FLOW-Q: FLOW CONSISTENCY FOR POST-TRAINING QUANTIZATION OF VISION TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

We present FLOW-Q, a general post-training quantization framework for Vision Transformers. Our analysis shows that the existing sensitivity-weighted reconstruction paradigm induces feature distortion during quantization, degrading the robustness of quantized feature representations. To overcome this challenge, we first propose an evolution-aware learning (EAL) strategy that captures feature evolution information to construct a manifold representation with global topological structure. Moreover, a geometric constraint rectification (GCR) module is designed to utilize the manifold representation to rectify quantized features from both direction and magnitude perspectives, thereby alleviating feature distortion. EAL provides the geometric structural foundation for GCR, enabling the quantized model to inherit the robustness of visual representations from the full-precision network. Extensive experiments on **nine** tasks across **twelve** benchmarks demonstrate the strong generalizability of FLOW-Q, which achieves new state-of-the-art quantization performance across DINOv3, conventional ViTs, and mobile-efficient models. Code is available in the supplementary material.

1 INTRODUCTION

Vision Transformers (Dosovitskiy et al., 2021; Touvron et al., 2021; Liu et al., 2021) exhibit strong contextual modeling capabilities and have achieved remarkable performance across various vision tasks (Yin et al., 2025; Li et al., 2026). However, their high computational complexity and substantial resource demands pose significant challenges for deployment on resource-constrained edge devices. These limitations have driven growing interest in model compression (Simoncini et al., 2025; Zhu et al., 2025) to achieve the trade-off between computational efficiency and prediction accuracy.

Post-training quantization (PTQ) (Wu et al., 2024b; Zhong et al., 2025) compresses pretrained Vision Transformers through block-wise reconstruction using a small calibration set. As shown in Figure 1(a), existing methods (Wu et al., 2025b;a; Hwang et al., 2026) typically use the Hessian matrix H to estimate positional sensitivity to quantization perturbations, and assign greater weights to feature errors at more sensitive positions (Ding et al., 2024). This associates block-level feature matching with the estimated impact of quantization on the model loss, providing a loss-aware criterion for approximating the full-precision (FP) model. Yet a key question remains: *can sensitivity-weighted reconstruction enable the quantized model to inherit the robustness of FP feature representations?*

We investigate this question in §3.1. **First**, we examine response maps derived from the final-layer dense features of DINOv3 (Siméoni et al., 2025). While the FP model clearly delineates foreground objects, its quantized counterparts exhibit fragmented responses, with substantial foreground regions missing. This observation motivates us to seek complementary cues for preserving the robustness of feature representations during quantization. **Second**, given that modern visual backbones progressively transform features through successive blocks, we assess whether Hessian sensitivity can reflect the changes between block inputs and outputs (*i.e.*, feature evolution) in the FP model. We find that position pairs with nearly identical sensitivity can exhibit substantially different feature evolution patterns, with the ratio of evolution discrepancy to sensitivity difference mainly ranging from 53 to 197. Importantly, such input-output changes characterize how representations are updated at individual positions and have been shown to serve as useful cues for token importance estimation (Choi et al., 2025). These findings suggest that feature evolution holds the potential to provide complementary guidance for preserving the robustness of feature representations during quantization.

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.

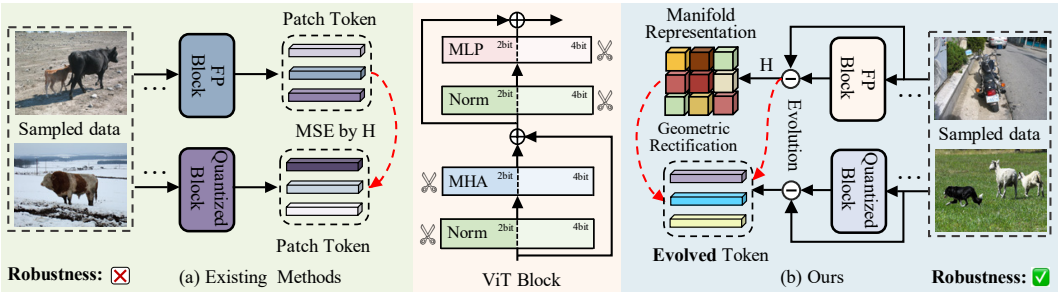

Figure 1: Comparison with prior work. (a) Existing methods perform quantized feature reconstruction by optimizing an MSE objective weighted by Hessian (H) sensitivity; (b) FLOW-Q leverages feature evolution to construct a structured manifold representation for geometric rectification of quantized features, enabling the quantized model to retain FP representation robustness.

To address this challenge, as illustrated in Figure 1(b), we propose FLOW-Q, a general PTQ framework for Vision Transformers that incorporates evolution-aware structural modeling to preserve visual representations during quantization. We first design an **evolution-aware learning (EAL)** strategy that constructs a structural reference from FP feature evolution. With the FP model kept frozen, EAL jointly encodes positional sensitivity and feature update to determine the contribution of each evolution response to this reference. By computing second-order correlations across feature dimensions, EAL aggregates these weighted local responses into a structured manifold representation for subsequent geometric rectification. This representation is computed only ***once*** per block using a small calibration set, which incurs negligible computational overhead (§4.2). We further introduce a **geometric constraint rectification (GCR)** module to calibrate quantized feature evolution with its FP counterpart. Specifically, GCR decomposes the manifold representation to obtain a set of orthonormal bases which span a compact subspace that captures its dominant contextual structure. The FP and quantized evolution matrices are then projected onto this shared subspace, where directional and magnitude matching losses align their orientations and overall magnitudes, respectively. Both losses are invariant to the choice of orthonormal basis within the same subspace, making rectification depend on the underlying geometry rather than arbitrary basis choices. As such, FLOW-Q mitigates feature distortion and allows quantized models to inherit the robustness of FP visual representations.

Extensive experiments on **nine** tasks across **twelve** datasets demonstrate that FLOW-Q consistently achieves state-of-the-art quantization performance across DINOv3, conventional ViTs, and mobile-efficient models. Taking its integration with APHQ-ViT (Wu et al., 2025b) as an example, quantizing DINOv3 (Siméoni et al., 2025) with FLOW-Q under W4A4 yields gains of **2.40%**, **2.80%**, **3.11%**, **0.83%**, and **3.03%** on five training-free tasks of semantic matching, image retrieval, saliency detection, multi-view correspondence, and video object segmentation, respectively. On three training-based tasks under W2A4, FLOW-Q achieves gains of **1.12%** for object detection, **1.30%/1.25%** for depth estimation, and **1.56%** for semantic segmentation. It improves average image classification performance by **1.42%** across ViT, DeiT, and Swin under W2A3. For mobile-efficient models, FLOW-Q achieves average gains of **2.05%** and **2.46%** on MobileNetV2 (Sandler et al., 2018) and MobileViT (Mehta & Rastegari, 2022) under W3A4, respectively.

2 RELATED WORK

Model Compression. Vision Transformer architectures (Dosovitskiy et al., 2021; Touvron et al., 2021; Liu et al., 2021), along with ViT-based vision foundation models such as DINOv3 (Siméoni et al., 2025), have achieved remarkable performance across diverse vision tasks (Cai et al., 2026). However, their substantial computational and resource demands hinder deployment on resource-constrained edge devices. To address this limitation, knowledge distillation (Sun et al., 2024; Gu et al., 2024; Li et al., 2025b), pruning (Haberer et al., 2024; Lei et al., 2025; Simoncini et al., 2025), and quantization (Gholami et al., 2025; Ramachandran et al., 2025; Huang et al., 2026) have emerged as three major directions in model compression. Knowledge distillation (Li et al., 2024) transfers knowledge from a large teacher model to a lightweight student model but often incurs substantial training overhead. Pruning (Muralidharan et al., 2024) removes redundant network components according to carefully designed importance criteria. In contrast, quantization (Ding et al., 2025;

Egiazarian et al., 2026; Xu et al., 2026) reduces computational and memory costs by representing weights and activations at lower bit widths and is broadly categorized into quantization-aware training (QAT) (He et al., 2024; Li et al., 2025a; Morreale et al., 2026) and post-training quantization (PTQ) (Xu et al., 2024; Wang et al., 2025a; Liu et al., 2026). QAT typically requires retraining on large-scale datasets, limiting its practical applicability. By contrast, PTQ relies on only a small calibration set and typically performs block-wise reconstruction, making it the focus of this work.

Post-Training Quantization for ViTs. Existing PTQ paradigms (Wu et al., 2024a) can be broadly categorized into two directions: activation distribution adaptation (Ma et al., 2024; Chen et al., 2025; Wang et al., 2025b) and quantization error compensation (Jiang et al., 2025; Deng et al., 2025). The former focuses on addressing the quantization challenges posed by activation distributions. For instance, NoisyQuant (Liu et al., 2023) and AdaLog (Wu et al., 2024b) improve activation quantizability through uniform noise injection and adaptive logarithmic quantization, respectively. TWEO (Liang et al., 2026) further reveals the structural origins of extreme outliers and suppresses their formation. The latter directly corrects quantization-induced errors. Representative methods include ERQ (Zhong et al., 2025) and AIQViT (Jiang et al., 2025), which reduce block-level reconstruction errors through staged optimization and low-rank compensation, respectively. However, these approaches fail to account for the varying sensitivity of individual positions to the loss under low-bit quantization, thereby limiting their ability to prioritize highly sensitive regions during optimization.

Hessian Sensitivity Learning for PTQ. More recent methods (Zhang et al., 2024; Ding et al., 2024; Edalati et al., 2025) employ Hessian-based importance metrics to characterize the loss sensitivity of individual positions, thereby guiding low-bit quantization toward lower-loss solutions. For instance, APHQ-ViT (Wu et al., 2025b) employs the Average Perturbation Hessian to guide block-wise reconstruction and replaces GELU with ReLU for more accurate activation quantization. FIMA-Q (Wu et al., 2025a) establishes a theoretical connection between KL divergence and the Fisher Information Matrix (FIM), based on which a structured FIM is constructed as an efficient approximation to the Hessian. LS-ViT (Hwang et al., 2026) fits sample-wise second-order estimates under a least-squares objective, which enables it to account for cross-sample covariance terms. The above approaches (Shen et al., 2025; Hwang et al., 2026) rely on Hessian-based sensitivity weighting, which leads to feature distortion and degrades the robustness of visual representations in quantized models. Conversely, FLOW-Q leverages feature evolution to construct a manifold representation, which captures global topological information and guides the geometric rectification of quantized features from both direction and magnitude perspectives, thereby effectively alleviating feature distortion.

3 FLOW CONSISTENCY FOR POST-TRAINING QUANTIZATION

3.1 MOTIVATION ANALYSIS

Existing PTQ methods (Ding et al., 2024; Wu et al., 2025b; Hwang et al., 2026) typically reconstruct quantized models with sensitivity-weighted objectives. However, whether such a paradigm enables the quantized model to inherit the robustness of feature representations from its FP counterpart remains unclear. We investigate this question through the following representation assessment.

Representation Assessment. We obtain different DINOv3 (Siméoni et al., 2025) models quantized using two sensitivity-weighted reconstruction methods, APHQ-ViT (Wu et al., 2025b) and LS-ViT (Hwang et al., 2026). The similarity between the final-layer dense representations and the corresponding `CLS` token is then visualized as response maps in Figure 2 (**Left**). The quantized models exhibit fragmented responses and substantial foreground omissions, indicating that sensitivity-weighted reconstruction fails to effectively transfer the robustness of full-precision feature representations to the quantized models. More visualization results are provided in the *Appendix*.

This observation motivates us to seek complementary cues that can better preserve feature robustness. R-Shift (Choi et al., 2025) has shown that the differences between the inputs and outputs (*i.e.*, feature evolution) of successive blocks provide valuable information for token importance estimation. We then analyze the relationship between sensitivity and feature evolution numerically.

Numerical Analysis. We rank all position pairs by their sensitivity differences in ascending order and select the *top-K* (*i.e.*, 1% and 2%) with the smallest differences in the sixth layer, respectively. We define L as the ratio between the evolution discrepancy and the sensitivity difference at the cor-

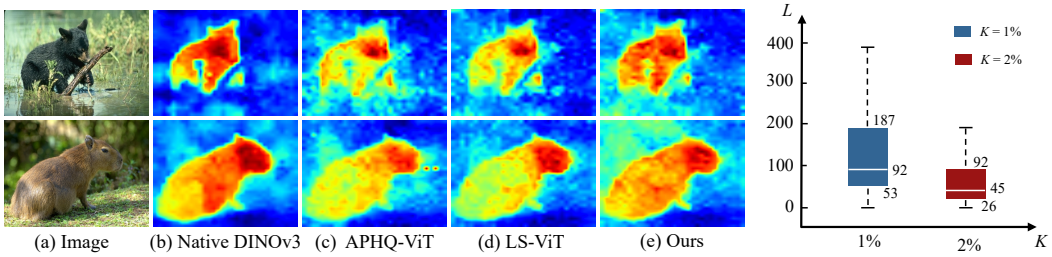

Figure 2: Motivation analysis (§3.1) from the representation and numerical perspectives.

responding positions. Figure 2 (**Right**) illustrates how L varies across different top-ranked ranges of sensitivity differences. For instance, within the top 1% range, the ratio of evolution difference to the corresponding sensitivity discrepancy is mostly concentrated between 53 and 187. This directly demonstrates that even position pairs with highly similar sensitivity can exhibit substantially different feature evolution patterns. The above experimental analysis highlights the potential of feature evolution information to provide effective complementary guidance for PTQ block reconstruction.

Therefore, we propose FLOW-Q, which captures visual information from feature evolution to facilitate the geometric rectification of quantization errors, as illustrated in Figure 3. The evolution-aware learning strategy (§3.2) extracts position-level importance information from feature evolution and constructs a manifold representation that captures the global topological structure. Building on this representation, the geometric constraint rectification module (§3.3) leverages the encoded geometric information to rectify quantized features in terms of both *direction* and *magnitude* alignment. As shown in Figure 2(e), FLOW-Q mitigates feature distortion, enabling the quantized model to preserve the robustness of visual representations from the full-precision network.

3.2 EVOLUTION-AWARE LEARNING

Beyond sensitivity-weighted reconstruction, feature evolution holds the potential to provide effective complementary guidance for PTQ. Building on this insight, we first design an evolution-aware learning strategy that captures visual evolution information and constructs a manifold representation, providing geometric structural information for subsequent rectification of quantized features.

For the i-th FP Transformer block, we denote its input and output by X_{i-1}^{FP} and X_i^{FP}, respectively. Similarly, the quantized block corresponds to X_{i-1}^{Q} and X_i^{Q}. The feature evolution matrices of both blocks are obtained by directly subtracting the inputs from their outputs:

$$\Delta X^{FP} = X_i^{FP} - X_{i-1}^{FP}, \quad \Delta X^{Q} = X_i^{Q} - X_{i-1}^{Q}. \tag{1}$$

Since the full-precision block is pretrained and fixed, it provides stable feature evolution information, with ΔX^{FP} serving as the reference. Regions with larger value variations reflect the point-wise semantic information learned by the current block. The Hessian matrix H inherently characterizes the sensitivity of each position to quantization errors, indicating the importance of the corresponding evolution information. We thus modulate ΔX^{FP} with H to emphasize positions that exhibit pronounced semantic variations and are highly vulnerable to quantization perturbations:

$$\mathcal{W} = \Delta X^{FP} \odot \sqrt{|\mathrm{H}|}, \qquad \mathcal{G} = \mathcal{W}^{\top}\mathcal{W}, \tag{2}$$

$\odot$ denotes the dot product. We use $|\mathrm{H}|$ because our analysis focuses solely on the absolute numerical values of sensitivity, rather than the direction of local curvature. Gram anchoring (Siméoni et al., 2025) is employed to transform the point-wise semantic information contained in the diversified matrix $\mathcal{W}$ into a second-order structured manifold representation $\mathcal{G}$. Its advantage lies in aggregating dispersed local semantic responses and modeling structural relationships across feature dimensions through second-order correlations. The resulting global topology provides a structural basis for subsequent rectification. Note that the manifold representation is computed only ***once*** per block on a small calibration set, introducing no learnable parameters and negligible overhead (§4.2).

3.3 GEOMETRIC CONSTRAINT RECTIFICATION

In this section, we further propose a geometric constraint rectification module, which leverages the rich structural information encoded in the manifold representation to rectify quantized features

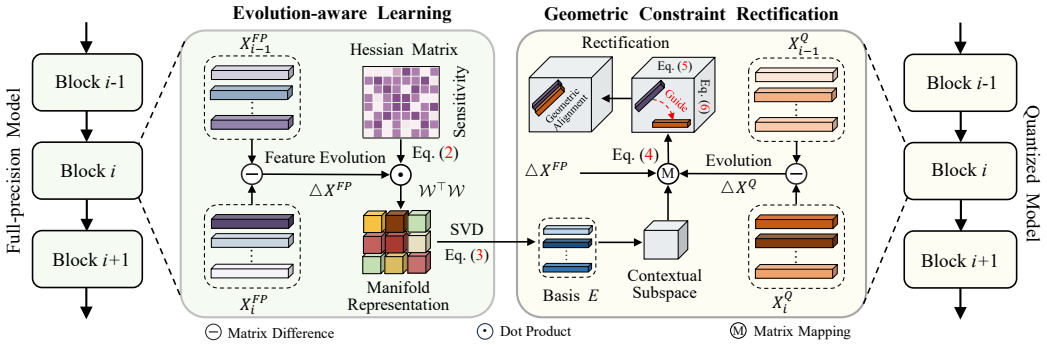

Figure 3: The overall pipeline of FLOW-Q. The evolution-aware learning (§3.2) integrates the local responses from the feature evolution matrix ΔX^{FP} of the FP model and the Hessian matrix into a manifold representation with structured interactions. Moreover, geometric constraint rectification (§3.3) performs SVD on the manifold representation to derive a contextual subspace, and projects both ΔX^{FP} and the evolution matrix ΔX^{Q} of the quantized model into this subspace. It then rectifies the quantized features in terms of both direction (Eq. 5) and magnitude (Eq. 6).

from both *direction* and *magnitude* perspectives. This mitigates feature distortion and enables the quantized model to inherit the robustness of visual representations from the full-precision network.

Subspace Mapping. Although the manifold representation $\mathcal{G}$ encapsulates rich contextual interaction information, not all of its components are equally informative for quantization. To preserve the most important patterns while suppressing noise (Pang et al., 2024), we apply SVD to $\mathcal{G}$ and retain the leading singular vectors to construct a subspace that captures global semantic topology:

$$\mathcal{G} \approx U\Sigma V^{\top}, \quad E = V_{[:,1:\mu]}, \tag{3}$$

where U and V denote the matrices of left and right singular vectors, respectively. Since $\mathcal{G}$ is symmetric positive semidefinite, U and V can be chosen to be identical. Σ is the diagonal matrix of singular values, and E comprises the right singular vectors associated with the truncation rank μ largest singular values, forming a set of orthonormal basis vectors for the contextual subspace.

Given the quantized and FP feature evolutions ΔX^{Q} and ΔX^{FP} of the i-th Transformer block, obtained by Eq. 1, we project the ΔX^{Q} and ΔX^{FP} into the subspace by using E as the basis:

$$M^{Q} = \Delta X^{Q} \otimes E, \quad M^{FP} = \Delta X^{FP} \otimes E. \tag{4}$$

$\otimes$ represents matrix multiplication. M^{Q} and M^{FP} are the mapped feature representations of the quantized and FP models in the subspace, respectively. This mapping enables the subsequent direction and magnitude constraints to leverage the contextual structural information captured by EAL, rather than performing indiscriminate alignment in the original feature space.

Geometric Rectification. To mitigate the geometric deviation, we first impose a directional constraint on M^{Q} to align it with M^{FP}, yielding the directional matching loss $\mathcal{L}_{direct}$, as follows:

$$\mathcal{L}_{direct} = 1 - \frac{Tr((M^{Q})^{\top} \otimes M^{FP})}{||M^{Q}||_{F} \cdot ||M^{FP}||_{F}}, \tag{5}$$

here, $Tr(\cdot)$ denotes the trace of a matrix, and $||\cdot||_{F}$ is the Frobenius norm. The $\mathcal{L}_{direct}$ reaches zero when the mapped representation of the quantized model is perfectly aligned with the FP counterpart. In fact, direction captures only one aspect of feature information, while magnitude constitutes another important aspect (Park et al., 2023). Therefore, we design the magnitude matching loss $\mathcal{L}_{magn}$ to ensure that M^{Q} and M^{FP} exhibit equivalent evolution intensity, formulated as:

$$\mathcal{L}_{magn} = (log(\frac{||M^{Q}||_{F}}{||M^{FP}||_{F}}))^{2}. \tag{6}$$

The complementary directional and magnitude constraints mitigate feature distortion, enabling the quantized model to preserve the robustness of feature representations from its full-precision counterpart. Although the subspace basis obtained through SVD is not unique, the losses defined in Eqs. 5

and 6 are basis-invariant and yield identical values under any orthonormal basis spanning the same subspace. This property makes geometric rectification independent of arbitrary basis choices and ensures stable geometric constraints on quantized features, thereby preserving consistent geometric relationships across different valid subspace bases. *A detailed proof is provided in the Appendix*.

4 EXPERIMENT

This section provides a detailed comparison with state-of-the-art methods, demonstrating the performance gains achieved by FLOW-Q and validating the effectiveness of its individual components. *More dataset details and experimental results are provided in the supplementary material.*

Datasets. The experiments are conducted on **twelve** datasets under different weight (**W**) and activation (**A**) bit-widths. Specifically, we use FG3DCar (Lin et al., 2014b), JODS (Rubinstein et al., 2013), and PASCAL (Hariharan et al., 2011) for semantic matching; Oxford (Radenović et al., 2018) for image retrieval; ECSSD (Shi et al., 2015) and DUTS (Wang et al., 2017) for unsupervised saliency detection; NAVI (Jampani et al., 2023) for multi-view correspondence; DAVIS-2017 (Pont-Tuset et al., 2017) for video object segmentation; COCO (Lin et al., 2014a), NYUv2 (Silberman et al., 2012), and ADE20K (Zhou et al., 2017) for object detection, depth estimation, and semantic segmentation, respectively; and ImageNet (Russakovsky et al., 2015) for image classification.

Evaluation Metrics. Following standard metrics, we report PCK@0.05 for semantic matching; mAP under the medium protocol for image retrieval; F_{max} for saliency detection; average recall at 1, 2, and 5 cm for multi-view correspondence; mean region similarity $\mathcal{J}_m$, mean contour accuracy $\mathcal{F}_m$, and $(\mathcal{J}\&\mathcal{F})_m$ for video object segmentation; $\text{mAP}_{50:95}$ for object detection; ARel / δ_1 for depth estimation; mIoU for semantic segmentation; and Top-1 accuracy for image classification.

Generalization Setting. We evaluate the advantages of FLOW-Q from three perspectives: DINOv3, conventional ViTs, and mobile-efficient models. **First**, we integrate FLOW-Q into two state-of-the-art PTQ frameworks and apply them to the ViT-B-based foundation model DINOv3 (Siméoni et al., 2025), evaluating the quantized backbone on five tasks to assess the robustness of its dense representations. **Second**, we extend FLOW-Q to ViT (Dosovitskiy et al., 2021), DeiT (Touvron et al., 2021), and Swin (Liu et al., 2021) architectures of varying scales, and evaluate the quantized models on image classification to verify architectural generalization. **Third**, we examine its effectiveness on two mobile-efficient models, demonstrating the applicability of FLOW-Q to efficient vision architectures.

Implementation Details. The proposed FLOW-Q is implemented in the PyTorch framework, and all experiments are conducted on four NVIDIA RTX 4090 GPUs. Following APHQ-ViT (Wu et al., 2025b) and LS-ViT (Hwang et al., 2026), we quantize DINOv3 (Siméoni et al., 2025), conventional ViTs, and mobile-efficient models on ImageNet under a unified experimental setup. Specifically, we employ channel-wise uniform quantization for weights and layer-wise uniform quantization for activations, using 1,024 calibration images with a batch size of 32 and optimizing the quantization parameters for 20k iterations. The learning rates for weight and activation quantization are set to 1e-3 and 4e-5, respectively. The quantized models are subsequently frozen and evaluated on the five training-free tasks. *Details of the three training-based tasks are provided in the Appendix*.

4.1 QUANTITATIVE COMPARISON WITH STATE-OF-THE-ARTS

Dense Representation Preservation. We quantize DINOv3 (Siméoni et al., 2025) and evaluate it across five *training-free* and three *training-based* downstream tasks to compare FLOW-Q with SOTA methods in preserving the robustness of dense representations, as reported in Tables 1 and 2. Notably, our method consistently delivers substantial performance gains across all quantization settings. For instance, under W4A4, integrating FLOW-Q into APHQ-ViT (Wu et al., 2025b) yields gains of **2.40%/1.73%/1.35%** on semantic matching, **2.80%** on image retrieval, **3.11%/2.12%** on saliency detection, **0.83%** on multi-view correspondence, and **3.03%/3.11%/2.94%** on video object segmentation. *Interestingly, unlike conventional image-based tasks, video object segmentation is particularly sensitive to cross-frame structural consistency rather than numerical quantization error alone, which may partly explain the superior performance of W3A4 over W4A4.* Consistent performance gains can also be observed on the three training-based tasks. Overall, the results highlight the advantage of FLOW-Q in preserving the robustness of dense representations.

Table 1: Comparison (§4.1) of quantized DINOv3 (Siméoni et al., 2025) across **training-free** tasks. Semantic matching and saliency detection are evaluated on FG3DCar / JODS / PASCAL and ECSSD / DUTS, respectively, while video object segmentation reports $(\mathcal{J}\&\mathcal{F})_m$ / $\mathcal{J}_m$ / $\mathcal{F}_m$.

Method	W/A	Semantic Matching	Image Retrieval	Saliency Detection	Multi-view Corresp.	Video Object Segmentation
Full-Precision DINOv3	32/32	92.00 / 77.10 / 59.70	79.16	57.95 / 44.68	55.91	69.70 / 67.00 / 72.50
QDrop (Wei et al., 2022)	2/4	72.01 / 52.13 / 33.11	8.83	27.90 / 15.32	42.12	30.53 / 29.01 / 32.06
FIMA-Q (Wu et al., 2025a)	2/4	81.35 / 60.84 / 43.80	17.56	31.50 / 20.51	44.26	38.81 / 39.15 / 38.48
TWEO (Liang et al., 2026)	2/4	81.87 / 61.01 / 44.03	15.11	30.42 / 18.60	44.59	37.75 / 38.01 / 37.50
APHQ-ViT (Wu et al., 2025b)	2/4	81.50 / 61.65 / 40.38	11.27	29.38 / 19.61	44.51	35.29 / 35.18 / 35.40
+ FLOW-Q	2/4	**82.25 / 62.17 / 42.73**	**14.69**	**31.07 / 20.37**	**45.25**	**36.93 / 37.49 / 36.37**
LS-ViT (Hwang et al., 2026)	2/4	81.25 / 62.27 / 44.34	22.67	32.79 / 21.79	45.05	42.77 / 42.69 / 42.85
+ FLOW-Q	2/4	**82.16 / 63.32 / 45.78**	**23.86**	**33.81 / 23.00**	**45.63**	**44.14 / 43.95 / 44.34**
QDrop (Wei et al., 2022)	3/4	73.28 / 54.09 / 37.99	29.77	34.21 / 19.12	43.90	37.74 / 37.16 / 38.31
FIMA-Q (Wu et al., 2025a)	3/4	81.85 / 64.07 / 45.23	36.74	51.98 / 36.65	47.41	42.09 / 41.03 / 43.15
TWEO (Liang et al., 2026)	3/4	83.15 / 63.56 / 46.52	34.49	50.82 / 34.63	47.86	40.06 / 39.52 / 40.60
APHQ-ViT (Wu et al., 2025b)	3/4	81.82 / 62.68 / 45.98	34.98	47.01 / 29.93	47.26	38.78 / 38.04 / 39.52
+ FLOW-Q	3/4	**83.71 / 63.09 / 48.79**	**38.60**	**47.88 / 31.09**	**48.21**	**39.33 / 38.86 / 39.81**
LS-ViT (Hwang et al., 2026)	3/4	85.02 / 66.67 / 51.43	48.11	52.77 / 38.25	48.76	48.03 / 47.45 / 48.61
+ FLOW-Q	3/4	**85.68 / 68.01 / 52.90**	**49.28**	**53.39 / 39.58**	**49.25**	**49.70 / 49.41 / 49.99**
QDrop (Wei et al., 2022)	4/4	78.07 / 61.64 / 41.67	31.85	48.82 / 34.38	44.56	37.33 / 37.27 / 37.39
FIMA-Q (Wu et al., 2025a)	4/4	83.27 / 65.81 / 50.31	42.76	56.10 / 41.26	47.87	41.43 / 40.02 / 42.84
TWEO (Liang et al., 2026)	4/4	82.98 / 64.22 / 51.43	38.14	52.10 / 37.92	48.58	38.68 / 38.33 / 39.03
APHQ-ViT (Wu et al., 2025b)	4/4	82.16 / 62.95 / 50.48	37.01	51.24 / 36.49	48.06	35.46 / 34.85 / 36.07
+ FLOW-Q	4/4	**84.56 / 64.68 / 51.83**	**39.81**	**54.35 / 38.61**	**48.89**	**38.49 / 37.96 / 39.01**
LS-ViT (Hwang et al., 2026)	4/4	85.38 / 67.87 / 54.71	53.60	56.74 / 41.23	49.42	46.95 / 46.67 / 47.23
+ FLOW-Q	4/4	**86.10 / 69.52 / 55.68**	**54.92**	**57.57 / 42.19**	**49.93**	**48.11 / 48.11 / 48.10**

Table 2: Comparison (§4.1) of quantized DINOv3 (Siméoni et al., 2025) on **training-based** tasks.

Method	W2A4			W3A4		
	Object Detection	Depth Estimation	Semantic Segmentation	Object Detection	Depth Estimation	Semantic Segmentation
Full-Precision DINOv3	57.80	3.00 / 99.90	51.80	57.80	3.00 / 99.90	51.80
QDrop (Wei et al., 2022)	39.16	19.20 / 88.80	29.81	44.24	17.10 / 90.60	38.65
FIMA-Q (Wu et al., 2025a)	45.88	12.10 / 94.60	33.93	49.13	10.00 / 96.90	41.07
TWEO (Liang et al., 2026)	46.14	11.50 / 95.20	34.20	49.45	8.50 / 97.00	41.23
APHQ-ViT (Wu et al., 2025b)	45.80	12.30 / 94.90	33.68	48.97	9.80 / 96.40	40.61
+ FLOW-Q	**46.92**	**11.00 / 96.15**	**35.24**	**49.84**	**9.10 / 97.20**	**41.79**
LS-ViT (Hwang et al., 2026)	46.19	9.10 / 95.40	35.84	49.96	8.30 / 97.10	42.07
+ FLOW-Q	**48.23**	**6.80 / 97.60**	**36.89**	**51.28**	**5.10 / 98.50**	**43.41**

Architectural Generalization. As shown in Table 3, we seamlessly integrate FLOW-Q into two state-of-the-art methods, APHQ-ViT (Wu et al., 2025b) and LS-ViT (Hwang et al., 2026), and evaluate the resulting performance gains across ViT, DeiT, and Swin models of varying scales on ImageNet (Russakovsky et al., 2015). The experimental results demonstrate that our method delivers substantial performance gains under low-bit quantization. For instance, under the W2A3 setting, FLOW-Q improves LS-ViT (Hwang et al., 2026) by **2.23%**, **2.09%**, **1.43%**, **1.26%**, **1.72%**, and **1.89%** on ViT-S, ViT-B, DeiT-B, DeiT-L, Swin-S, and Swin-B, respectively. When integrated with APHQ-ViT (Wu et al., 2025b), FLOW-Q yields corresponding improvements of **0.86%**, **1.35%**, **0.83%**, **2.52%**, **1.23%**, and **1.66%**. Similar performance gains are consistently observed under the W2A4 quantization setting. Taken together, these results demonstrate the superiority of FLOW-Q in enhancing the robustness of feature representations in quantized models across diverse architectures.

Mobile-efficient Model Adaptability. Table 4 presents the performance of FLOW-Q integrated into SOTA methods across efficient networks of different scales. The results show that our method yields substantial improvements not only on the ViT-based MobileViT (Mehta & Rastegari, 2022) but also

Table 3: Comparison (§4.1) of the Top-1 accuracy on ImageNet (Russakovsky et al., 2015) with different quantization bit-widths.

Method	W/A	ViT-S	ViT-B	DeiT-S	DeiT-B	DeiT-L	Swin-S	Swin-B	Avg.
Full-Precision	32/32	81.39	84.54	79.85	81.80	86.91	83.23	85.27	83.28
BRECQ (Li et al., 2021)	2/3	15.62	52.54	30.88	58.91	72.16	51.92	60.20	48.89
QDrop (Wei et al., 2022)	2/3	17.90	54.22	21.48	54.57	73.40	57.28	57.57	48.06
AdaLog (Wu et al., 2024b)	2/3	35.43	58.36	34.28	60.68	75.31	53.46	62.52	54.29
FIMA-Q (Wu et al., 2025a)	2/3	38.82	61.15	52.57	66.24	76.25	59.57	63.15	59.68
TWEO (Liang et al., 2026)	2/3	41.78	59.20	53.59	66.23	75.99	63.18	65.51	60.78
APHQ-ViT (Wu et al., 2025b)	2/3	41.63	58.06	53.04	66.09	75.40	62.90	65.03	60.31
+ FLOW-Q	2/3	**42.49**	**59.41**	**54.57**	**66.92**	**77.92**	**64.13**	**66.69**	**61.73**
LS-ViT (Hwang et al., 2026)	2/3	42.05	62.89	53.86	67.06	75.84	65.77	67.73	62.17
+ FLOW-Q	2/3	**44.28**	**64.98**	**54.89**	**68.49**	**77.10**	**67.49**	**69.62**	**63.84**
BRECQ (Li et al., 2021)	2/4	45.20	67.18	61.20	69.72	79.04	69.21	70.36	65.99
QDrop (Wei et al., 2022)	2/4	49.35	71.44	45.99	70.01	77.77	70.13	68.64	64.76
AdaLog (Wu et al., 2024b)	2/4	54.69	72.39	54.70	71.34	79.88	70.02	71.72	67.82
FIMA-Q (Wu et al., 2025a)	2/4	56.76	74.41	65.38	73.32	80.66	70.64	72.36	70.50
TWEO (Liang et al., 2026)	2/4	56.18	73.26	65.05	73.40	80.78	72.21	72.95	70.55
APHQ-ViT (Wu et al., 2025b)	2/4	56.04	72.85	65.02	73.07	80.29	71.75	72.64	70.24
+ FLOW-Q	2/4	**57.29**	**73.57**	**65.60**	**73.98**	**81.68**	**73.10**	**73.58**	**71.26**
LS-ViT (Hwang et al., 2026)	2/4	58.48	74.91	65.66	73.78	80.66	73.89	74.95	71.76
+ FLOW-Q	2/4	**60.44**	**75.93**	**66.50**	**74.51**	**81.49**	**74.95**	**76.10**	**72.81**

Table 4: Comparison (§4.1) with SOTA methods across lightweight networks on ImageNet in terms of Top-1 accuracy, including MobileNetV2 (1.0× / 1.4×) and MobileViT (XS / S) at two scales.

Method	W3A3		W3A4		W4A4	
	MobileNetV2	MobileViT	MobileNetV2	MobileViT	MobileNetV2	MobileViT
Full-Precision	72.94 / 76.53	74.63 / 78.40	72.94 / 76.53	74.63 / 78.40	72.94 / 76.53	74.63 / 78.40
BRECQ (Li et al., 2021)	12.57 / 28.59	12.43 / 40.39	12.65 / 28.78	20.13 / 49.36	62.92 / 72.18	65.32 / 73.51
QDrop (Wei et al., 2022)	60.49 / 69.48	52.43 / 64.15	60.81 / 69.70	58.08 / 68.57	68.07 / 74.63	70.21 / 75.59
FIMA-Q (Wu et al., 2025a)	63.86 / 69.85	61.56 / 70.59	64.17 / 70.37	66.13 / 72.07	70.94 / 74.95	71.15 / 76.32
TWEO (Liang et al., 2026)	65.35 / 71.06	59.62 / 69.91	65.59 / 71.15	64.33 / 70.83	71.57 / 75.39	71.20 / 76.07
APHQ-ViT (Wu et al., 2025b)	65.14 / 70.74	60.91 / 70.23	65.42 / 71.08	65.36 / 71.49	71.45 / 75.36	71.79 / 76.22
+ FLOW-Q	**67.27 / 72.39**	**63.66 / 71.38**	**67.75 / 72.84**	**68.02 / 73.75**	**71.79 / 75.69**	**72.41 / 76.68**
LS-ViT (Hwang et al., 2026)	66.70 / 70.12	61.64 / 70.74	66.98 / 70.72	66.44 / 72.64	71.29 / 75.04	72.04 / 76.39
+ FLOW-Q	**67.51 / 72.44**	**62.31 / 71.20**	**67.77 / 72.59**	**67.25 / 73.42**	**71.72 / 75.65**	**72.21 / 76.76**

on the CNN-based MobileNetV2 (Sandler et al., 2018). For instance, when integrated with APHQ-ViT (Wu et al., 2025b), FLOW-Q improves the average performance of MobileNetV2 variants by **1.89%**, **2.05%**, and **0.34%** under W3A3, W3A4, and W4A4, respectively. When integrated with LS-ViT (Hwang et al., 2026), the corresponding average gains are **1.57%**, **1.33%**, and **0.52%**. These results highlight the strong adaptability of FLOW-Q to the quantization of efficient models.

4.2 DIAGNOSTIC EXPERIMENTS

This section presents the ablation studies in detail, using LS-ViT (Hwang et al., 2026) as the baseline.

Component Efficiency Analysis. We analyze the components of FLOW-Q in Tables 5 and 6. Both the evolution-aware learning (EAL) strategy and geometric constraint rectification (GCR) module provide clear gains individually, while GCR further improves performance based on the manifold representation learned by EAL. Combined, they yield a **2.23%** improvement with only 12.63% more training time and 1.06 GB additional GPU memory, demonstrating substantial and consistent gains with **acceptable** computational overhead across different evaluation settings.

Subspace Source Analysis. We compare different matrix choices for SVD in Table 7. Applying SVD to either the Hessian matrix H or the feature evolution matrix ΔX^{FP} yields notable gains, indicating that subspace calibration suppresses noisy information. When applied to the manifold representation constructed via Eq. 2, the model achieves the best performance, improving the three video

Table 5: Component efficiency analysis (§4.2) on ViT-S under W2A3 quantization setting using a single RTX 4090 GPU.

Baseline	EAL	GCR	Accuracy	Time (min)	GPU (GB)
✓			42.05	105.3	3.87
✓	✓		42.75	106.1	4.04
✓		✓	42.92	115.9	4.78
✓	✓	✓	**44.28**	118.6	4.93

Table 6: Component efficiency analysis (§4.2) of geometric constraint rectification on ViT-S under W2A3 quantization setting using a single RTX 4090 GPU.

ESL	Direction $\mathcal{L}_{direct}$	Magnitude $\mathcal{L}_{magn}$	Accuracy	Time (min)	GPU (GB)
✓			42.75	106.1	4.04
✓	✓		43.69	113.5	4.61
✓		✓	43.16	111.6	4.45
✓	✓	✓	**44.28**	118.6	4.93

Table 7: Subspace source analysis (§4.2) on the quantized DINOv3 under W3A4 setting.

Setting	Video Object Segmentation
Baseline	48.03 / 47.45 / 48.61
+ SVD to Hessian	48.54 / 48.11 / 48.96
+ SVD to Feature Evolution	48.79 / 48.10 / 49.48
+ SVD to Manifold Rep.	**49.70 / 49.41 / 49.99**

Table 8: Rectification strategy analysis (§4.2) on the quantized DINOv3 under the W3A4 setting.

Setting	Semantic Matching
Hessian for MSE	85.02 / 66.67 / 51.43
Hessian for GCR	85.41 / 67.24 / 52.46
Manifold Rep. for MSE	85.39 / 67.33 / 52.50
Manifold Rep. for GCR	**85.68 / 68.01 / 52.90**

Table 9: Evolution analysis (§4.2) on the quantized DINOv3 under the W4A4 setting for semantic matching.

Setting	Semantic Matching
Baseline	85.38 / 67.87 / 54.71
+ $(X^{FP} - X^{Q})$	85.61 / 68.39 / 55.05
+ $(X_i^{FP} - X_{i-1}^{FP})$ in Eq. 1	**86.10 / 69.52 / 55.68**

Table 10: Parameter analysis (§4.2) of truncation rank u on DeiT-S under different settings. CEVR denotes the cumulative explained variance ratio.

μ	128	144	192	256	300	360
CEVR	94.35	95.80	98.48	99.62	99.85	99.99
Time (ms)	19.01	22.22	28.83	36.10	42.94	47.46
W2A3 (Cls.)	54.61	54.70	54.89	54.92	54.93	54.93
W2A4 (Cls.)	66.30	66.43	66.50	66.51	66.52	66.53

object segmentation metrics by **1.67%**, **1.96%**, and **1.38%**, respectively. These results demonstrate the superiority of the rich contextual structural information encoded in the manifold representation.

Rectification Strategy Analysis. Table 8 compares MSE with geometric constraint rectification (GCR) using information derived from either Hessian sensitivity or the manifold representation. The results demonstrate that GCR consistently outperforms MSE across all settings. In particular, GCR combined with the manifold representation achieves substantial improvements over Hessian-weighted MSE, with gains of **0.66%**, **1.34%**, and **1.47%** on the three benchmarks, respectively.

Evolution Analysis. We investigate the source of feature evolution information in evolution-aware learning, with the results reported in Table 9. Quantized representations are less reliable, and their deviations from full-precision features may not faithfully capture the contextual information learned by the block, leading to marginal gains. In contrast, the input and output representations of the full-precision model remain robust under ideal conditions, making their difference a more reliable estimate of block-wise contextual information and yielding larger performance improvements.

Parameter Analysis. Table 10 analyzes the truncation rank μ in Eq. 3 in terms of cumulative explained variance ratio (CEVR), singular value decomposition time, and performance. CEVR measures the information preserved from the original matrix, while Time denotes the runtime of low-rank SVD on a 384×384 matrix. Increasing μ from 128 to 192 substantially improves CEVR and performance with only 9.8 ms additional runtime. Beyond 192, however, performance gains become marginal while latency increases considerably. These experimental results show that $\mu = 192$ achieves a **trade-off** among information preservation, computational efficiency, and performance.

5 Conclusion

In this paper, we propose FLOW-Q to alleviate the degraded robustness of visual representations in quantized models caused by the prevailing sensitivity-weighted reconstruction paradigm. Specifically, we design an evolution-aware learning strategy that transforms discrete local responses into structured second-order correlations, yielding a manifold representation that provides guidance for rectification of quantized features. Moreover, we propose a geometric constraint rectification module that leverages the rich structural information encoded in the manifold representation to rectify quantized features in terms of both direction and magnitude, thereby enabling them to inherit the robust visual representations of the full-precision model. Extensive experiments on nine tasks and twelve datasets demonstrate the strong generalizability of FLOW-Q, which consistently achieves SOTA quantization performance across DINOv3, conventional ViTs, and mobile-efficient models.

6 AI USE STATEMENT

We confirm that LLMs were used solely for minor grammatical corrections and phrasing suggestions. They were not involved in providing research ideas, including motivation, algorithm design, or the development of the core method. Furthermore, they were not used in generating any scientific content, such as the introduction, methodology, or experimental results presented in this paper.

7 REPRODUCIBILITY STATEMENT

We provide reproducible PyTorch-based code in the supplementary material. All experiments are conducted using publicly available vision models and datasets. The main paper provides the relevant implementation details, computational overhead, and evaluation settings for reproduction, while additional reproducibility details are included in the supplementary material.

REFERENCES

Jonathan T Barron and Ben Poole. The fast bilateral solver. In *ECCV*, pp. 617–632, 2016.

Xinhao Cai, Liulei Li, Gensheng Pei, Tao Chen, Jinshan Pan, Yazhou Yao, and Wenguan Wang. Unbiased object detection beyond frequency with visually prompted image synthesis. In *ICLR*, pp. 37332–37355, 2026.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, pp. 9630–9640, 2021.

Lei Chen, Yuan Meng, Chen Tang, Xinzhu Ma, Jingyan Jiang, Xin Wang, Zhi Wang, and Wenwu Zhu. Q-dit: Accurate post-training quantization for diffusion transformers. In *CVPR*, pp. 28306–28315, 2025.

Joonmyung Choi, Sanghyeok Lee, Byungoh Ko, Eunseo Kim, Jihyung Kil, and Hyunwoo J Kim. Representation shift: Unifying token compression with flashattention. In *ICCV*, pp. 20456–20466, 2025.

Juncan Deng, Shuaiting Li, Zeyu Wang, Hong Gu, Kedong Xu, and Kejie Huang. Vq4dit: Efficient post-training vector quantization for diffusion transformers. In *AAAI*, volume 39, pp. 16226–16234, 2025.

Xin Ding, Xiaoyu Liu, Zhijun Tu, Yun Zhang, Wei Li, Jie Hu, Hanting Chen, Yehui Tang, Zhiwei Xiong, Baoqun Yin, et al. Cbq: Cross-block quantization for large language models. In *ICLR*, pp. 7056–7075, 2025.

Yifu Ding, Weilun Feng, Chuyan Chen, Jinyang Guo, and Xianglong Liu. Reg-ptq: Regression-specialized post-training quantization for fully quantized object detector. In *CVPR*, pp. 16174–16184, 2024.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

Ali Edalati, Alireza Ghaffari, Mahsa Ghazvini Nejad, Lu Hou, Boxing Chen, Masoud Asgharian, and Vahid Partovi Nia. Oac: Output-adaptive calibration for accurate post-training quantization. In *AAAI*, volume 39, pp. 16453–16461, 2025.

Vage Egiazarian, Roberto Castro, Denis Kuznedelev, Andrei Panferov, Eldar Kurtic, Shubhra Pandit, Alexandre Marques, Mark Kurtz, Saleh Ashkboos, Torsten Hoefler, et al. Bridging the gap between promise and performance for microscaling fp4 quantization. In *ICLR*, pp. 113529–113563, 2026.

Mohamed El Banani, Amit Raj, Kevis-Kokitsi Maninis, Abhishek Kar, Yuanzhen Li, Michael Rubinstein, Deqing Sun, Leonidas Guibas, Justin Johnson, and Varun Jampani. Probing the 3d awareness of visual foundation models. In *CVPR*, pp. 21795–21806, 2024.

Mohsen Gholami, Mohammad Akbari, Kevin Cannons, and Yong Zhang. Casp: Compression of large multimodal models based on attention sparsity. In *CVPR*, pp. 9372–9381, 2025.

Yuxian Gu, Li Dong, Furu Wei, and Minlie Huang. Minillm: Knowledge distillation of large language models. In *ICLR*, pp. 32694–32717, 2024.

Janek Haberer, Ali Hojjat, and Olaf Landsiedel. Hydravit: Stacking heads for a scalable vit. In *NeurIPS*, pp. 40254–40277, 2024.

Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *ICCV*, pp. 991–998, 2011.

Yefei He, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. EfficientDM: Efficient quantization-aware fine-tuning of low-bit diffusion models. In *ICLR*, 2024.

Wei Huang, Yi Ge, Shuai Yang, Yicheng Xiao, Huizi Mao, Yujun Lin, Hanrong Ye, Sifei Liu, Ka Chun Cheung, Hongxu Danny Yin, et al. Qerl: Beyond efficiency-quantization-enhanced reinforcement learning for llms. In *ICLR*, pp. 99338–99361, 2026.

Hyunha Hwang, Xuan Truong Nguyen, and Hyuk-Jae Lee. Ls-vit: Least-squares hessian based block reconstruction for low-bit post-training quantization of vision transformers. In *CVPR*, pp. 33588–33597, 2026.

Varun Jampani, Kevis-Kokitsi Maninis, Andreas Engelhardt, Arjun Karpur, Karen Truong, Kyle Sargent, Stefan Popov, André Araujo, Ricardo Martin Brualla, Kaushal Patel, et al. Navi: Category-agnostic image collections with high-quality 3d shape and pose annotations. In *NeurIPS*, pp. 76061–76084, 2023.

Runqing Jiang, Ye Zhang, Longguang Wang, Pengpeng Yu, and Yulan Guo. Aiqvit: Architecture-informed post-training quantization for vision transformers. In *AAAI*, volume 39, pp. 17635–17643, 2025.

Cheng Lei, Ao Li, Hu Yao, Ce Zhu, and Le Zhang. Rethinking token reduction with parameter-efficient fine-tuning in vit for pixel-level tasks. In *CVPR*, pp. 14954–14964, 2025.

Changhao Li, Xinrui Chen, Ji Wang, Kang Zhao, and Jianfei Chen. Task-specific zero-shot quantization-aware training for object detection. In *ICCV*, pp. 22868–22878, 2025a.

Liulei Li, Yi Yang, and Wenguan Wang. Reconciling visual perception and generation in diffusion models. In *ICLR*, 2026.

Mingcheng Li, Dingkang Yang, Xiao Zhao, Shuaibing Wang, Yan Wang, Kun Yang, Mingyang Sun, Dongliang Kou, Ziyun Qian, and Lihua Zhang. Correlation-decoupled knowledge distillation for multimodal sentiment analysis with incomplete modalities. In *CVPR*, pp. 12458–12468, 2024.

Yuhang Li, Ruihao Gong, Xu Tan, Yang Yang, Peng Hu, Qi Zhang, Fengwei Yu, Wei Wang, and Shi Gu. Brecq: Pushing the limit of post-training quantization by block reconstruction. In *ICLR*, 2021.

Yuqi Li, Chuanguang Yang, Hansheng Zeng, Zeyu Dong, Zhulin An, Yongjun Xu, Yingli Tian, and Hao Wu. Frequency-aligned knowledge distillation for lightweight spatiotemporal forecasting. In *ICCV*, pp. 7262–7272, 2025b.

Guang Liang, Jie Shao, Ningyuan Tang, Xinyao Liu, and Jianxin Wu. Tweo: Transformers without extreme outliers enables fp8 training and quantization for dummies. In *CVPR*, pp. 6095–6105, 2026.

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. In *ECCV*, pp. 740–755, 2014a.

Yen-Liang Lin, Vlad I Morariu, Winston Hsu, and Larry S Davis. Jointly optimizing 3d model fitting and fine-grained classification. In *ECCV*, pp. 466–480, 2014b.

Xuewen Liu, Zhikai Li, Jing Zhang, Mengjuan Chen, Jianquan Li, and Qingyi Gu. Ptq4arvg: Post-training quantization for autoregressive visual generation models. In *ICLR*, pp. 80163–80187, 2026.

Yijiang Liu, Huanrui Yang, Zhen Dong, Kurt Keutzer, Li Du, and Shanghang Zhang. Noisyquant: Noisy bias-enhanced post-training activation quantization for vision transformers. In *CVPR*, pp. 20321–20330, 2023.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, pp. 10012–10022, 2021.

Yuexiao Ma, Huixia Li, Xiawu Zheng, Feng Ling, Xuefeng Xiao, Rui Wang, Shilei Wen, Fei Chao, and Rongrong Ji. Outlier-aware slicing for post-training quantization in vision transformer. In *ICML*, pp. 33811–33825, 2024.

Sachin Mehta and Mohammad Rastegari. Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer. In *ICLR*, 2022.

Luca Morreale, Gil CP Ramos, Malcolm Chadwick, Mehdi Noroozi, Ruchika Chavhan, Abhinav Mehrotra, et al. Fraqat: Quantization aware training with fractional bits. In *CVPR*, pp. 8514–8523, 2026.

Saurav Muralidharan, Sharath Turuvekere Sreenivas, Raviraj Joshi, Marcin Chochowski, Mostofa Patwary, Mohammad Shoeybi, Bryan Catanzaro, Jan Kautz, and Pavlo Molchanov. Compact language models via pruning and knowledge distillation. In *NeurIPS*, pp. 41076–41102, 2024.

Li Pang, Xiangyu Rui, Long Cui, Hongzhong Wang, Deyu Meng, and Xiangyong Cao. Hir-diff: Unsupervised hyperspectral image restoration via improved diffusion models. In *CVPR*, pp. 3005–3014, 2024.

Jaewoo Park, Jacky Chen Long Chai, Jaeho Yoon, and Andrew Beng Jin Teoh. Understanding the feature norm for out-of-distribution detection. In *ICCV*, pp. 1557–1567, 2023.

Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

Filip Radenović, Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondřej Chum. Revisiting oxford and paris: Large-scale image retrieval benchmarking. In *CVPR*, pp. 5706–5715, 2018.

Akshat Ramachandran, Mingyu Lee, Huan Xu, Souvik Kundu, and Tushar Krishna. Ouromamba: A data-free quantization framework for vision mamba. In *ICCV*, pp. 21177–21186, 2025.

Michael Rubinstein, Armand Joulin, Johannes Kopf, and Ce Liu. Unsupervised joint object discovery and segmentation in internet images. In *CVPR*, pp. 1939–1946, 2013.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015.

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*, pp. 4510–4520, 2018.

Xuan Shen, Weize Ma, Jing Liu, Changdi Yang, Rui Ding, Quanyi Wang, Henghui Ding, Wei Niu, Yanzhi Wang, Pu Zhao, et al. Quartdepth: Post-training quantization for real-time depth estimation on the edge. In *CVPR*, pp. 11448–11460, 2025.

Jianping Shi, Qiong Yan, Li Xu, and Jiaya Jia. Hierarchical image saliency detection on extended cssd. *IEEE TPAMI*, 38(4):717–729, 2015.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, pp. 746–760, 2012.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Walter Simoncini, Michael Dorkenwald, Tijmen Blankevoort, Cees Snoek, and Yuki Asano. Elastic vits from pretrained models without retraining. In *NeurIPS*, pp. 24384–24414, 2025.

Shangquan Sun, Wenqi Ren, Jingzhi Li, Rui Wang, and Xiaochun Cao. Logit standardization in knowledge distillation. In *CVPR*, pp. 15731–15740, 2024.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, pp. 10347–10357, 2021.

Hailing Wang, Jianglin Lu, Yitian Zhang, and Yun Fu. Outlier-aware post-training quantization for image super-resolution. In *ICCV*, pp. 16175–16184, 2025a.

Lijun Wang, Huchuan Lu, Yifan Wang, Mengyang Feng, Dong Wang, Baocai Yin, and Xiang Ruan. Learning to detect salient objects with image-level supervision. In *CVPR*, pp. 3796–3805, 2017.

Weitian Wang, Rai Shubham, Cecilia De La Parra, and Akash Kumar. Mixa-q: Revisiting activation sparsity for vision transformers from a mixed-precision quantization perspective. In *ICCV*, pp. 22143–22152, 2025b.

Yangtao Wang, Xi Shen, Shell Xu Hu, Yuan Yuan, James L Crowley, and Dominique Vaufreydaz. Self-supervised transformers for unsupervised object discovery using normalized cut. In *CVPR*, pp. 14523–14533, 2022.

Xiuying Wei, Ruihao Gong, Yuhang Li, Xianglong Liu, and Fengwei Yu. QDrop: Randomly dropping quantization for extremely low-bit post-training quantization. In *ICLR*, 2022.

Junyi Wu, Haoxuan Wang, Yuzhang Shang, Mubarak Shah, and Yan Yan. Ptq4dit: Post-training quantization for diffusion transformers. In *NeurIPS*, pp. 62732–62755, 2024a.

Zhuguanyu Wu, Jiaxin Chen, Hanwen Zhong, Di Huang, and Yunhong Wang. Adalog: Post-training quantization for vision transformers with adaptive logarithm quantizer. In *ECCV*, pp. 411–427, 2024b.

Zhuguanyu Wu, Shihe Wang, Jiayi Zhang, Jiaxin Chen, and Yunhong Wang. Fima-q: Post-training quantization for vision transformers by fisher information matrix approximation. In *CVPR*, pp. 14891–14900, 2025a.

Zhuguanyu Wu, Jiayi Zhang, Jiaxin Chen, Jinyang Guo, Di Huang, and Yunhong Wang. Aphq-vit: Post-training quantization with average perturbation hessian based reconstruction for vision transformers. In *CVPR*, pp. 9686–9695, 2025b.

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, pp. 3485–3492, 2010.

Jiawei Xu, Yi Zheng, Chenghe Sun, Taiyu Zhou, Zuqi Zhang, Jie Li, Lirong Zheng, and Zhuo Zou. LogART: Pushing the limit of efficient logarithmic post-training quantization. In *ICLR*, 2026.

Ke Xu, Zhongcheng Li, Shanshan Wang, and Xingyi Zhang. Ptmq: Post-training multi-bit quantization of neural networks. In *AAAI*, volume 38, pp. 16193–16201, 2024.

Jianjian Yin, Tao Chen, Gensheng Pei, Huafeng Liu, Yazhou Yao, Liqiang Nie, and Xiansheng Hua. Semi-supervised semantic segmentation with multi-constraint consistency learning. *IEEE TMM*, 27:6449–6461, 2025.

Aozhong Zhang, Naigang Wang, Yanxia Deng, Xin Li, Zi Yang, and Penghang Yin. Magr: Weight magnitude reduction for enhancing post-training quantization. In *NeurIPS*, pp. 85109–85130, 2024.

Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. In *NeurIPS*, pp. 45533–45547, 2023.

Yunshan Zhong, You Huang, Jiawei Hu, Yuxin Zhang, and Rongrong Ji. Towards accurate post-training quantization of vision transformers via error reduction. *IEEE TPAMI*, 47(4):2676–2692, 2025.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, pp. 633–641, 2017.

Chen Zhu, Wangbo Zhao, Huiwen Zhang, Yuhao Zhou, Weidong Tang, Shuo Wang, Zhihang Yuan, Yuzhang Shang, Xiaojiang Peng, Kai Wang, et al. Ea-vit: Efficient adaptation for elastic vision transformer. In *ICCV*, pp. 1–10, 2025.

A APPENDIX

A.1 BASIS INVARIANCE PROOF OF FLOW-Q

Theoretically, the basis E derived from SVD in Eq. 3 is not unique. However, we show that such non-uniqueness does not compromise the effectiveness of Eqs. 5 and 6, as proved below.

Theorem. ***(Basis Invariance).*** The directional and magnitude matching losses ($\mathcal{L}_{direct}$ and $\mathcal{L}_{magn}$) depend solely on the subspace spanned by the basis, and they are invariant under arbitrary orthogonal transformations of the basis E.

Proof. Any two bases of the subspace are related by an orthogonal transformation. Accordingly, any orthonormal basis $\widetilde{E}$ spanning the same subspace as E can be expressed as $\widetilde{E} = ER$, where R is an appropriate orthogonal matrix satisfying $RR^\top = R^\top R = I$. We next examine the invariance of both $\mathcal{L}_{direct}$ and $\mathcal{L}_{magn}$ from the perspectives of the matrix trace and the Frobenius norm.

Matrix Trace. When $\widetilde{E}$ is adopted as the basis, Eq. 4 yields the updated principal-component representations $\widetilde{M}^Q$ and $\widetilde{M}^{FP}$ for the two models, respectively. The trace term of the directional loss $\mathcal{L}_{direct}$ can be equivalently reformulated as:

$$Tr((\widetilde{M}^Q)^\top \widetilde{M}^{FP}) = Tr((M^Q R)^\top (M^{FP} R)). \tag{7}$$

To provide a more direct and transparent justification, we establish the above equality by expressing the trace as the sum of the inner products between corresponding row vectors:

$$\begin{aligned} Tr((M^Q R)^\top (M^{FP} R)) &= \textstyle\sum_i^N (m_i^Q R)(m_i^{FP} R)^\top \\ &= \textstyle\sum_i^N m_i^Q (RR^\top)(m_i^{FP})^\top = Tr((M^Q)^\top M^{FP}). \end{aligned} \tag{8}$$

As shown in Eq. 7 and 8, the trace term remains unchanged under any orthogonal transformation applied to the basis, indicating that it is determined solely by the underlying subspace.

Frobenius Norm. For notational simplicity, M is used to uniformly represent M^Q and M^{FP}, and $\widetilde{M}$ for $\widetilde{M}^Q$ and $\widetilde{M}^{FP}$, with the relation $\widetilde{M} = MR$. The Frobenius norm of $\widetilde{M}$ can be written in terms of the matrix trace as:

$$||\widetilde{M}||_F = ||MR||_F = \sqrt{Tr((MR)^\top (MR))} = \sqrt{Tr(M^\top M)} = ||M||_F. \tag{9}$$

The intermediate equality follows from Eq. 8. We have thus proved that $||\widetilde{M}^Q||_F = ||M^Q||_F$ and $||\widetilde{M}^{FP}||_F = ||M^{FP}||_F$. This establishes the invariance of the Frobenius norm with respect to the choice of basis. The basis invariance of the matrix trace and Frobenius norm ensures that the directional and magnitude losses remain effective and stable under different orthonormal bases, enabling precise geometric rectification of quantized features. In this way, FLOW-Q consistently encourages the quantized model to inherit the robustness of feature representations from its full-precision counterpart. □

A.2 DISCUSSION AND FUTURE WORK

Impact. The impact of our work can be summarized in three aspects. **First**, FLOW-Q reveals that the sensitivity-weighted reconstruction adopted by existing PTQ methods is insufficient to preserve strong feature representation capability in quantized models. **Second**, our method introduces a new learning perspective based on feature evolution. Rather than focusing solely on how sensitive individual positions are to quantization errors, it emphasizes how feature representations evolve across layers. This perspective extends the prevailing PTQ learning paradigm from pure error minimization toward structural and geometric preservation. **Third**, FLOW-Q demonstrates strong generality, consistently improving performance across diverse vision architectures and downstream tasks. *These results suggest that evolution-aware structural modeling has the potential to serve as a general design principle for preserving the representation capability of quantized vision models.*

Limitation. Despite achieving superior quantization performance across diverse vision architectures and downstream tasks compared with existing PTQ methods (Liang et al., 2026; Hwang et al., 2026),

the computational efficiency of FLOW-Q can be further improved. According to the component-wise efficiency analysis in Tables 5 and 6, the evolution-aware learning strategy increases training time by only 0.8 minutes and GPU memory usage by 0.17 GB. In contrast, the computational cost of FLOW-Q is dominated by the geometric constraint rectification module, which increases training time by 10.6 minutes and GPU memory consumption by 0.91 GB. This cost is closely related to the choice of the truncated rank μ, as shown in Table 10. A larger μ increases the computational time required for truncated singular value decomposition, while also enlarging the dimensionality of the evolution mapping in Eq. 4 and the subsequent geometric rectification in Eqs. 5 and 6, leading to higher overall computational and memory costs.

Future Work. Different blocks may exhibit distinct spectral characteristics of feature evolution, suggesting that the same subspace dimension may not be necessary for every layer. An interesting direction is therefore to adaptively determine a more compact μ for each block according to its spectral distribution, potentially reducing redundant computation and memory consumption while maintaining competitive quantization performance.

A.3 MORE EXPERIMENTAL DETAILS

A.3.1 DATASET

The datasets corresponding to each task in the paper are as follows:

- **FG3DCar** (Lin et al., 2014b) contains 195 pairs of automobile images and is used to evaluate semantic matching across different car instances.
- **JODS** (Rubinstein et al., 2013) is a standard semantic matching dataset comprising 81 image pairs from three object categories, namely airplanes, horses, and cars.
- **PASCAL** (Hariharan et al., 2011) contains 124 image pairs for semantic matching across five object categories, including bicycle, motorbike, bus, car, and train.
- **Oxford** (Radenović et al., 2018) is a landmark-based image retrieval benchmark containing 5k database images and 70 query images.
- **ECSSD** (Shi et al., 2015) is a standard salient object detection dataset containing 1k natural images with structurally complex salient objects and pixel-level annotations.
- **DUTS** (Wang et al., 2017) contains 5,019 test images collected from the ImageNet test set and the SUN dataset (Xiao et al., 2010) for salient object detection.
- **NAVI** (Jampani et al., 2023) is a standard multi-view correspondence dataset comprising approximately 29k images of 36 objects captured under diverse viewpoints and scene conditions, with accurate camera poses and high-quality 3D annotations.
- **DAVIS-2017** (Pont-Tuset et al., 2017) is a standard video object segmentation dataset comprising 60 training and 30 validation videos, with pixel-level annotations for multiple object instances.
- **COCO** (Lin et al., 2014a) is one of the largest benchmarks for object detection, with object distributions closely reflecting real-world scenes. It contains 118K training images and 5K validation images, making it well suited for evaluating the object recognition and localization capabilities of vision models.
- **NYUV2** (Silberman et al., 2012) is widely used for monocular depth estimation and contains 1,449 densely annotated RGB-D images. The dataset is split into 795 training images and 654 test images, and is commonly adopted to evaluate the ability of models to understand indoor geometric structures and spatial depth information.
- **ADE20K** (Zhou et al., 2017) serves as a large-scale benchmark for semantic segmentation, covering 150 semantic categories. It contains approximately 25k images, split into 20k for training, 2k for validation, and 3k for testing, and is widely used to evaluate semantic understanding in complex scenes.
- **ImageNet** (Russakovsky et al., 2015) is a large-scale image classification dataset containing 1k categories, with a training/validation/test split of 1.2M/50k/100k images.

The above provides detailed descriptions of the datasets used in our experiments.

A.3.2 EXPERIMENTAL SETUP

We evaluate our method across nine tasks. For DINOv3 (Siméoni et al., 2025) quantization, the evaluation includes five training-free tasks and three training-based tasks, covering a diverse range of downstream scenarios. For the quantization of conventional ViTs and mobile-efficient models, image classification is employed as the benchmark for performance evaluation. Detailed descriptions of all tasks are provided below.

First, we introduce the experimental settings for the five training-free tasks:

- **Semantic Matching**: We follow the evaluation protocol of SD (Zhang et al., 2023) on FG3DCar (Lin et al., 2014b), JODS (Rubinstein et al., 2013), and PASCAL (Hariharan et al., 2011). We perform zero-shot nearest-neighbor matching directly on the extracted feature maps without task-specific training. Following the TSS protocol, PCK@0.05 is reported for each dataset and their average, where the PCK threshold is normalized by the image dimensions.
- **Image Retrieval**: Following the image retrieval protocol of DINO (Caron et al., 2021), we evaluate on the Oxford (Radenović et al., 2018) dataset and report mAP under the Medium protocol.
- **Saliency Detection**: We follow the evaluation protocol of TokenCut (Wang et al., 2022) and perform unsupervised foreground segmentation using the extracted patch features. The patch features are used to construct a similarity graph, which is partitioned by Normalized Cut to obtain the salient object mask without any additional post-processing, such as the Bilateral Solver (Barron & Poole, 2016). We evaluate on ECSSD (Shi et al., 2015) and DUTS (Wang et al., 2017) using the standard saliency detection metrics.
- **Multi-view Correspondence**: The evaluation protocol of Probe3D (El Banani et al., 2024) is adopted on the NAVI (Jampani et al., 2023) dataset . Correspondences are established by nearest-neighbor matching between dense features. Average recall is reported under different viewpoint changes.
- **Video Object Segmentation**: Following DINO (Caron et al., 2021), we use the first frame and the preceding seven frames as references, and propagate labels based on the top-5 nearest neighbors within a local spatial neighborhood. All input frames are resized such that the shorter side is 480 pixels.

Next, the experimental settings for the three training-based tasks are described as follows:

- **Object Detection**: The learning rate is set to 5e-4, the batch size to 16, and the model is trained for 72 epochs.
- **Depth Estimation**: We train the model for 60 epochs using a batch size of 16 and a learning rate of 1e-6.
- **Semantic Segmentation**: The learning rate is set to 1e-3, with training conducted for 100 epochs using a batch size of 16.

Finally, we present the experimental setup for the image classification task on ImageNet:

- **Image Classification**: Each block is trained for 20k iterations with a batch size of 32, while the weight and activation learning rates are set to 1e-3 and 4e-5, respectively.

The above describes the experimental settings for all tasks considered in this work.

A.4 MORE QUALITATIVE COMPARISONS

Figure S1 presents a qualitative comparison of heatmaps produced by DINOv3 (Siméoni et al., 2025) quantized with different methods. The results demonstrate the superiority of the proposed FLOW-Q. For example, for the sheep and squirrel in the first row, the cat and building in the second row, and the flower and signboard in the third row, FLOW-Q captures the target regions more comprehensively and precisely than LS-ViT (Hwang et al., 2026), while producing activation patterns that are closer to those of the original full-precision DINOv3. These results indicate that FLOW-Q better enables the quantized model to inherit the robust visual representations of the original model.

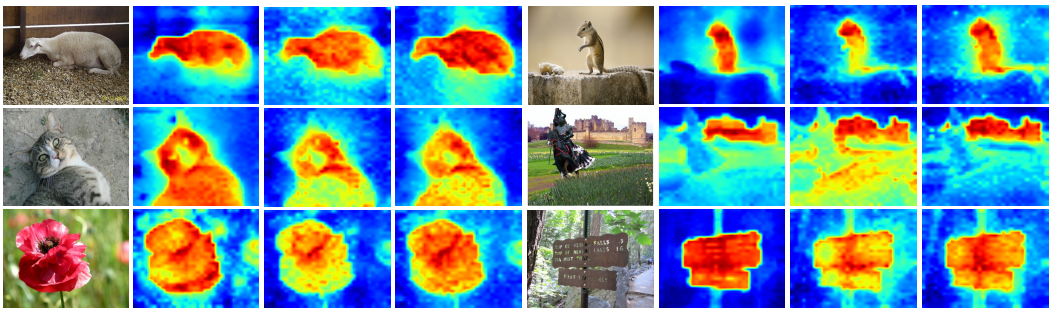

Figure S1: More visualization results (§$A.4$) for representation analysis.

Table S1: Comparison (§$A.5$) of the Top-1 accuracy on ImageNet (Russakovsky et al., 2015).

Method	W/A	ViT-S	ViT-B	DeiT-S	DeiT-B	DeiT-L	Swin-S	Swin-B	Avg.
Full-Precision	32/32	81.39	84.54	79.85	81.80	86.91	83.23	85.27	83.28
BRECQ (Li et al., 2021)	2/2	3.58	9.43	6.48	24.11	46.97	11.12	23.16	17.84
QDrop (Wei et al., 2022)	2/2	5.23	16.16	5.78	25.63	48.30	22.54	25.30	21.28
AdaLog (Wu et al., 2024b)	2/2	7.91	9.88	14.9	27.05	57.04	13.66	22.52	19.98
FIMA-Q (Wu et al., 2025a)	2/2	12.24	27.52	28.22	45.79	62.96	29.53	36.62	34.69
TWEO (Liang et al., 2026)	2/2	11.23	26.80	29.06	45.65	64.87	32.38	37.95	35.42
APHQ-ViT (Wu et al., 2025b)	2/2	10.86	26.38	28.52	45.01	64.13	30.97	37.10	34.71
+ FLOW-Q	2/2	**11.78**	**27.05**	**30.04**	**46.12**	**65.80**	**31.91**	**38.22**	**35.84**
LS-ViT (Hwang et al., 2026)	2/2	14.64	29.71	28.39	46.29	63.92	40.11	43.19	38.03
+ FLOW-Q	2/2	**15.65**	**30.58**	**29.78**	**47.34**	**64.77**	**41.64**	**44.25**	**39.14**
BRECQ (Li et al., 2021)	3/3	54.33	66.62	63.72	72.82	82.36	75.20	77.64	70.38
QDrop (Wei et al., 2022)	3/3	57.05	74.75	50.95	72.97	82.40	74.67	76.57	67.62
AdaLog (Wu et al., 2024b)	3/3	62.50	76.73	63.24	75.44	83.12	76.01	77.97	73.57
FIMA-Q (Wu et al., 2025a)	3/3	64.09	77.63	69.13	76.54	83.39	77.26	78.82	75.27
TWEO (Liang et al., 2026)	3/3	63.59	76.65	69.00	76.62	83.47	76.15	78.19	74.81
APHQ-ViT (Wu et al., 2025b)	3/3	63.17	76.31	68.76	76.31	83.36	76.10	78.14	74.59
+ FLOW-Q	3/3	**64.31**	**76.84**	**69.37**	**77.03**	**83.91**	**76.61**	**78.73**	**75.25**
LS-ViT (Hwang et al., 2026)	3/3	64.10	77.65	69.41	76.57	83.49	77.39	79.40	75.43
+ FLOW-Q	3/3	**64.77**	**77.92**	**69.77**	**76.90**	**84.15**	**77.86**	**79.75**	**75.87**
QDrop (Wei et al., 2022)	4/4	71.84	82.63	72.64	79.96	83.89	81.21	82.99	79.31
OASQ (Ma et al., 2024)	4/4	72.88	76.59	76.00	78.83	-	81.02	82.46	-
AdaLog (Wu et al., 2024b)	4/4	72.75	79.68	72.06	78.03	85.37	80.77	82.47	78.73
FIMA-Q (Wu et al., 2025a)	4/4	76.68	83.04	76.87	80.33	85.61	81.82	83.60	81.14
TWEO (Liang et al., 2026)	4/4	76.01	82.46	76.45	80.11	85.72	81.34	83.50	80.80
APHQ-ViT (Wu et al., 2025b)	4/4	76.07	82.41	76.40	80.21	85.52	81.81	83.42	80.83
+ FLOW-Q	4/4	**76.69**	**82.53**	**76.92**	**80.46**	**86.01**	**81.96**	**83.60**	**81.17**
LS-ViT (Hwang et al., 2026)	4/4	76.61	82.68	76.69	80.23	85.64	81.75	83.53	81.02
+ FLOW-Q	4/4	**77.00**	**83.18**	**77.15**	**80.54**	**86.26**	**82.04**	**83.81**	**81.43**

A.5 MORE QUANTITATIVE RESULTS

Table S1 shows the performance gains achieved by integrating our method into the APHQ-ViT (Wu et al., 2025b) and LS-ViT (Hwang et al., 2026) frameworks under the W2A2, W3A3 and W4A4 settings for both weights and activations. The experimental results further demonstrate the superiority of our method across these quantization settings. For example, under the W2A2 setting, integrating FLOW-Q into LS-ViT (Hwang et al., 2026) yields performance gains of **1.01%**, **0.87%**, **1.39%**, **1.05%**, **0.85%**, **1.53%**, **1.06%** across ViT-S, ViT-B, DeiT-S, DeiT-B, DeiT-L, Swin-S, Swin-B, respectively. Similarly, under the W4A4 setting, consistent performance improvements are

observed across all corresponding architectures. The above experimental results, together with those reported in the main text under the W2A3 and W2A4 settings, demonstrate that the proposed method effectively preserves the robustness of visual representations in conventional ViTs.

A.6 PSEUDO CODE

To facilitate a better understanding of FLOW-Q, we provide pseudo code for the evolution-aware learning and geometric constraint rectification, as presented in Algorithms S1 and S2, respectively.

Algorithm S1 Pseudo-code for Evolution-aware Learning Strategy.

```
"""
FP_Model: full-precision model
Quant_Model: quantized model
Data_loader: a calibration set consisting of 1,024 images
"""

def Evolution_aware_Learning(FP_Model, Quant_Model, Data_loader):
    #--- Image-to-token mapping ---#
    for d in Data_loader:
        patch_fp, patch_q = FP_Model.patch_embed(d), Quant_Model.patch_embed(d)
        fp_pat.append(patch_fp)
        q_pat.append(patch_q)

    fp_In = torch.cat(fp_pat, dim=0)
    q_In = torch.cat(q_pat, dim=0)

    #--- Collect the manifold representation of each layer ---#
    for l in layers:
        #--- Compute the Hessian matrix ---#
        H = get_Hessian(FP_Model, Quant_Model, fp_In, q_In)
        X_FP, X_q = FP_Model.get_Block(l)(fp_In), Quant_Model.get_Block(l)(q_In)

        #--- Obtain feature evolution information (Eq. 1)---#
        delta_X_FP[l], delta_X_Q[l] = X_FP - fp_In, X_q - q_In

        #--- Generate the diversified matrix (Eq. 2)---#
        W = delta_X_FP[l] * torch.sqrt(torch.abs(H))

        #--- Derive the manifold representation of the current block (Eq. 2)---#
        G[l] = W.transpose(-1, -2) @ W

        fp_In, q_In = X_FP, X_q

    #--- Return the manifold representation and the FP and quantized feature evolution ---#
    return G, delta_X_FP, delta_X_Q
```

Algorithm S2 Pseudo-code for Geometric Constraint Rectification Module.

```
"""
delta_X_FP: full-precision feature evolution matrix
delta_X_Q: quantized feature evolution matrix
G: the manifold representation generated by the evolution-aware learning strategy
u: the rank used for truncated SVD
"""
L_direct, L_magn = 0, 0
def Geometric_Constraint_Rectification(delta_X_FP, delta_X_Q, G, u):
    #--- Perform block-wise reconstruction ---#
    for l in layers:
        #--- Obtain the subspace basis through singular value decomposition (Eq. 3)---#
        U, S, V = torch.svd_lowrank(G[l], q=u)
        E = V[:, :u]

        #--- Project the feature evolution representation onto the subspace (Eq. 4)---#
        M_q = delta_X_Q[l] @ E
        M_fp = delta_X_FP[l] @ E

        #--- Calculate the directional matching loss (Eq. 5)---#
        numerator = torch.trace(M_q.T @ M_fp)
        norm_q = torch.norm(M_q, p='fro')
        norm_fp = torch.norm(M_fp, p='fro')
        L_direct = L_direct + 1.0 - numerator / (norm_q * norm_fp)

        #--- Calculate the magnitude matching loss (Eq. 6)---#
        L_magn = L_magn + torch.log(norm_q / norm_fp).pow(2)

    #--- Return the loss for backpropagation ---#
    return L_direct + L_magn
```