ORFC: Orthogonal Reparameterization for Low-Bitrate ViT Feature Coding

Letian Zhang[1], Wenhan Yang[2], and Ling-Yu Duan[1,2,†]

[1] School of Computer Science, Peking University, Beijing, China
[2] Pengcheng Laboratory, Shenzhen, China
zhangletian@stu.pku.edu.cn,yangwh@pcl.ac.cn,
lingyu@pku.edu.cn

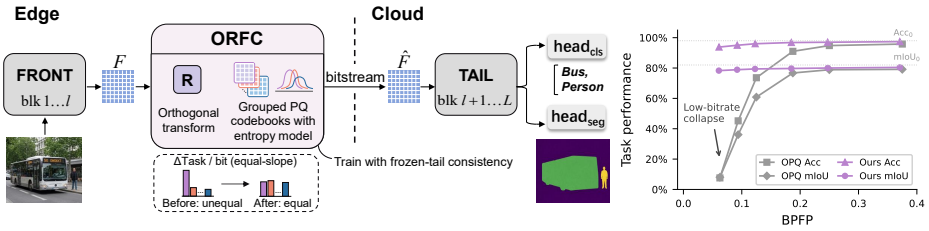
Abstract. Existing feature coding methods typically minimize feature-space reconstruction error, implicitly assuming that preserving feature similarity ensures downstream task performance. However, in Vision Transformers (ViTs), global correlation mixing makes downstream predictions more sensitive to small feature perturbations, so similar feature distortion can lead to significantly different task performance. In particular, our analysis reveals that under fixed-group product quantization, task sensitivity is highly uneven across channel subspaces. Consequently, near-uniform per-group rates may waste bits on insensitive directions while leaving critical ones underrepresented, causing rapid task-performance degradation at low bitrate. To address this issue, we propose an **O**rthogonal **R**eparameterized product quantization for intermediate **F**eature **C**oding (**ORFC**). It introduces a learnable orthogonal transform that rotates the representation before grouped quantization, enabling fixed-size groups to better align with task-relevant directions. We further use the output deviation of the remaining layers as a proxy distortion, and optimize an entropy-constrained rate-distortion objective to jointly learn the orthogonal transform and grouped quantization in an end-to-end manner. Extensive experiments across multiple ViT architectures and downstream tasks demonstrate that ORFC achieves a superior rate-task trade-off, especially in the low-bitrate regime. The code is available at `https://github.com/zhangletian2/ORFC`.

Keywords: Feature coding · Product quantization · Orthogonal reparameterization · Split inference

1 Introduction

Vision Transformers (ViTs) [2, 9, 33] have become a widely used backbone for vision understanding [31] and visual representation learning [40]. However, their large model size and computational cost still hinder deployment on resource-constrained edge devices [43]. Split inference [34, 45] addresses this challenge in edge-cloud collaborative settings by partitioning the model into an edge front

† Corresponding author.



(a) Overview of ORFC **(b)** RD performance

Fig. 1: ORFC overview and RD performance. (a) Intermediate feature coding for split ViT inference with grouped PQ and equal-slope intuition across groups. (b) RD curves on DINOv2 ViT-L/14 features (15^{th} block) for classification and segmentation, where our ORFC significantly outperforms OPQ and shows reduced collapse at low bitrate.

and a cloud tail. The edge executes the early layers to utilize on-device compute and transmits intermediate features to the cloud, where the heavier remaining computation is handled to complete inference. This design shifts the bottleneck from on-device computation to bandwidth-limited edge-cloud communication, making efficient transmission of intermediate features essential for practical split deployment of ViTs [47].

Feature coding [5, 6, 10, 17] addresses this communication bottleneck by compressing intermediate features into compact bitstreams. From a rate-distortion (RD) perspective [7], a feature codec is trained to reduce distortion under a bitrate constraint. In practice, most methods optimize a feature-space reconstruction loss, typically mean squared error (MSE), because it is differentiable and easy to train. This design is common in pipelines that reuse video codecs such as Versatile Video Coding (VVC) [4] as well as in neural compression models [3, 22]. While these approaches achieve small feature-space reconstruction error, they implicitly assume that preserving Euclidean feature similarity is sufficient to maintain the behavior of the remaining network.

However, when compressing ViT intermediate features, especially in the low-bitrate regime required by split inference, the assumption often breaks down. In this setting, reconstructed features are consumed by a machine vision tail rather than by humans, so the distortion should reflect downstream inference degradation. At low bitrate, ViT intermediate feature coding therefore faces two coupled issues that are easy to miss when only tracking reconstruction error in feature space. The first challenge is that feature-space distortion may not reflect task behavior. A small MSE on intermediate features can still lead to large deviations in the tail outputs, so reducing MSE does not necessarily preserve task performance. The second challenge is uneven sensitivity across subspaces. Different directions in the representation contribute differently to the task loss, so the tail may respond very differently to perturbations along different directions.

These issues become more pronounced under product quantization (PQ), which is widely used in feature coding. The high-dimensional feature is par-

titioned into multiple groups, and each group is quantized with a fixed codebook [11, 18, 24, 36]. This design imposes an implicit budget per group. RD optimality then calls for balancing the marginal task gain per bit across groups, which corresponds to the equal-slope condition [19]. However, optimizing feature-space MSE often balances reconstruction error across groups rather than task importance. As a result, bits may be allocated to less important groups while critical directions remain underrepresented. This mismatch appears as unequal ΔTask per bit in Fig. 1a and leads to the collapse at low bitrate in Fig. 1b.

To address this structural mismatch, we propose **ORFC**, an orthogonal reparameterized framework for low-bitrate ViT feature coding. The key idea is to learn a differentiable orthogonal transform that rotates the representation to redistribute task-sensitive directions across PQ groups, making the sensitivity across each group more balanced. Under grouped PQ with a fixed codebook size per group, this reparameterization better aligns the implicit per-group budgets with task importance, moving the codec closer to balanced marginal task gain across groups. To train the transform with a task-agnostic signal, we treat the frozen tail as a fixed evaluator and define a proxy distortion by measuring the deviation of its outputs after feature reconstruction. We then optimize an entropy-constrained RD objective based on this proxy distortion and jointly learn the orthogonal transform, grouped PQ codebooks, and group-specific entropy priors in an end-to-end manner. Through this joint optimization, ORFC reduces cross-group sensitivity imbalance and stabilizes inference in the low bitrate regime.

The contributions of this paper are summarized as follows:

- We show that feature-space MSE is misaligned with the task distortion at inference time for ViT intermediate features and leads to inefficient bit usage under product quantization at low bitrate.
- We propose ORFC, a low-bitrate ViT feature coding framework that learns an orthogonal reparameterization together with grouped PQ codebooks and entropy priors under an entropy-constrained RD objective with a frozen-tail output deviation proxy distortion.
- We achieve consistent RD gains across multiple ViTs and tasks compared with learning-based and VVC-based methods, substantially mitigating task loss under 0.1 BPFP.

2 Related Work

2.1 Feature Coding

Recent works on intermediate feature coding primarily explore general-purpose compression under bitrate constraints, with some recent variants incorporating task-aligned objectives. Early studies on CNN pipelines showed that intermediate tensors can be reshaped and compressed with conventional codecs or lightweight quantizers while keeping downstream accuracy at reasonable rates [5,

6]. Later, end-to-end trainable bottlenecks with learned entropy modeling became a standard approach in device-edge co-inference, demonstrating improved RD trade-offs by jointly optimizing the bottleneck and task objective [35, 41].

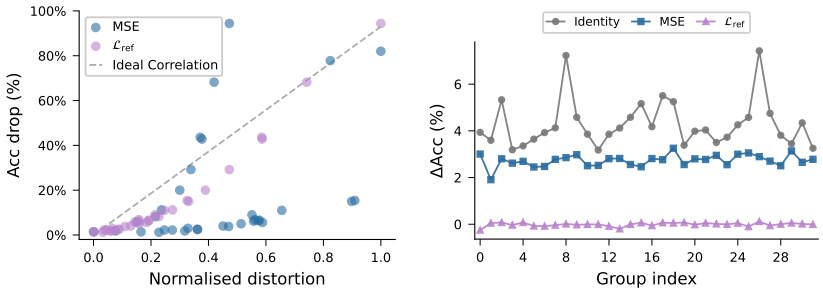
With the rise of foundation models, feature coding has been revisited for heterogeneous architectures and feature distributions. A large-model benchmark highlights the gap between off-the-shelf codecs and learned encoders [17], while universal schemes align formats and distributions to enable a single codec to generalize across different model families [15, 16]. Other work adapts distortion to semantics or task structure, including human-machine collaboration settings and identification-oriented objectives [14, 50, 51]. For ViT split inference at low bitrate, recent studies explore discrete tokenization and attention-guided communication [1, 52]. At a more principled level, RD theory in coding for machines [21] advocates task-aligned distortion measures, and related transform coding perspectives have appeared in KV-cache compression [46].

Despite this progress, many general-purpose feature codecs operate under fixed grouped structures and rely on generic feature-space reconstruction losses. When subspace sensitivities are highly uneven, such designs may fail to allocate bits in accordance with task importance. ORFC addresses this limitation by learning an orthogonal reparameterization that reshapes the representation before grouped quantization, and by optimizing an entropy-constrained RD objective with a frozen-tail output deviation proxy to better align grouped budgets with task sensitivity.

2.2 Product Quantization

Product quantization (PQ) is widely used as a compact representation tool beyond nearest neighbor search. Classical PQ factorizes the space into independent subspaces with separate codebooks [24], while OPQ and Cartesian k-means improve RD performance by learning data-dependent rotations or assignments [18, 37]. Deep and end-to-end variants integrate PQ into task networks, showing that supervised training and differentiable assignment can substantially improve performance under short codes [25, 28]. Recent work reduces manual design through automated or geometry-aware PQ variants [13, 39], and PQ is increasingly used as a memory reduction primitive in larger systems, including camera relocalization [26], diffusion modeling [42], and Transformer inference [53]. More broadly, vector quantization ideas influence large discrete codebooks in generative models [55].

Most PQ variants focus on minimizing Euclidean reconstruction error, without explicitly considering how fixed per-group budgets interact with downstream task sensitivity. In contrast, ORFC learns an orthogonal reparameterization specifically to balance subspace sensitivity under grouped PQ, and couples it with entropy modeling to improve low-bitrate feature coding efficiency.



(a) Task and deviation correlation **(b)** Group-wise marginal gains

Fig. 2: Diagnosing task distortion under grouped ViT feature coding. (a) Tail-output deviation $\mathcal{L}_{\text{ref}}$ tracks accuracy drop more reliably than feature MSE. (b) Group-wise marginal gains ΔAcc_g are highly imbalanced under fixed grouped coding, only $\mathcal{L}_{\text{ref}}$-driven rotation substantially balances them.

2.3 Split Inference

A broad line of work on split inference and split learning partitions networks across edge and server to balance latency, compute budgets, privacy, and communication. Early edge-cloud systems showed that inserting a bottleneck at a split point and optimizing it with the task objective can substantially reduce communication while preserving accuracy [35, 41]. For ViT models, recent studies revisit splitting under transformer architectures and deployment constraints, including adaptive split designs, collaborative inference with near-edge resources, and mobile offloading or scheduling strategies [30, 32, 49, 54].

Beyond partitioning, many approaches reduce communication by integrating structure-aware compression into the split pipeline, such as codebook-based tokenization and attention-guided token selection [1, 52]. Robustness under lossy links has also been explored, including splitting policies that handle device failures and training schemes tolerant to packet loss [20, 23].

These developments indicate that effective ViT split inference at low bitrate requires both task-aligned bottleneck objectives and subspace-aware resource allocation. ORFC follows this perspective by introducing an orthogonal reparameterization that aligns grouped quantization with task sensitivity under an entropy-constrained objective.

3 Diagnosing Task Distortion

Preliminary. Given an L-layer vision transformer $M_{1\to L}(\cdot)$, we split it at layer l into a front $M_{1\to l}(\cdot)$ and a tail $M_{l+1\to L}(\cdot)$. For an input image x, the intermediate representation is $F = M_{1\to l}(x) \in \mathbb{R}^{T\times D}$, where T is the number of tokens and D is the channel dimension. A feature codec consists of an encoder $Enc(\cdot)$ that maps F to a compact bitstream and a decoder $Dec(\cdot)$ that reconstructs $\hat{F} = Dec(Enc(F))$. The tail produces outputs $o = M_{l+1\to L}(F)$ and $\hat{o} = M_{l+1\to L}(\hat{F})$,

which are further fed into the task head for prediction. To evaluate how coding affects downstream inference, we use the frozen tail as a fixed evaluator and measure the output deviation:

$$\mathcal{L}_{\text{ref}} = \|o - \hat{o}\|_2^2. \tag{1}$$

In practical coding settings, directly quantizing the full D-dimensional token vector would require an exponentially large codebook. We therefore partition F along the channel dimension into G subspaces and quantize each group with a small codebook, forming a grouped quantization structure.

Empirical diagnosis. We conduct a diagnostic study on DINOv2 ViT-L/14 features (20^{th} block) using product quantization with G=32. Fig. 2a shows that feature-space MSE does not reliably track task degradation: configurations with similar MSE can yield very different accuracy drops, while $\mathcal{L}_{\text{ref}}$ varies more consistently with the drop. This indicates that minimizing feature reconstruction error may preserve Euclidean fidelity of F without maintaining downstream task performance.

We further analyze how task sensitivity is distributed across groups. For each group g, we restore only this group to its unquantized value while keeping all other groups quantized, and record the resulting accuracy improvement as ΔAcc_g. Fig. 2b reveals *strong non-uniformity* of ΔAcc_g under the Identity representation, indicating that task sensitivity is *highly anisotropic* across subspaces: some groups dominate task performance while others contribute only marginally. An MSE-driven orthogonal transform, as used in OPQ [18], does not significantly reduce this imbalance. It primarily reshapes the representation to minimize feature distortion, without explicitly considering how grouped budgets interact with task sensitivity. In contrast, a rotation learned under $\mathcal{L}_{\text{ref}}$ produces a *much flatter* ΔAcc_g profile, suggesting that task-aligned learning can reorganize the representation so that grouped quantization becomes *better aligned with downstream objectives*.

RD intuition and implication. From classical bit allocation in transform coding [19], when distortion decomposes into per-component functions $D = \sum_g D_g(R_g)$ under a total rate constraint $\sum_g R_g \leq R$, a necessary optimality condition is the equal-slope criterion: the marginal distortion reduction per bit should be balanced across components, $-\partial D_g/\partial R_g = \lambda$ for all group g.

Under grouped PQ with a fixed codebook size per group, the effective rate per group is nearly uniform, which limits the ability to directly adjust R_g according to task sensitivity. Introducing heterogeneous codebooks or explicit rate control increases system complexity. Instead, we pursue a representation-side solution. *By learning an orthogonal reparameterization, we redistribute sensitive directions across groups so that different groups have more comparable task influence.* This reshaping makes the implicit per-group budgets better aligned with downstream sensitivity, reducing inefficient bit usage and mitigating low-bitrate collapse. This analysis motivates ORFC, which integrates orthogonal reparameterization with a task-aligned output deviation distortion in an entropy-constrained objective.

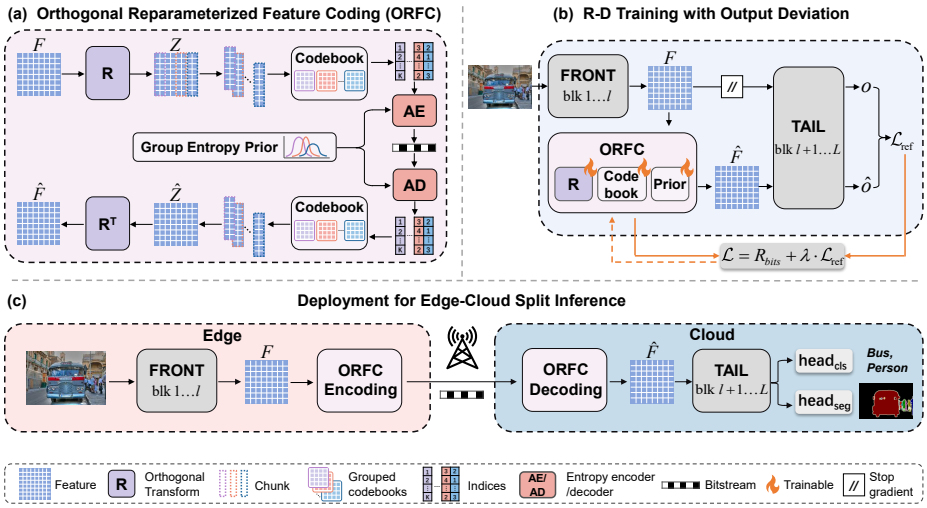


Fig. 3: Overview of ORFC. (a) The codec applies an orthogonal transform, grouped product quantization, and entropy coding with group-wise learned priors. (b) With the tail frozen, we train the transform, codebooks, and priors end-to-end using an entropy-constrained objective with rate term R_{bits} and output deviation distortion $\mathcal{L}_{\text{ref}}$. (c) In deployment for edge-cloud split inference, the edge sends entropy-coded indices, and the cloud reconstructs features for the tail and task heads.

4 Orthogonal Reparameterized Feature Coding

We propose **ORFC**, an orthogonal reparameterized feature coding framework for ViT intermediate representations in split inference (Fig. 3). The codec applies a learnable orthogonal transform along the channel dimension, quantizes the rotated features using grouped product quantization, and entropy-codes the resulting indices with group-wise learned priors. As discussed in Sec. 3, fixed-size grouped PQ induces near-uniform rate allocation across groups, which may be misaligned with the task objective. ORFC alleviates this mismatch at the representation level: the learned orthogonal transform reorganizes feature directions so that task influence is more comparable across groups, improving coding efficiency in the low-bitrate regime.

4.1 Orthogonal Reparameterization

Given the intermediate feature $F \in \mathbb{R}^{T\times D}$, we apply a learnable orthogonal transform $R \in \mathbb{R}^{D\times D}$ and perform quantization in the rotated space. We normalize the feature magnitude before rotation and reapply it after decoding, so the orthogonal transform is learned on a normalized magnitude without entangling the global activation scale. The forward and inverse transforms are:

$$Z = FR, \tag{2}$$

$$\hat{F} = \hat{Z}R^{\top}. \tag{3}$$

Orthogonality guarantees an exact inverse and preserves the global energy of the representation. Under grouped quantization, this allows the transform to redistribute channel directions across groups without introducing additional scaling effects. After rotation, the grouped partition aligns more closely with task-sensitive components.

To enforce orthogonality during training, we parameterize R using the Cayley transform. We optimize a skew-symmetric matrix A and construct:

$$R(A) = (I - A)^{-1}(I + A), \tag{4}$$

where $A^{\top} = -A$. This parameterization ensures $R^{\top}R = I$ by construction and avoids auxiliary regularization or projection steps. Additional implementation details are provided in the supplementary material.

4.2 Grouped Product Quantization with Entropy Prior

We partition Z into G groups along the channel dimension. Each group has dimension $d = D/G$ and is quantized with an independent codebook. For group g, the codebook is $C_g = \{c_{g,1}, \ldots, c_{g,N}\}$ with $c_{g,j} \in \mathbb{R}^d$. For token t and group g, let $z_{t,g} \in \mathbb{R}^d$ denote the corresponding subvector. The distortion of assigning codeword j is:

$$\mathrm{dist}_{t,g}(j) = \|z_{t,g} - c_{g,j}\|_2^2. \tag{5}$$

Grouped codebooks provide a scalable and parallelizable structure with predictable memory cost. In ORFC, the preceding orthogonal transform reorganizes channel directions so that this fixed grouped structure operates on subspaces with more balanced task influence.

To estimate bitrate and enable rate control, we introduce a learnable categorical prior for each group. For group g, we optimize logits $\mathbf{p}_g \in \mathbb{R}^N$ and define the prior distribution as:

$$\boldsymbol{\pi}_g = \mathrm{softmax}(\mathbf{p}_g). \tag{6}$$

Under entropy-constrained quantization, index selection follows an RD trade-off with a shared Lagrange multiplier. The selection cost is:

$$\mathrm{cost}_{t,g}(j) = \mathrm{dist}_{t,g}(j) + \frac{1}{\lambda}\big(-\log_2 \pi_{g,j}\big), \tag{7}$$

and the hard assignment is:

$$k_{t,g} = \arg\min_{j} \ \mathrm{cost}_{t,g}(j). \tag{8}$$

The entropy term biases assignments toward higher-probability codewords and provides a differentiable estimate of bitrate through $-\log_2 \pi_{g,j}$. The scalar λ controls the global rate-distortion trade-off and enables continuous bitrate adjustment without modifying codebook sizes or the grouped structure.

4.3 Optimization and Deployment

We train ORFC using an entropy-constrained objective composed of a rate term and the output deviation $\mathcal{L}_{\text{ref}}$. The rate is computed from hard indices to ensure consistency with inference-time entropy coding. The total bitrate is:

$$R_{\text{bits}} = \sum_{t=1}^{T} \sum_{g=1}^{G} \big(-\log_2 \pi_{g,\, k_{t,g}} \big), \tag{9}$$

and the objective is:

$$\mathcal{L} = R_{\text{bits}} + \lambda \, \mathcal{L}_{\text{ref}}. \tag{10}$$

Although the indices are discrete, R_{bits} is differentiable with respect to the prior parameters through $\boldsymbol{\pi}_g$. The same scalar λ governs both the global objective in Eq. (10) and the local assignment rule in Eq. (7), aligning index decisions with the overall operating point.

To propagate gradients to the rotation R and the codebooks $\{C_g\}$, we adopt a straight-through estimator. In the backward pass, we compute a temperature-controlled soft distribution over codewords:

$$\tilde{\mathbf{w}}_{t,g} = \text{softmax}\left(-\frac{\text{cost}_{t,g}}{\tau} \right), \tag{11}$$

and combine it with the hard one-hot vector $\mathbf{h}_{t,g} = \mathbf{e}_{k_{t,g}}$ as:

$$\mathbf{w}_{t,g} = \mathbf{h}_{t,g} - \text{sg}[\tilde{\mathbf{w}}_{t,g}] + \tilde{\mathbf{w}}_{t,g}. \tag{12}$$

Each group is reconstructed as the weighted sum of its codewords using $\mathbf{w}_{t,g}$. The reconstructed groups are concatenated to form $\hat{Z}$, and $\hat{F}$ is recovered via Eq. (3). This allows gradients to update both the codebooks and the orthogonal transform so that the rotation reorganizes channel directions under the fixed grouped structure.

At deployment, the edge device runs the front network to produce F, applies the learned rotation, performs grouped quantization to obtain hard indices $\{k_{t,g}\}$, and entropy-encodes them using $\{\boldsymbol{\pi}_g\}$ to generate the bitstream. The cloud decodes the indices, reconstructs $\hat{Z}$ by table lookup, applies $R^\top$ to recover $\hat{F}$, and executes the tail network and task heads for prediction.

Table 1: Tasks, datasets, models, split points, and feature shapes used for evaluation.

Task	Datasets	Model	Split Point	Feature Shape
Classification	ImageNet, 500 samples	DINOv2 ViT-G/14	$10^{\text{th}}, 20^{\text{th}}, 30^{\text{th}}$	257×1536
		DINOv2 ViT-L/14	$10^{\text{th}}, 15^{\text{th}}, 20^{\text{th}}$	257×1024
		CLIP ViT-L/14	$10^{\text{th}}, 15^{\text{th}}, 20^{\text{th}}$	257×1024
Segmentation	VOC2012, 100 samples	DINOv2 ViT-G/14	$10^{\text{th}}, 20^{\text{th}}, 30^{\text{th}}$	$2 \times 1370 \times 1536$
		DINOv2 ViT-L/14	$10^{\text{th}}, 15^{\text{th}}, 20^{\text{th}}$	$2 \times 1370 \times 1024$
Detection	COCO2017, 100 samples	VitDet ViT-B/16	$3^{\text{rd}}, 6^{\text{th}}, 9^{\text{th}}$	$64 \times 64 \times 768$
Retrieval	COCO2017, 500 samples	SigLIP2 So400m/14	$8^{\text{th}}, 16^{\text{th}}, 24^{\text{th}}$	256×1152

5 Experiments

We evaluate intermediate feature coding across five ViT backbones, four downstream tasks, and multiple split points under a consistent split-inference protocol. We compare ORFC with representative baselines, then quantify its runtime and memory overhead, followed by ablation and generalization analyses.

5.1 Experimental Setup

Models and split points. We evaluate five representative ViT backbones: DINOv2 ViT-G/14 and ViT-L/14 [38] as self-supervised visual foundation models, VitDet ViT-B/16 [27] as a MAE-pretrained plain ViT backbone, and CLIP ViT-L/14 [40] and SigLIP2 So400m/14 [48] as contrastive vision-language models. We use multiple split points to evaluate feature coding under different representation regimes: earlier features retain more local detail and redundancy, while deeper features are more semantic and more coupled to the tail computation. Unless otherwise specified, we report main results on the split points in Tab. 1, with additional layers in the supplementary.

Baselines. We compare ORFC with three representative low-bandwidth codecs. (i) VTM [4] is a strong generic compressor that reshapes intermediate tokens into pseudo frames and applies a video codec pipeline, but incurs substantial compute overhead. (ii) LaMoFC [17] represents learning-based entropy models by truncating features in the numeric domain and training a hyper-prior with feature-space reconstruction distortion. (iii) OPQ [18] learns a rotation and PQ codebooks to reduce MSE-based quantization distortion across subspaces, but does not explicitly address the mismatch between feature distortion and tail output deviation.

Datasets and metrics. For training, we sample 5,000 images from ImageNet [8] and extract intermediate features at the selected split points. For evaluation, we use the test subsets listed in Tab. 1 and report task-specific metrics: Acc@1 for ImageNet classification, mIoU for VOC2012 semantic segmentation [12], mAP@0.5:0.95 for COCO2017 object detection [29], and the average Recall@1 of image-to-text and text-to-image retrieval on COCO2017. Following the feature coding benchmark protocol [17], bitrate is measured as bits per feature point

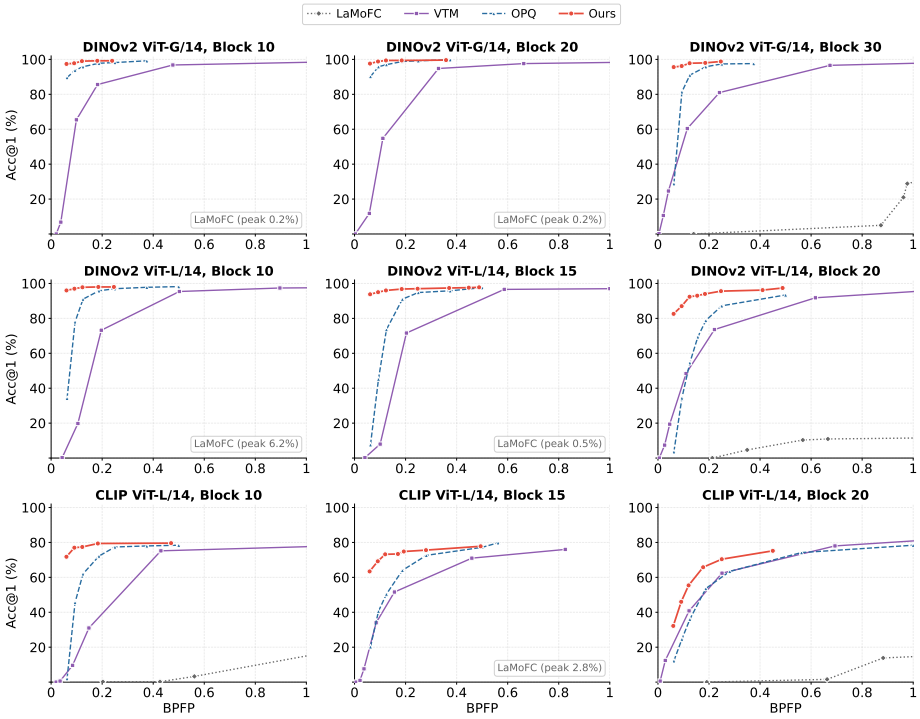


Fig. 4: Classification RD curves of DINOv2 ViT-G/14, DINOv2 ViT-L/14, and CLIP ViT-L/14. The original Acc@1 is 99.8%, 98.4%, and 81.0%, respectively.

(BPFP). To summarize RD efficiency over a curve, we report BD-rate relative to VTM computed from RD curves over the operating range.

Training details. We vary the codebook size $N \in \{4, 8, 16, 64, 256\}$ and the embedding dimension $d \in \{16, 32\}$ to cover a wide bitrate range. We use soft assignment with a temperature annealing schedule, where τ decays exponentially from 0.5 to 0.005. All learnable parameters are optimized with Adam at a learning rate of 3×10^{-4}. We set $\lambda = 0.5$ to balance entropy rate and the output deviation distortion $\mathcal{L}_{\text{ref}}$. Unless stated otherwise, we use $G \in \{32, 64\}$ for grouped PQ and the same grouping across methods for fair comparison.

5.2 Main Results

Classification RD curves across backbones and layers. Fig. 4 shows BPFP versus Acc@1 at different split points for DINOv2 and CLIP features. ORFC consistently achieves higher accuracy at matched bitrate, with the largest gains in the low-bitrate regime. For instance, on DINOv2 ViT-L/14 at the 10^{th} block, ORFC reaches 96.0% Acc@1 at 0.06 BPFP, whereas OPQ obtains 34.2% and VTM remains at 19.8% even at 0.11 BPFP. VTM often exhibits an abrupt

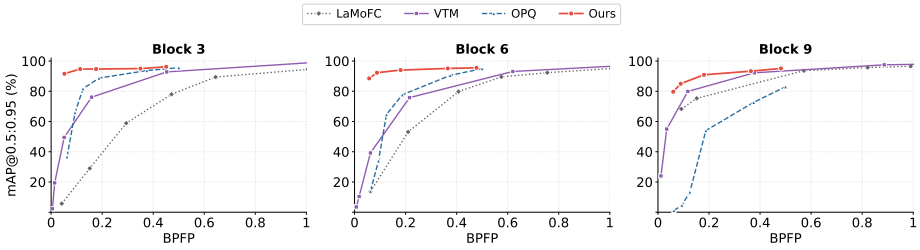


Fig. 5: Detection RD curves of VitDet ViT-B/16. The original mAP is 98.0%.

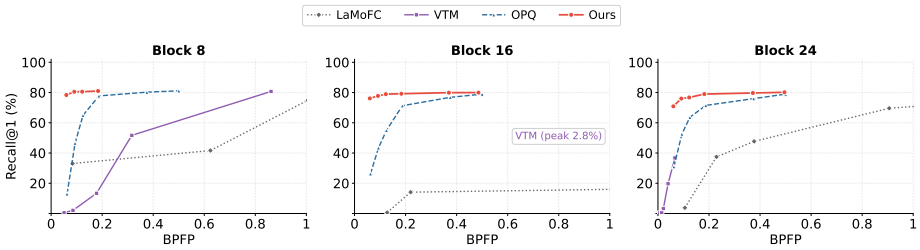


Fig. 6: Retrieval RD curves of SigLIP2 So400m/14. The original Recall@1 is 81.0%.

performance drop when BPFP falls below a threshold, suggesting that generic signal compression does not reliably preserve task-relevant directions. LaMoFC yields a weak RD trade-off and remains below other methods at most operating points. OPQ improves over identity grouping by reducing MSE-based feature distortion, but its gains diminish in the low-bitrate regime. In contrast, ORFC maintains higher accuracy as bitrate decreases. By optimizing the rotation and codebooks under output-deviation distortion, it makes the representation more compatible with the fixed grouped structure and delays accuracy collapse.

Detection and retrieval RD curves. Figs. 5 and 6 further evaluates ORFC on object detection with VitDet and image-text retrieval with SigLIP2. These tasks impose different requirements on the reconstructed features: detection relies on spatial localization and region-level evidence, while retrieval depends on preserving global cross-modal alignment. ORFC consistently achieves better rate-task trade-offs in both settings, especially at low bitrates. For example, on VitDet ViT-B/16 at the 3^{rd} block, ORFC reaches 91.6% mAP at 0.05 BPFP, while VTM and OPQ obtain 49.5% and 36.7% at comparable bitrates. On SigLIP2 So400m/14, ORFC maintains high Recall@1 with substantially fewer bits than VTM and LaMoFC, and also improves over OPQ in the low-bitrate regime.

Segmentation RD curves and qualitative results. Fig. 7 reports the segmentation results on DINOv2 features, where preserving spatial coherence after reconstruction is important. ORFC shows the largest gains at shallow and intermediate layers, whose representations retain richer spatial structure and are more vulnerable to reconstruction errors. On DINOv2 ViT-G/14 at the 20^{th}

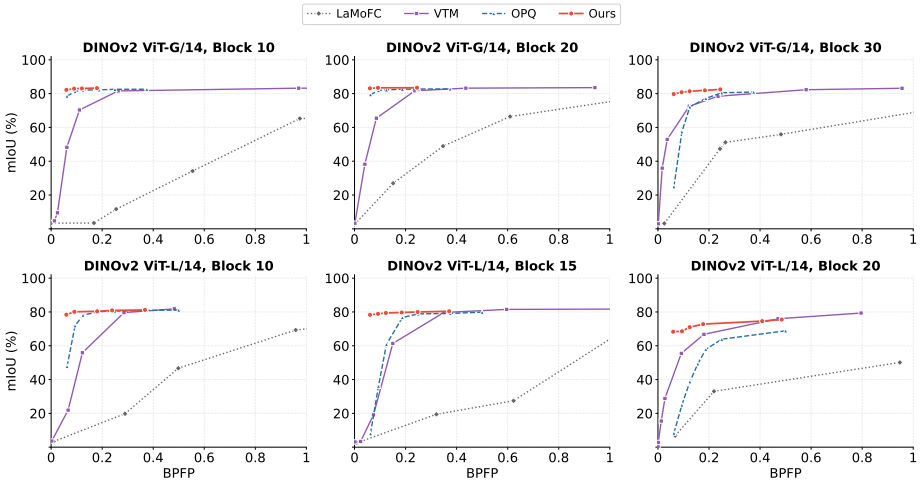


Fig. 7: Segmentation RD curves of DINOv2 ViT-G/14 and ViT-L/14. The original mIoU is 83.4% and 81.0%, respectively.

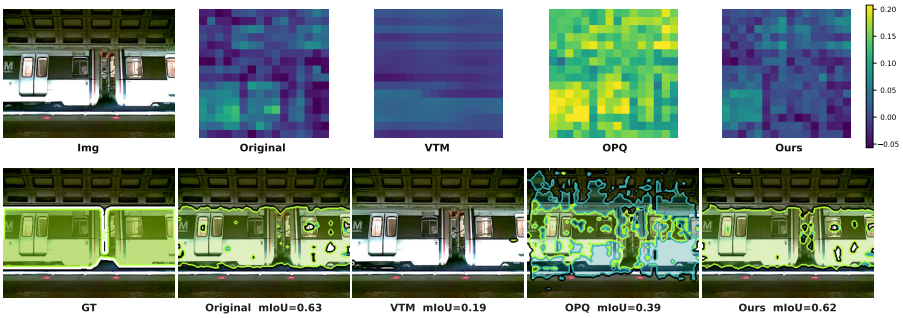


Fig. 8: Qualitative comparison of reconstructed features and segmentation results.

block, ORFC reaches 83.1% mIoU at about 0.06 BPFP, close to the original 83.3%, while VTM drops to 65.5% at 0.09 BPFP. Although the gaps narrow at deeper layers, ORFC remains consistently better at matched bitrates. The qualitative examples in Fig. 8 further support the RD trends. We visualize reconstructed features from the 10^{th} block of DINOv2 ViT-L/14 and report the corresponding segmentation results at a matched bitrate of 0.09 BPFP. VTM and OPQ introduce noticeable structural artifacts, leading to spurious regions and fragmented mask boundaries. In contrast, ORFC better preserves the original feature patterns and produces cleaner segmentation results with fewer isolated false positives.

Runtime and memory overhead. We quantify the overhead of the orthogonal transform by reporting the latency (ms) and peak memory (MB) of the transform

Table 2: Runtime and memory overhead of ORFC.

Model	D	Stage	Latency (ms)		Memory (MB)	
			R	All	R	All
ViT-L/14	1024	Enc	0.36	1.23	25.5	39.8
		Dec	0.08	0.87	28.5	36.6
ViT-G/14	1536	Enc	0.42	1.53	45.1	62.8
		Dec	0.11	1.25	49.6	59.4

Table 3: Ablation on DINOv2 ViT-L/14 features at the 20^{th} block. We report the BD-rate relative to VTM for classification and segmentation.

	$\mathcal{L}_{\text{ref}}$	R	Soft assign	BD-rate$_{\text{Cls}}$	BD-rate$_{\text{Seg}}$
(a)				-6.7%	190.7%
(b)		✓	✓	-5.2%	153.2%
(c)	✓		✓	-57.0%	14.4%
(d)	✓	✓		-24.8%	181.2%
(e)	✓	✓	✓	**-77.5%**	**-41.4%**

alone (R) and the full ORFC codec pipeline (All) on DINOv2 ImageNet features with $N = 64$ and $G = 32$. All measurements are conducted on a workstation with an NVIDIA RTX 4090 GPU and an Intel Xeon Silver 4310 CPU. As shown in Tab. 2, the transform adds only modest cost within the complete pipeline, and the total latency and memory footprint remain lightweight for both ViT-L/14 and ViT-G/14, making ORFC practical for split inference.

5.3 Ablation Study and Analysis

Effect of output deviation distortion $\mathcal{L}_{\text{ref}}$. Tab. 3 (b) evaluates the role of the distortion definition. Replacing $\mathcal{L}_{\text{ref}}$ with feature-space MSE changes the optimization target to pure reconstruction error. While this reduces feature distortion, it does not consistently preserve tail behavior, leading to inferior BD-rate, especially for segmentation. In contrast, $\mathcal{L}_{\text{ref}}$ provides a distortion signal aligned with downstream inference and yields stable gains across tasks.

Effect of the orthogonal transform. Removing the rotation R forces grouped PQ to operate in the original basis. Under sensitivity imbalance, this causes heterogeneous task-relevant directions to be mixed within groups. With near-uniform per-group rates, such mixing limits coding efficiency. Learning R under $\mathcal{L}_{\text{ref}}$ reorganizes channel directions so that grouped quantization better matches task sensitivity, significantly improving BD-rate. The improvement is more pronounced for segmentation, which is more sensitive to structured distortions.

Soft assignment and codebook stability. Soft assignment is critical for stable end-to-end optimization with a rate term. With only hard assignments, gradients primarily pass through selected codewords while the entropy term encourages sparse usage, which can reduce effective codebook diversity. This results in inferior BD-rate, as shown in Tab. 3 (d). Soft assignment with temperature an-

Table 4: Generalization analysis on DINOv2 ViT-L/14. We report BD-rate (%) relative to VTM across downstream tasks.

Task	blk05	blk10	blk15	blk20
Classification	-84.2	-88.6	-78.5	-77.5
Segmentation	-59.5	-65.8	-59.9	-41.4
Depth estimation	-73.3	-90.0	-64.1	-50.6

nealing distributes gradients across candidate codewords during early training, enabling stable joint optimization of codebooks and priors.

Generalization analysis. We further evaluate the generalizability of the learned reparameterization across tasks and datasets. We train the codec with DINOv2 ViT-L/14 features extracted from ImageNet and evaluate it on ImageNet classification, Pascal VOC2012 segmentation, and NYU Depth v2 [44] depth estimation. As shown in Tab. 4, ORFC consistently reduces BD-rate relative to VTM across all evaluated split points. The gains on semantic and geometric dense prediction indicate that the learned reparameterization better organizes task-relevant directions in the ViT representation, yielding coding benefits across different downstream objectives.

6 Conclusion

We study intermediate feature coding for vision transformers and present ORFC, an orthogonal reparameterized product quantization framework tailored for split inference. Our analysis shows that under fixed grouped product quantization, task sensitivity across channel subspaces is highly uneven, and naive distortion minimization in feature space does not reliably preserve downstream performance at low bitrate. To address this structural mismatch, we introduce a learnable orthogonal reparameterization that transforms the representation before grouped quantization, making task influence more balanced across groups under fixed group budgets. Combined with entropy-constrained training and an output-deviation distortion defined at the frozen tail, the proposed framework aligns representation geometry with the grouped coding structure without modifying codebook size or rate allocation mechanisms. Extensive experiments across multiple ViT backbones, split points, and downstream tasks demonstrate that ORFC consistently improves RD performance, with particularly strong gains in the low-bitrate regime.

Acknowledgements

This work was supported by AI Joint Lab of Future Urban Infrastructure sponsored by Fuzhou Chengtou New Infrastructure Group and Boyun Vision Co. Ltd, and in part by the PKU-NTU Joint Research Institute (JRI) sponsored by a donation from the Ng Teng Fong Charitable Foundation.

References

1. Alvetreti, F., Pomponi, J., Di Lorenzo, P., Scardapane, S.: Communication efficient split learning of vits with attention-based double compression. arXiv preprint arXiv:2509.15058 (2025)
2. Arnab, A., Dehghani, M., Heigold, G., Sun, C., Lučić, M., Schmid, C.: Vivit: A video vision transformer. In: IEEE/CVF International Conference on Computer Vision. pp. 6836–6846 (2021)
3. Ballé, J., Minnen, D., Singh, S., Hwang, S.J., Johnston, N.: Variational image compression with a scale hyperprior. In: International Conference on Learning Representations (2018)
4. Bross, B., Wang, Y.K., Ye, Y., Liu, S., Chen, J., Sullivan, G.J., Ohm, J.R.: Overview of the versatile video coding (VVC) standard and its applications. IEEE Transactions on Circuits and Systems for Video Technology **31**(10), 3736–3764 (2021)
5. Chen, Z., Fan, K., Wang, S., Duan, L.Y., Lin, W., Kot, A.: Lossy intermediate deep learning feature compression and evaluation. In: ACM International Conference on Multimedia. pp. 2414–2422 (2019)
6. Choi, H., Bajić, I.V.: Deep feature compression for collaborative object detection. In: IEEE International Conference on Image Processing. pp. 3743–3747 (2018)
7. Davisson, L.: Rate distortion theory: A mathematical basis for data compression. IEEE Transactions on Communications **20**(6), 1202–1202 (1972)
8. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255 (2009)
9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021)
10. Duan, L.Y., Lou, Y., Bai, Y., Huang, T., Gao, W., Chandrasekhar, V., Lin, J., Wang, S., Kot, A.C.: Compact descriptors for video analysis: The emerging mpeg standard. IEEE Multimedia **26**(2), 44–54 (2018)
11. El-Nouby, A., Muckley, M.J., Ullrich, K., Laptev, I., Verbeek, J., Jegou, H.: Image compression with product quantized masked image modeling. Transactions on Machine Learning Research (2023)
12. Everingham, M., Eslami, S.A., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. International Journal of Computer Vision **111**(1), 98–136 (2015)
13. Gan, X., Wang, Y., Zhao, X., Wang, W., Wang, Y., Liu, Z.: Autodpq: automated differentiable product quantization for embedding compression. In: International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 1833–1837 (2023)
14. Gao, C., Jiang, Y., Wu, S., Ma, Y., Li, L., Liu, D.: Imofc: Identity-level metric optimized feature compression for identification tasks. IEEE Transactions on Circuits and Systems for Video Technology **35**(2), 1855–1869 (2024)
15. Gao, C., Liu, S., Wu, F., Lin, W.: Cross-architecture universal feature coding via distribution alignment. In: IEEE International Conference on Image Processing Workshops. pp. 428–433 (2025)
16. Gao, C., Liu, Z., Li, L., Liu, D., Sun, X., Lin, W.: Dt-ufc: Universal large model feature coding via peaky-to-balanced distribution transformation. In: ACM International Conference on Multimedia. pp. 5198–5207 (2025)

17. Gao, C., Ma, Y., Chen, Q., Xu, Y., Liu, D., Lin, W.: Feature coding in the era of large models: Dataset, test conditions, and benchmark. In: IEEE/CVF International Conference on Computer Vision. pp. 1068–1077 (2025)
18. Ge, T., He, K., Ke, Q., Sun, J.: Optimized product quantization for approximate nearest neighbor search. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 2946–2953 (2013)
19. Goyal, V.K.: Theoretical foundations of transform coding. IEEE Signal Processing Magazine **18**(5), 9–21 (2002)
20. Guo, X., Jiang, Q., Shen, Y., Pimentel, A.D., Stefanov, T.: Easter: Learning to split transformers at the edge robustly. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **43**(11), 3626–3637 (2024)
21. Harell, A., Foroutan, Y., Ahuja, N., Datta, P., Kanzariya, B., Somayazulu, V.S., Tickoo, O., de Andrade, A., Bajić, I.V.: Rate-distortion theory in coding for machines and its applications. IEEE Transactions on Pattern Analysis and Machine Intelligence (2025)
22. He, D., Yang, Z., Peng, W., Ma, R., Qin, H., Wang, Y.: Elic: Efficient learned image compression with unevenly grouped space-channel contextual adaptive coding. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5718–5727 (2022)
23. Itahara, S., Nishio, T., Yamamoto, K.: Packet-loss-tolerant split inference for delay-sensitive deep learning in lossy wireless networks. In: IEEE Global Communications Conference. pp. 1–6 (2021)
24. Jegou, H., Douze, M., Schmid, C.: Product quantization for nearest neighbor search. IEEE Transactions on Pattern Analysis and Machine Intelligence **33**(1), 117–128 (2010)
25. Klein, B., Wolf, L.: End-to-end supervised product quantization for image search and retrieval. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5041–5050 (2019)
26. Laskar, Z., Melekhov, I., Benbihi, A., Wang, S., Kannala, J.: Differentiable product quantization for memory efficient camera relocalization. In: European Conference on Computer Vision. pp. 470–489 (2024)
27. Li, Y., Mao, H., Girshick, R., He, K.: Exploring plain vision transformer backbones for object detection. In: European conference on computer vision. pp. 280–296. Springer (2022)
28. Liang, Y., Zhang, S., Li, L.K., Wang, X.: Unleashing the full potential of product quantization for large-scale image retrieval. Advances in Neural Information Processing Systems **36**, 61712–61724 (2023)
29. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)
30. Liu, H., Fahmy, S.A.: Ask the expert: Collaborative inference for vision transformers with near-edge accelerators. arXiv preprint arXiv:2602.13334 (2026)
31. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. Advances in Neural Information Processing Systems **36**, 34892–34916 (2023)
32. Liu, Y., Lu, R., Wang, D.: Lvmscissor: Split and schedule large vision model inference on mobile edges via salp swarm algorithm. IEEE Transactions on Mobile Computing (2025)
33. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: IEEE/CVF International Conference on Computer Vision. pp. 10012–10022 (2021)

34. Matsubara, Y., Levorato, M., Restuccia, F.: Split computing and early exiting for deep learning applications: Survey and research challenges. ACM Computing Surveys **55**(5), 1–30 (2022)
35. Matsubara, Y., Yang, R., Levorato, M., Mandt, S.: Supervised compression for resource-constrained edge computing systems. In: IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 2685–2695 (2022)
36. Niu, B., Wei, Z., He, Y.: Entropy-based deep product quantization for visual search and deep feature compression. In: International Conference on Visual Communications and Image Processing (VCIP). pp. 1–5 (2021)
37. Norouzi, M., Fleet, D.J.: Cartesian k-means. In: IEEE Conference on computer Vision and Pattern Recognition. pp. 3017–3024 (2013)
38. Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., HAZIZA, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. Transactions on Machine Learning Research (2024)
39. Qiu, Z., Liu, J., Chen, Y., King, I.: Hihpq: Hierarchical hyperbolic product quantization for unsupervised image retrieval. In: AAAI Conference on Artificial Intelligence. vol. 38, pp. 4614–4622 (2024)
40. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning. pp. 8748–8763 (2021)
41. Shao, J., Zhang, J.: Bottlenet++: An end-to-end approach for feature compression in device-edge co-inference systems. In: IEEE International Conference on Communications Workshops. pp. 1–6 (2020)
42. Shao, J., Zhang, H., Yu, H., Wu, J.: Memory-efficient generative models via product quantization. In: IEEE/CVF International Conference on Computer Vision. pp. 16871–16881 (2025)
43. Shuvo, M.M.H., Islam, S.K., Cheng, J., Morshed, B.I.: Efficient acceleration of deep learning inference on resource-constrained edge devices: A review. Proceedings of the IEEE **111**(1), 42–91 (2022)
44. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from rgbd images. In: European conference on computer vision. pp. 746–760. Springer (2012)
45. Singh, A., Sharma, V., Sukumaran, R., Mose, J., Chiu, J., Yu, J., Raskar, R.: Simba: Split inference—mechanisms, benchmarks and attacks. In: European Conference on Computer Vision. pp. 214–232 (2024)
46. Staniszewski, K., La'ncucki, A.: Kv cache transform coding for compact storage in llm inference. arXiv preprint arXiv:2511.01815 (2025)
47. Thapa, C., Arachchige, P.C.M., Camtepe, S., Sun, L.: Splitfed: When federated learning meets split learning. In: AAAI Conference on Artificial Intelligence. vol. 36, pp. 8485–8493 (2022)
48. Tschannen, M., Gritsenko, A., Wang, X., Naeem, M.F., Alabdulmohsin, I., Parthasarathy, N., Evans, T., Beyer, L., Xia, Y., Mustafa, B., et al.: Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. arXiv preprint arXiv:2502.14786 (2025)
49. Wang, L., Shang, B., Li, Y., Mohapatra, P., Dong, W., Wang, X., Zhu, Q.: Split adaptation for pre-trained vision transformers. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20092–20102 (2025)
50. Wang, S., Yang, W., Wang, S.: End-to-end facial deep learning feature compression with teacher-student enhancement. arXiv preprint arXiv:2002.03627 (2020)

51. Wang, W., An, P., Huang, X., Huang, K., Yang, C.: Intermediate deep feature coding for human–machine vision collaboration. Journal of Visual Communication and Image Representation **95**, 103859 (2023)
52. Wang, X., Zhou, Z., Li, Y., An, X., Wang, H.: Codebook-based adaptive feature compression with semantic enhancement for edge-cloud systems. arXiv preprint arXiv:2509.18481 (2025)
53. Zhang, H., Ji, X., Chen, Y., Fu, F., Miao, X., Nie, X., Chen, W., Cui, B.: Pq-cache: Product quantization-based kvcache for long context llm inference. ACM on Management of Data **3**(3), 1–30 (2025)
54. Zhao, S., Liu, T., Jin, H., Yao, D.: Spvit: Accelerate vision transformer inference on mobile devices via adaptive splitting and offloading. IEEE Transactions on Mobile Computing (2025)
55. Zhu, L., Wei, F., Lu, Y., Chen, D.: Scaling the codebook size of vq-gan to 100,000 with a utilization rate of 99%. Advances in Neural Information Processing Systems **37**, 12612–12635 (2024)