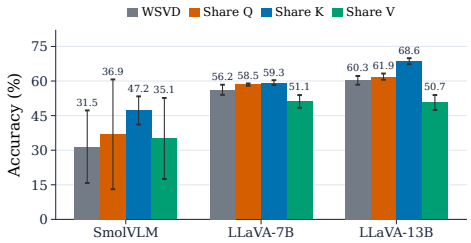


Figure 3: ScienceQA-IMG accuracy under projection sharing at $\rho = 0.3$.

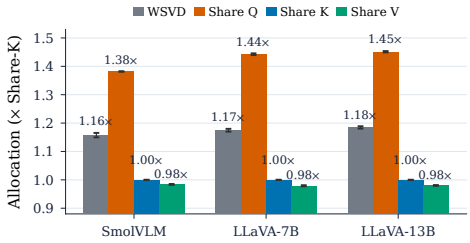


Figure 4: Low-rank KV-cache memory relative to Share-K at $B = 1$, $L = 2048$, and $\rho = 0.3$.

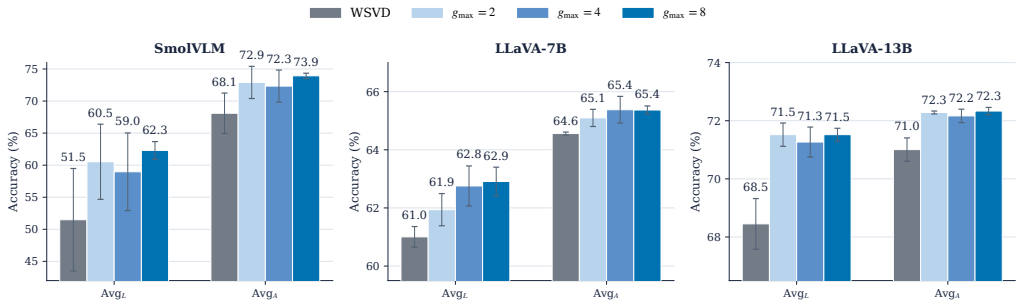


Figure 5: Maximum-group-size ablation on ScienceQA-IMG.

Consequently, Share-K offers the most compelling trade-off, combining superior task performance with consistent latent-cache reduction, justifying its selection as our core architecture.

4.4 How Many Heads Should Share?

We analyze how the maximum group size $g_{\max}$ affects the capacity released by adaptive Key sharing. Figure 5 reports the results for $g_{\max} \in \{2, 4, 8\}$. Averaged across the three models, increasing $g_{\max}$ from 2 to 8 raises Avg_L from 64.66% to 65.58% and Avg_A from 70.10% to 70.54%. Appendix Table 6 helps explain this improvement. As $g_{\max}$ increases from 2 to 4 and 8, the realized mean group size increases from 1.66 to 2.37 and 2.87, respectively. Larger realized groups allow each shared Key input direction to serve more heads, further reducing the input-factor parameter cost and releasing additional capacity for Query and Value. Despite the differences among the three settings, every value of $g_{\max}$ outperforms WSVD for every model in both Avg_L and Avg_A. This consistent advantage shows that the improvement comes from adaptive Key sharing itself and is robust to the exact choice of the maximum group size.

4.5 Why Geometry-Guided Grouping?

We compare geometry-guided grouping with a random grouping baseline that preserves the same all-layer group-size profile. As shown in Table 2, geometry-guided grouping achieves higher average accuracy than matched random grouping for every model, with improvements of 5.97%, 2.07%, and 1.50% for SmolVLM, LLaVA-7B, and LLaVA-13B, respectively. This advantage is particularly evident in SmolVLM, where random grouping exhibits erratic performance scaling—suffering a severe degradation to 45.74% at $\rho = 0.4$ despite an increased parameter budget. In contrast, Grassmann grouping ensures robust and predictable improvements across all retention ratios. With group sizes and budgets matched, the gains reflect geometry-aware head assignment rather than additional parameter savings. Unlike random grouping, which forces incompatible Key heads into a common low-dimensional basis, Grassmann grouping aligns heads with compatible subspaces, yielding a more effective shared representation. These

Table 2: Grassmann versus matched random grouping on ScienceQA-IMG.

Model	Grouping	ρ = 0.3	ρ = 0.4	ρ = 0.5	Avg.
SmolVLM	Random	**49.58**	45.74	**73.67**	56.33
	Grassmann	47.23	**66.19**	73.49	**62.30**
LLaVA-7B	Random	55.54	62.29	64.68	60.84
	Grassmann	**59.31**	**63.86**	**65.56**	**62.91**
LLaVA-13B	Random	63.39	**73.00**	**73.67**	70.02
	Grassmann	**68.62**	72.45	73.49	**71.52**

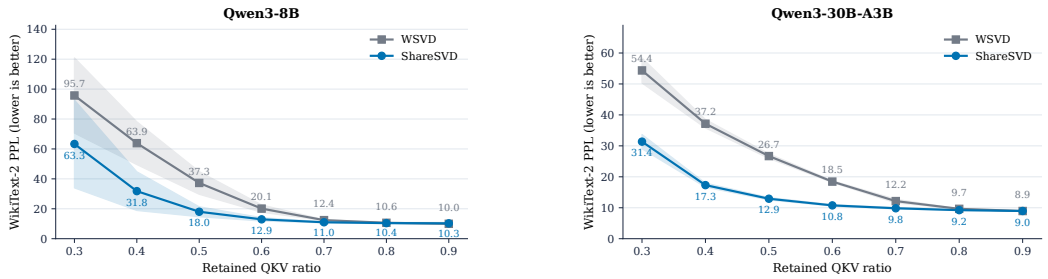


Figure 6: WikiText-2 perplexity on dense and MoE Qwen3 models.

results validate geometry-guided head compatibility as an important component of ShareSVD, particularly in stabilizing compression under constrained parameter budgets.

4.6 Quantization Compatibility

ShareSVD and quantization address complementary forms of redundancy. ShareSVD reduces Key input-factor parameters through groupwise basis sharing, whereas W8A8 reduces the numerical precision of model weights and intermediate activations. We apply W8A8 quantization directly to the low-rank ShareSVD. As shown in Table 3, W8A8 reduces the average accuracy over the evaluated ratios with $\rho \leq 0.5$ by 0.86% across the three models. The reduction ranges from 0.34% to 1.34% for individual models. These results show that ShareSVD remains effective after W8A8 quantization and that structural Key sharing can be combined with numerical compression.

Table 3: Accuracy of FP16 and W8A8 ShareSVD on ScienceQA-IMG.

Model	Precision	ρ 0.3	ρ 0.4	ρ 0.5	Avg.
SmolVLM	FP16	**47.23**	66.19	**73.49**	**62.30**
	W8A8	45.81	**66.55**	71.86	61.41
LLaVA-7B	FP16	**59.31**	**63.86**	**65.56**	**62.91**
	W8A8	58.68	63.66	65.36	62.57
LLaVA-13B	FP16	**68.62**	**72.45**	**73.49**	**71.52**
	W8A8	66.45	71.72	72.35	70.18

4.7 Generalization beyond VLMs

To test generalization beyond VLMs, we apply ShareSVD to the text-only dense Qwen3-8B and MoE Qwen3-30B-A3B models (Yang et al., 2025), using 256 WikiText-2 calibration samples and each model's native tokenizer. We evaluate perplexity on 146 non-overlapping length-2048 windows from the WikiText-2 test split (Merity et al., 2017) and apply ShareSVD to the QKV projections while preserving the native GQA configuration of each model.

Figure 6 shows that ShareSVD reduces three-seed average perplexity over all evaluated ratios from 35.72 to 22.54 on Qwen3-8B and from 23.92 to 14.34 on Qwen3-30B-A3B, relative reductions of 36.9% and 40.0%. ShareSVD achieves lower perplexity than WSVD at six of the seven evaluated ratios on each model, with the largest improvements appearing at ratios with $\rho \leq 0.5$. At $\rho = 0.9$, the two methods converge to similar perplexity because both already retain sufficient attention capacity. These results demonstrate that ShareSVD generalizes from VLMs to text-only dense and MoE architectures and remains effective under native GQA configurations.

5 Conclusion

We present ShareSVD, an SVD-based compression method that groups compatible Key heads to share their input bases and reallocates the resulting parameter savings to retain more directions in the sensitive Query and Value projections. Across three VLMs and two benchmarks, ShareSVD consistently outperforms baselines under aggressive compression. Beyond accuracy gains, ShareSVD reduces the materialized low-rank KV-cache memory by 14.7% on average, while our grouping ablations confirm the necessity of aligning Key-head geometry for effective sharing. Finally, its compatibility with W8A8 quantization and strong results on text-only dense and MoE architectures demonstrate that ShareSVD generalizes across diverse compression regimes and model families.

AI USE STATEMENT

Generative AI tools were employed exclusively to refine phrasing and assist in code implementation. All AI-assisted code was inspected and tested by the authors. The authors maintained complete ownership of the research pipeline, having conceived, executed, and verified all core methodologies, experimental results, and scientific conclusions. The authors take full responsibility for the accuracy and integrity of this work.

ETHICS STATEMENT

This work studies model compression using publicly available pretrained models and datasets, including ScienceQA-IMG, SEED-Bench-IMG, and WikiText-2. It does not involve the collection of new human-subject data, personal information, or other sensitive data. The proposed method aims to reduce the resource requirements of large-model deployment and may consequently help lower its computational and environmental costs. We are not aware of any direct ethical concerns specific to this work.

REPRODUCIBILITY STATEMENT

We take reproducibility seriously and provide the following resources and experimental details:

- **Code and artifacts:** The anonymous repository at `https://anonymous.4open.science/r/ShareSVD-F0E0` provides the complete implementation of ShareSVD, experiment configurations, preprocessing and evaluation scripts, and the commands used to reproduce the reported results.
- **Datasets and models:** All evaluated datasets and pretrained models are publicly available. We report results on ScienceQA-IMG, SEED-Bench-IMG, and WikiText-2, and identify the corresponding model configurations in Section 4.1.
- **Experimental protocol:** Section 4.1 specifies the calibration and fine-tuning procedure, parameter-ratio definition, target QKV budgets, hyperparameter settings, and evaluation protocol. Appendix C provides the complete compression pipeline.
- **Deterministic reproducibility:** All experiments are conducted under strict algorithmic determinism (with fixed seeds, disabled TF32, and deterministic cuBLAS operations). All reported experiments use three fixed random seeds specified in the released scripts, and the reported results are averaged across these runs.
- **Compute resources:** All experiments are conducted on NVIDIA H20 GPUs. Reproducing experiments on the largest evaluated models may require access to comparable computational resources.

Together, these materials provide the information needed to reproduce the experiments and evaluate ShareSVD under the same settings.

REFERENCES

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 4895–4901, 2023.

Jianing An, Luyang Jiang, Jie Luo, Wenjun Wu, and Lei Huang. Voqa: Visual-only question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9100–9109, 2026.

Yushi Bai, Jiajie Zhang, Xin Lv, Linzhi Zheng, Siqi Zhu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. Longwriter: Unleashing 10,000+ word generation from long context llms. In *International Conference on Learning Representations*, 2025.

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

Yilong Chen, Linhao Zhang, Junyuan Shang, Zhenyu Zhang, Tingwen Liu, Shuohuan Wang, and Yu Sun. DHA: Learning decoupled-head attention from transformer checkpoints via adaptive heads fusion. *arXiv preprint arXiv:2406.06567*, 2024.

DeepSeek-AI. DeepSeek-V2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*, 2024.

Emily Denton, Wojciech Zaremba, Joan Bruna, Yann LeCun, and Rob Fergus. Exploiting linear structure within convolutional networks for efficient evaluation. In *Advances in Neural Information Processing Systems*, volume 27, pp. 1269–1277, 2014.

Yuyang Hong, Jiaqi Gu, Yang Qi, Lubin Fan, Yue Wu, Ying Wang, Kun Ding, SHIMING XIANG, Jieping Ye, et al. Knowledge-based visual question answer with multimodal processing, retrieval and filtering. *Advances in Neural Information Processing Systems*, 38:119251–119282, 2026.

Yen-Chang Hsu, Ting Hua, Sung-En Chang, Qian Lou, Yilin Shen, and Hongxia Jin. Language model compression with weighted low-rank factorization. In *International Conference on Learning Representations*, 2022.

Sukjun Hwang, Brandon Wang, and Albert Gu. Dynamic chunking for end-to-end hierarchical sequence modeling. In *International Conference on Learning Representations*, 2026.

Jihyoung Jang and Hyounghun Kim. Aqua: Toward strategic response generation for ambiguous visual questions. In *International Conference on Learning Representations*, 2026.

Tao Ji, Bin Guo, Yuanbin Wu, Qipeng Guo, Lixing Shen, Zhan Chen, Xipeng Qiu, Qi Zhang, and Tao Gui. Towards economical inference: Enabling DeepSeek's multi-head latent attention in any transformer-based LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 33313–33328, 2025.

Qingyun Jin, Xiaohui Song, Feng Zhou, and Zengchang Qin. Align attention heads before merging them: An effective way for converting MHA to GQA. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 8804–8816, 2025.

Zhuoran Jin, Kejian Zhu, Hongbang Yuan, Yupu Hao, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. Look light, think heavy: What multimodal chain-of-thought reasoning can and cannot do. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8567–8608, 2026.

Minbeom Kim, Thibaut Thonet, Jos Rozen, Hwaran Lee, Kyomin Jung, and Marc Dymetman. Guaranteed generation from large language models. In *International Conference on Learning Representations*, 2025.

Junlin Lee, Yequan Wang, Jing Li, and Min Zhang. Multimodal reasoning with multimodal knowledge graph. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10767–10782, 2024.

Bohao Li, Rui Wang, Guangzhi Wang, Yuying Ge, Yixiao Ge, and Ying Shan. SEED-Bench: Benchmarking multimodal LLMs with generative comprehension. *arXiv preprint arXiv:2307.16125*, 2023.

Chengzu Li, Wenshan Wu, Huanyu Zhang, Yan Xia, Shaoguang Mao, Li Dong, Ivan Vulić, and Furu Wei. Imagine while reasoning in space: Multimodal visualization-of-thought. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 36340–36364. PMLR, 13–19 Jul 2025.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023a.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, 2023b.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *Advances in Neural Information Processing Systems*, 2022.

Andrés Marafioti, Orr Zohar, Miquel Farré, Merve Noyan, Elie Bakouch, Pedro Cuenca, Cyril Zakka, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, Vaibhav Srivastav, Joshua Lochner, Hugo Larcher, Mathieu Morlon, Lewis Tunstall, Leandro von Werra, and Thomas Wolf. SmolVLM: Redefining small and efficient multimodal models. *arXiv preprint arXiv:2504.05299*, 2025.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017.

Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems*, 2019.

Noam Shazeer. Fast transformer decoding: One write-head is all you need. *arXiv preprint arXiv:1911.02150*, 2019.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 5797–5808, 2019.

Haiyu Wang, Yutong Wang, Jack Jiang, and Sai Qian Zhang. WSVD: Weighted low-rank approximation for fast and efficient execution of low-precision vision-language models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Xin Wang, Samiul Alam, Zhongwei Wan, Hui Shen, and Mi Zhang. SVD-LLM v2: Optimizing singular value truncation for large language model compression. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 4287–4296, 2025a.

Xin Wang, Yu Zheng, Zhongwei Wan, and Mi Zhang. SVD-LLM: Truncation-aware singular value decomposition for large language model compression. In *The Thirteenth International Conference on Learning Representations*, 2025b.

Jian Xue, Jinyu Li, and Yifan Gong. Restructuring of deep neural network acoustic models with singular value decomposition. In *Proceedings of INTERSPEECH*, pp. 2365–2369, 2013.

An Yang et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Zhihang Yuan, Yuzhang Shang, Yue Song, Qiang Wu, Yan Yan, and Guangyu Sun. ASVD: Activation-aware singular value decomposition for compressing large language models. *arXiv preprint arXiv:2312.05821*, 2023.

A Optimal Output Factors for a Selected Shared Subspace

Consider a fixed layer ℓ and a Key group $\mathcal{G} \in \Pi_{\ell,K}$. For compactness, write the joint SVD as

$$\widetilde{W}_{\mathcal{G}} = \left[\widetilde{W}_K^{(\ell,i)}\right]_{i\in\mathcal{G}} = U_{\mathcal{G}}\,\mathrm{diag}(\boldsymbol{\sigma}_{\mathcal{G}})V_{\mathcal{G}}^{\top}, \tag{15}$$

where $V_{\mathcal{G}}^{\top} = [(V_{\mathcal{G}}^{(i)})^{\top}]_{i\in\mathcal{G}}$. For a selected direction set $\mathcal{J}$, let $U_{\mathcal{G},\mathcal{J}} = (U_{\mathcal{G}})_{:,\mathcal{J}}$.

Table 4: Materialized BF16 low-rank KV-cache allocation (MiB), reported as mean ± sample standard deviation over three seeds.

Model	WSVD	Share Q	Share K	Share V
SmolVLM	113.43$_{\pm 0.81}$	135.40$_{\pm 0.08}$	97.99$_{\pm 0.09}$	96.40$_{\pm 0.07}$
LLaVA-7B	301.82$_{\pm 1.08}$	370.64$_{\pm 0.48}$	256.91$_{\pm 0.23}$	251.56$_{\pm 0.29}$
LLaVA-13B	479.39$_{\pm 1.34}$	587.29$_{\pm 0.34}$	404.64$_{\pm 0.42}$	396.65$_{\pm 0.16}$

Proposition 1 (Least-squares output factors and exact recovery)**.** *Assume that S_ℓ is nonsingular. For any selected direction set $\mathcal{J}$, the head-specific factor in Equation 13 satisfies*

$$\begin{aligned} B_K^{(\ell,i)} &= \operatorname{diag}\left(\boldsymbol{\sigma}_{\mathcal{G}}[\mathcal{J}]\right)\left(V_{\mathcal{G}}^{(i)}\right)_{:,\mathcal{J}}^{\top} \\ &= U_{\mathcal{G},\mathcal{J}}^{\top}\widetilde{W}_K^{(\ell,i)} \\ &= \arg\min_B \left\|\widetilde{W}_K^{(\ell,i)} - U_{\mathcal{G},\mathcal{J}}B\right\|_F^2. \end{aligned} \tag{16}$$

Equivalently, it minimizes the Cholesky-weighted reconstruction error in the original weight coordinates. If $\mathcal{J}$ contains all nonzero joint singular directions, then every group member is reconstructed exactly.

Proof. Restricting the joint SVD to the columns associated with head i gives

$$\widetilde{W}_K^{(\ell,i)} = U_{\mathcal{G}} \operatorname{diag}(\boldsymbol{\sigma}_{\mathcal{G}})(V_{\mathcal{G}}^{(i)})^{\top}. \tag{17}$$

Left-multiplying by $U_{\mathcal{G},\mathcal{J}}^{\top}$ yields

$$U_{\mathcal{G},\mathcal{J}}^{\top}\widetilde{W}_K^{(\ell,i)} = \operatorname{diag}\left(\boldsymbol{\sigma}_{\mathcal{G}}[\mathcal{J}]\right)\left(V_{\mathcal{G}}^{(i)}\right)_{:,\mathcal{J}}^{\top} = B_K^{(\ell,i)}. \tag{18}$$

Because $U_{\mathcal{G},\mathcal{J}}$ has orthonormal columns, the standard least-squares solution for the fixed basis is $B = U_{\mathcal{G},\mathcal{J}}^{\top}\widetilde{W}_K^{(\ell,i)}$, which proves Equation 16. Moreover, with $A_K^{(\ell,\mathcal{G})} = S_\ell^{-1}U_{\mathcal{G},\mathcal{J}}$,

$$\left\|\widetilde{W}_K^{(\ell,i)} - U_{\mathcal{G},\mathcal{J}}B\right\|_F^2 = \left\|S_\ell\left(W_K^{(\ell,i)} - A_K^{(\ell,\mathcal{G})}B\right)\right\|_F^2, \tag{19}$$

establishing the equivalent Cholesky-weighted interpretation. Finally, if $\mathcal{J}$ contains all nonzero joint singular directions, then $U_{\mathcal{G},\mathcal{J}}U_{\mathcal{G},\mathcal{J}}^{\top}$ acts as the identity on every $\widetilde{W}_K^{(\ell,i)}$. Hence,

$$\begin{aligned} A_K^{(\ell,\mathcal{G})}B_K^{(\ell,i)} &= S_\ell^{-1}U_{\mathcal{G},\mathcal{J}}U_{\mathcal{G},\mathcal{J}}^{\top}\widetilde{W}_K^{(\ell,i)} \\ &= S_\ell^{-1}\widetilde{W}_K^{(\ell,i)} \\ &= W_K^{(\ell,i)}. \end{aligned} \tag{20}$$

□

The result holds for any selected index set $\mathcal{J}$, including the non-contiguous sets produced by the task-aware selector.

B Additional Experimental Analysis

B.1 Absolute Low-Rank KV-Cache Memory

Table 4 reports the absolute CUDA allocation underlying the normalized comparison in Figure 4. The buffers are materialized in BF16 at $B = 1$, $L = 2048$, and $\rho = 0.3$.

Table 5: ScienceQA-IMG accuracy for the maximum-group-size ablation across all evaluated retained QKV ratios (ρ), reported as mean $\pm$ sample standard deviation over three seeds. Bold marks the best result for each model and ratio. Avg_L averages $\rho \leq 0.5$ and Avg_A averages all evaluated ratios.

Model	Configuration	$\rho = .3$	$\rho = .4$	$\rho = .5$	$\rho = .6$	$\rho = .7$	$\rho = .8$	$\rho = .9$	Avg_L	Avg_A
SmolVLM	WSVD	31.52±15.74	53.15±12.01	69.81±0.72	74.53±0.25	80.23±2.11	83.04±0.86	84.40±0.27	51.49	68.10
	$g_{\max} = 2$	40.32±15.14	**66.52**±2.59	**74.75**±0.12	79.42±1.04	82.35±0.69	82.55±0.62	84.43±0.17	60.53	72.91
	$g_{\max} = 4$	39.55±18.12	64.24±0.23	73.13±0.95	79.52±0.55	82.30±0.28	83.44±0.69	84.15±0.17	58.97	72.33
	$g_{\max} = 8$	**47.23**±6.12	66.19±1.50	73.49±1.31	**80.10**±0.08	**82.47**±0.75	**83.57**±0.54	**84.48**±0.18	**62.30**	**73.93**
LLaVA-7B	WSVD	56.19±2.20	62.45±0.46	64.37±0.83	66.29±0.60	67.18±0.60	**67.63**±0.13	67.81±0.33	61.00	64.56
	$g_{\max} = 2$	58.06±0.90	63.18±0.75	64.58±0.68	66.88±0.47	**67.34**±0.08	67.54±0.19	**68.09**±0.20	61.94	65.10
	$g_{\max} = 4$	58.88±0.97	63.66±0.83	**65.72**±0.41	**67.15**±0.63	67.16±0.56	67.46±0.31	67.61±0.22	62.75	**65.38**
	$g_{\max} = 8$	**59.31**±1.05	**63.86**±0.83	65.56±0.32	66.67±0.47	67.28±0.10	66.96±0.32	67.94±0.25	**62.91**	65.37
LLaVA-13B	WSVD	60.29±1.90	71.53±0.40	**73.54**±0.32	73.18±0.45	72.91±0.16	72.76±0.19	72.83±0.10	68.45	71.01
	$g_{\max} = 2$	67.99±1.60	**73.48**±0.48	73.10±0.17	**73.33**±0.15	**72.93**±0.13	72.38±0.40	72.78±0.60	**71.52**	72.28
	$g_{\max} = 4$	67.96±2.03	72.52±0.78	73.33±0.31	73.29±0.35	72.81±0.25	**72.81**±0.03	72.43±0.28	71.27	72.16
	$g_{\max} = 8$	**68.62**±1.29	72.45±0.19	73.49±0.67	73.08±0.50	72.83±0.18	72.78±0.55	**73.08**±0.18	**71.52**	**72.33**

Table 6: Key-group structure at $\rho = 0.5$, reported as mean $\pm$ sample standard deviation over three seeds. Overall averages the three models within each seed, and singleton percentage is measured over Key heads.

Metric	$g_{\max}$	SmolVLM	LLaVA-7B	LLaVA-13B	Overall
Mean group size	2	1.60±0.00	1.69±0.00	1.70±0.01	1.66±0.01
	4	2.18±0.03	2.48±0.01	2.46±0.02	2.37±0.01
	8	2.55±0.02	3.03±0.04	3.03±0.04	2.87±0.00
Singletons (%)	2	24.83±0.30	18.03±0.30	17.88±0.66	20.25±0.36
	4	22.66±0.72	14.03±0.15	15.56±0.68	17.42±0.19
	8	22.14±0.00	14.39±0.41	15.27±0.63	17.26±0.08

B.2 Maximum Number of Shared Heads

Table 5 expands the aggregate results in Figure 5 into all evaluated retention ratios. Each entry reports the mean accuracy and sample standard deviation over three seeds.

The ratio-wise results show that the advantage of a larger grouping cap is concentrated in the most compressed setting. At $\rho = 0.3$, the cross-model mean increases from 49.33% for WSVD to 55.46%, 55.46%, and 58.39% for $g_{\max} = 2$, 4, and 8, respectively. The differences narrow as the retained ratio increases, and no single value of $g_{\max}$ is uniformly best at every higher-rank setting. Thus, $g_{\max} = 8$ provides the strongest aggregate trade-off reported in the main text while remaining a permissive cap rather than forcing every Key group to contain eight heads.

Table 6 explains why the effect saturates. On average across the three models, increasing $g_{\max}$ from 2 to 8 raises the realized mean group size only from 1.66 to 2.87 heads, while singleton Key heads remain near 17.3% for $g_{\max} \in \{4, 8\}$. Thus, a larger $g_{\max}$ provides greater grouping flexibility, while the geometry-based criterion still merges only compatible heads rather than forcing groups to reach the size limit.

B.3 Closed-Form Initialization versus Local Fine-Tuning

To isolate the contribution of local fine-tuning, we compare the standard 100-step procedure with an update-free variant of ShareSVD that constructs candidate bases via joint SVD, ranks their task utility using calibration gradients, and instantiates the selected directions in closed form through Equation 13. Table 7 shows that local fine-tuning improves average accuracy across the three models by 11.85% at $\rho = 0.3$, compared with 1.49% at $\rho = 0.4$ and 0.34% at $\rho = 0.5$. Across all evaluated settings, its mean gain is 4.56%, with most of the improvement coming from the unstable SmolVLM initialization at $\rho = 0.3$. The closed-form initialization already matches or exceeds local fine-tuning for LLaVA-7B at $\rho = 0.4$ and 0.5, indicating that the initialized factors are sufficient in the more stable settings while fine-tuning is most useful when the low-rank approximation is harder.

Table 7: Closed-form initialization versus local fine-tuning on ScienceQA-IMG, reported as mean $\pm$ sample standard deviation over three seeds. Bold marks the best result for each model and ratio.

Model	Method	$\rho = 0.3$	$\rho = 0.4$	$\rho = 0.5$	**Avg.**
SmolVLM	No FT	16.16$_{\pm 18.07}$	61.01$_{\pm 1.77}$	71.89$_{\pm 1.72}$	49.69
	Local FT	**47.23**$_{\pm 6.12}$	**66.19**$_{\pm 1.50}$	**73.49**$_{\pm 1.31}$	**62.30**
LLaVA-7B	No FT	58.59$_{\pm 1.09}$	**64.67**$_{\pm 1.19}$	**66.77**$_{\pm 0.50}$	**63.34**
	Local FT	**59.31**$_{\pm 1.05}$	63.86$_{\pm 0.83}$	65.56$_{\pm 0.32}$	62.91
LLaVA-13B	No FT	64.85$_{\pm 4.46}$	72.34$_{\pm 0.22}$	72.88$_{\pm 0.13}$	70.02
	Local FT	**68.62**$_{\pm 1.29}$	**72.45**$_{\pm 0.19}$	**73.49**$_{\pm 0.67}$	**71.52**

C Complete ShareSVD Pipeline

Algorithm 1 outlines the end-to-end operational pipeline of ShareSVD.

Algorithm 1 ShareSVD compression pipeline

Require: Pretrained model f_Θ; calibration set $\mathcal{D}_{\text{cal}}$; retained QKV ratio ρ; maximum Key-group size $g_{\max}$

Ensure: Compressed factors $\widehat{\Theta}$ and Key-head partitions $\{\Pi_{\ell,K}\}_{\ell=1}^{N}$

1: Set $\mathcal{P}_{\text{QKV}}(\rho) \leftarrow \rho|\Theta_{\text{QKV}}|$
2: Collect $\{\overline{X}_\ell\}$ and compute the Cholesky activation factors $\{S_\ell\}$ using Equation 3
3: **for** $\ell = 1, \ldots, N$ **do**
4: Transform the Q/K/V head weights with S_ℓ and extract the Key probe bases $\{\Psi^{(\ell,i)}\}$
5: Form $\Pi_{\ell,K}$ by complete-linkage clustering under Equation 5, and keep Q and V as singleton groups
6: **for** $t \in \{Q, K, V\}$ **do**
7: **for** $\mathcal{G} \in \Pi_{\ell,t}$ **do**
8: Construct the groupwise SVD candidates and their scores $\phi_{t,j}^{(\ell,\mathcal{G})}$ using Equations 6–7
9: **end for**
10: **end for**
11: **end for**
12: Derive the independent Q/K/V reference budgets from ρ, and apply the corresponding reference retention rate to the adaptive Key groups in descending score order
13: Compute the realized shared-Key saving $\mathcal{S}_K(\rho)$ and the projection budgets in Equations 11–12
14: Select candidates within the Q/K/V budgets in descending ϕ order and reconcile the remaining discrete costs to $\mathcal{P}_{\text{QKV}}(\rho)$
15: Construct $A_t^{(\ell,\mathcal{G})}$ and $\{B_t^{(\ell,i)}\}_{i\in\mathcal{G}}$ from the selected direction sets using Equation 13
16: With groups and selected directions fixed, locally fine-tune the factors by minimizing the Fisher-weighted objective in Equation 14
17: **return** $\widehat{\Theta}$ and $\{\Pi_{\ell,K}\}_{\ell=1}^{N}$