AXISQ: DEPTH-ADAPTIVE CHANNEL–FREQUENCY QUANTIZATION FOR CAMOUFLAGED OBJECT DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Camouflaged object detection (COD) depends on weak texture and boundary cues that can be destroyed by aggressive post-training quantization (PTQ). Existing PTQ methods mainly optimize quantizers or reconstruct network blocks, while using a single precision-allocation domain throughout hierarchical backbones. Instead, we introduce AxisQ, a depth-adaptive bit-allocation framework motivated by the observation that the useful allocation axis changes with feature depth. Under a fixed per-layer average activation-bit budget, AxisQ allocates precision in the channel domain for shallow features and in the local-frequency domain for deep features, using COD-aware sensitivity to better preserve subtle foreground regions and ambiguous boundaries. We also formulate a boundary-aware Hessian approximation for reconstruction by incorporating boundary-weighted gradient and error statistics. Experiments across multiple COD models and diverse benchmarks demonstrate improved accuracy over the strongest baselines at low bit widths, with relative weighted F-measure gains of 12.7% for CFRN and 33.3% for CamoFormer-P on CAMO under W2A2 quantization. These results support adapting the allocation domain to feature depth as a transferable quantization strategy for hierarchical COD models.

1 INTRODUCTION

Camouflaged object detection (COD) segments objects whose color, texture, and boundaries resemble their surroundings (Fan et al., 2020). This makes accurate prediction unusually dependent on low-amplitude and spatially structured evidence. Modern state-of-the-art COD models (Hu et al., 2023; Yin et al., 2024; Song et al., 2025) usually use hierarchical Transformers to combine fine local details with semantic context, but their memory and arithmetic cost complicate deployment. Post-training quantization (PTQ) is attractive because it compresses a trained model without end-to-end retraining (Nagel et al., 2020; Li et al., 2021). At low bit widths, however, a small calibration error can erase precisely the cues that distinguish a camouflaged foreground from its background. Figure 1 illustrates foreground omissions and inaccurate boundaries under low-bit quantization.

Sensitivity-aware mixed precision and curvature-based reconstruction already account for the unequal effects of quantization perturbations (Li et al., 2021; Wu et al., 2025a). In the context of COD, recent calibration work addresses the loss of weak target cues caused by activation-range imbalance (Li et al., 2026). However, these methods typically retain the same precision-allocation domain throughout the model. Whether this domain remains suitable as features change from shallow to deep stages is less explored. This motivates exploring whether allocating precision in different domains across stages can improve quantization performance.

To investigate this possibility, we examine how feature characteristics at different depths affect the choice of precision-allocation domain. Shallow features capture local patterns, and their channels differ in sensitivity to quantization. Allocating bits by channel allows more precision to be assigned to those carrying weak target cues. Deeper features combine information over larger spatial regions. In these features, local frequency bands separate smooth spatial content from variations associated with boundaries and textures, allowing these components to receive different precision. This difference suggests that shallow and deep features may benefit from different allocation domains. The

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.

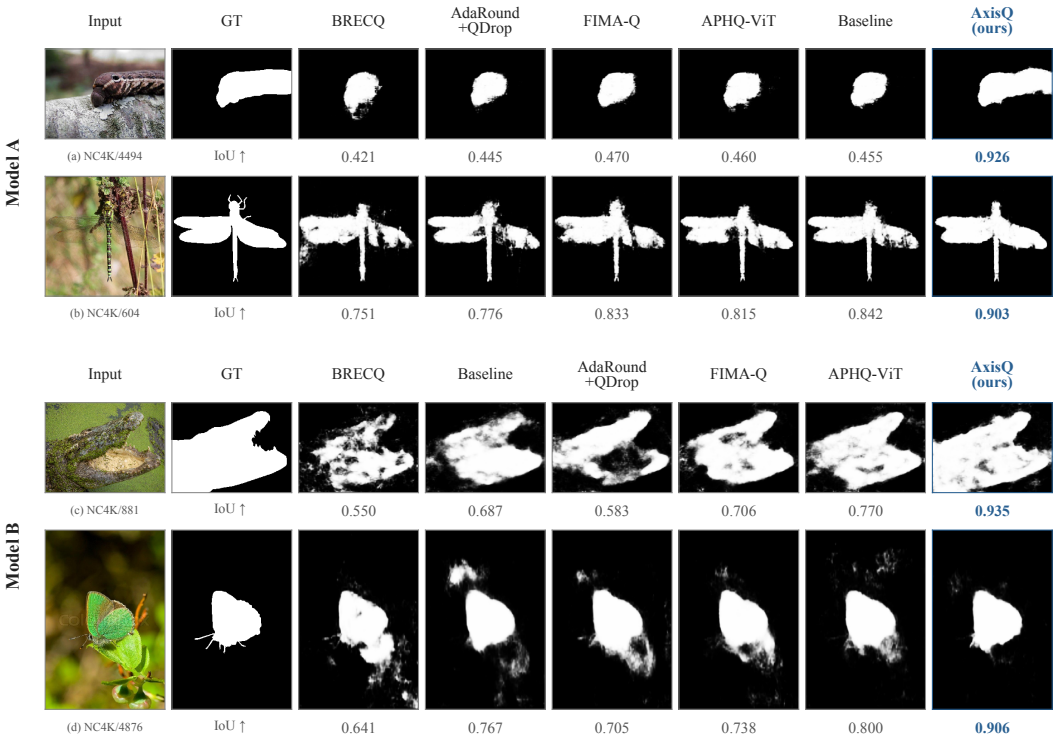


Figure 1: Qualitative comparison of quantization methods on two COD models at W2A2. Model A is CFRN, and Model B is CamoFormer-P. Each pair of rows corresponds to one model, with IoU reported for each prediction against the ground-truth mask. The compared methods show incomplete foreground regions and false detections in the background. In these examples, AxisQ recovers more complete object shapes and clearer boundaries while reducing background false positives.

analysis in Figure 2 supports channel-wise allocation in shallow stages and local-frequency allocation in deep stages.

Accordingly, we propose AxisQ, which allocates precision across channels in shallow stages and across local-frequency bands in deep stages. Figure 3 shows the overall framework. Across both model architectures, the qualitative examples in Figure 1 show that this allocation preserves complete foreground regions and boundaries at W2A2. Under a fixed per-layer average activation-bit budget, AxisQ uses COD-aware sensitivity to assign higher precision to components important for foreground and boundary prediction. We also formulate a boundary-aware Hessian approximation for reconstruction using boundary-weighted gradient and error statistics. Experiments demonstrate AxisQ's gains across COD models and precision settings. At W2A2, AxisQ improves mean weighted F-measure over the best competing results by 6.6% on CFRN and 21.5% on CamoFormer-P across four benchmarks.

In summary, our main contributions include:

- We propose AxisQ, a depth-adaptive mixed-domain PTQ framework allocating activation precision across shallow-stage channels and deep-stage local-frequency bands under a fixed per-layer average-bit budget, with an auxiliary boundary-aware Hessian approximation.
- We develop task-sensitive channel-bit allocation for COD. Label-assisted empirical-Fisher costs measure channel sensitivity, while reverse water filling followed by exact budget correction assigns channel bit widths while satisfying each layer's budget exactly.
- We derive a positive-semidefinite COD-GGN cost for local-frequency allocation. It balances foreground and background contributions and emphasizes multi-scale boundaries and uncertain pixels, using one first-order probe per layer and calibration batch.

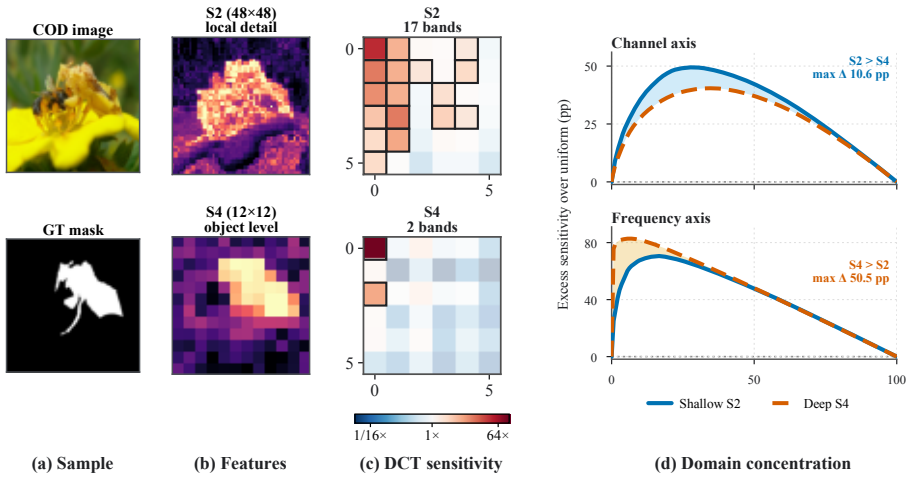


Figure 2: Depth-dependent sensitivity on a selected COD image. (a) shows the input and ground-truth mask, and (b) shows independently normalized S2 and S4 feature responses. In (c), 17 bands in S2 and two in S4 account for 80% of DCT sensitivity. (d) plots cumulative sensitivity above uniform versus the percentage of top-ranked channels or frequency bands, with separate vertical scales. Sensitivity is more concentrated across channels in S2 and across frequency bands in S4, supporting channel-wise allocation in shallow stages and local-frequency allocation in deep stages.

2 Related Work

Camouflaged object detection. COD methods aim to distinguish concealed targets from similar backgrounds while recovering fine object structures (Mei et al., 2021; Sun et al., 2022). Early approaches addressed target localization through classification-assisted segmentation in ANet (Le et al., 2019) and a search-and-identification framework in SINet (Fan et al., 2020). Later work introduced complementary information to recover less visible regions: ZoomNet fused features across image scales (Pang et al., 2022), while FDCOD incorporated frequency-domain cues (Zhong et al., 2022). Transformer-based methods combine semantic context and spatial detail. HitNet used high-resolution feedback to refine low-resolution representations (Hu et al., 2023), and CamoFormer progressively refined predictions through masked separable attention (Yin et al., 2024). Our work examines how to preserve foreground localization and detailed structures under low-bit quantization.

Post-training quantization. PTQ research addresses quantization errors through reconstruction and adapts quantizers to feature distributions through calibration (Banner et al., 2019; Liu et al., 2023). AdaRound and BRECQ match full-precision outputs by optimizing rounding at layer and block levels, respectively (Nagel et al., 2020; Li et al., 2021), while QDrop improves robustness to activation quantization (Wei et al., 2022). Vision Transformers require calibration tailored to attention and normalization layers (Ding et al., 2022). This motivates attention-ranking constraints (Liu et al., 2021b), distribution-aware quantizers in PTQ4ViT and FQ-ViT (Yuan et al., 2022; Lin et al., 2022), and scale reparameterization in RepQ-ViT (Li et al., 2023). These methods quantize existing feature representations; our work explores depth-dependent allocation domains.

Sensitivity-aware quantization. Quantization errors affect model predictions unequally (Dong et al., 2020; Tai & Wu, 2024). HAWQ uses Hessian information to allocate layer-wise precision (Dong et al., 2019). For reconstruction, FIMA-Q weights feature errors with diagonal and low-rank Fisher approximations (Wu et al., 2025a), while LS-ViT fits curvature through least squares (Hwang et al., 2026). COD-TDQ constrains token-group ranges to protect weak boundary cues from background-dominated quantization (Li et al., 2026). These methods consider information importance beyond numerical error, motivating allocation domains that expose task-relevant features at different depths.

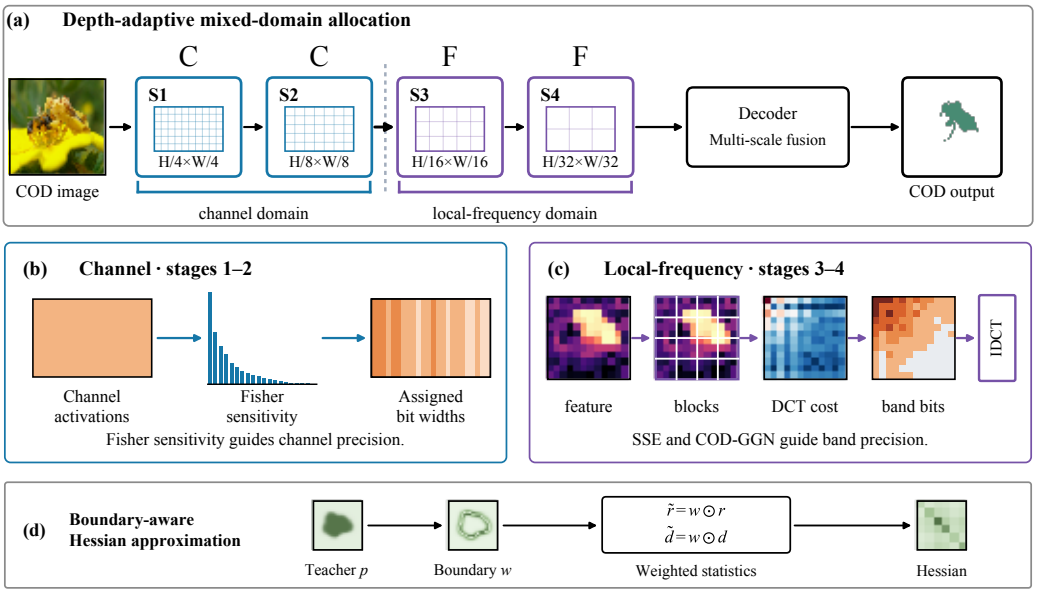


Figure 3: Overview of AxisQ for a four-stage hierarchical encoder. Channel-domain allocation is used in shallow and eligible non-frequency paths, while deep encoder features use local-frequency allocation. The strip and grid visualize channel-wise and frequency-wise activation-bit assignments at W2A2; darker orange indicates higher precision, and gray denotes zero-bit bands. (d) sketches the auxiliary boundary-aware Hessian approximation using teacher-derived boundary weights and weighted gradient statistics.

3 METHOD

This section presents our depth-adaptive mixed-domain quantization framework for hierarchical COD models. It covers channel-wise allocation in shallow stages, frequency-wise allocation in deep stages, and the corresponding reconstruction procedure.

3.1 DEPTH-ADAPTIVE MIXED-DOMAIN ALLOCATION

Low-bit quantization can suppress the weak foreground and boundary cues required for COD. Figure 2 suggests that sensitivity is more concentrated across channels in shallow features and across local-frequency bands in deep features, so we adapt the allocation domain to network depth.

Let $f(\boldsymbol{x};\boldsymbol{\theta})$ denote a trained hierarchical COD model, and let $\boldsymbol{A}_\ell \in \mathbb{R}^{B\times N_\ell \times C_\ell}$ denote the input activation of layer ℓ. We use an asymmetric b-bit uniform quantizer:

$$Q_b(a;s,z) = s\left[\mathrm{clip}\big(\mathrm{round}(a/s)+z, 0, 2^b-1\big) - z\right], \tag{1}$$

where s and z are the quantization scale and zero point, respectively. Weights are quantized at a fixed precision of W_b bits. The activation of layer ℓ is partitioned into K_ℓ allocation units, namely channels or local-frequency bands, and each unit k receives a bit width $b_{\ell,k}$ from a set of allowed tiers $\mathcal{B}$. We seek

$$\min_{\{b_{\ell,k}\}\subset\mathcal{B}} \sum_{k=1}^{K_\ell} \mathcal{D}_{\ell,k}(b_{\ell,k}) \quad \text{s.t.} \quad \frac{1}{K_\ell}\sum_{k=1}^{K_\ell} b_{\ell,k} = A_b, \tag{2}$$

where $\mathcal{D}_{\ell,k}(b)$ estimates how quantizing unit k at b bits affects COD predictions.

As shown in Figure 3, AxisQ instantiates Eq. (2) with two depth-specific branches followed by a shared reconstruction stage. Stages 1–2 use channels as units with label-assisted empirical-Fisher costs, while stages 3–4 use block-DCT local-frequency bands with costs that combine reconstruction error and COD-GGN sensitivity. The stage partition is fixed before calibration and retained across precision settings; frequency allocation takes precedence when both branches are applicable.

After calibration, the assignments are frozen for blockwise reconstruction, where the boundary-aware Hessian approximation serves as an auxiliary extension. Sections 3.2–3.4 develop these three components in order.

3.2 TASK-SENSITIVE CHANNEL-BIT ALLOCATION

Empirical-Fisher channel cost. We compute the supervised binary-cross-entropy loss on a small paired COD calibration subset. For layer ℓ, channel c, and candidate bit tier b, the per-channel empirical-Fisher distortion (Kunstner et al., 2019) is

$$D_{\ell,c}^{(b)} = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})}\left[\left(\sum_n \frac{\partial \mathcal{L}}{\partial \boldsymbol{A}_{\ell,n,c}}\left(Q_b(\boldsymbol{A}_{\ell,n,c}) - \boldsymbol{A}_{\ell,n,c}\right)\right)^2\right], \tag{3}$$

where n indexes spatial or token positions and any remaining non-channel dimensions, Q_b is the b-bit activation quantizer, and the expectation is taken over image-label pairs in the calibration subset. The inner sum approximates the first-order loss change caused by quantizing channel c; squaring and averaging this response gives the channel distortion used for bit allocation. The score retains interactions within a channel while ignoring cross-channel interactions. It uses only first derivatives and 64 ground-truth masks.

Reverse water filling. In Eq. (2), $\mathcal{D}_{\ell,c}(b) = D_{\ell,c}^{(b)}$. We use the reference-tier costs in a budget-preserving heuristic. Given a target average activation precision A_b, we initialize a continuous precision for each channel by adjusting A_b according to its log sensitivity relative to the layer mean:

$$\widetilde{b}_{\ell,c} = A_b + \gamma\left[\log_2\left(D_{\ell,c}^{(b_{\mathrm{ref}})} + \epsilon\right) - \frac{1}{C_\ell}\sum_{j=1}^{C_\ell}\log_2\left(D_{\ell,j}^{(b_{\mathrm{ref}})} + \epsilon\right)\right], \tag{4}$$

where b_{ref} is a reference bit tier, γ controls the strength of sensitivity-based reallocation, and ϵ improves numerical stability. Channels with larger distortion therefore receive higher initial precision. We round the continuous values to the allowed bit tiers and greedily adjust the assignments until $C_\ell^{-1}\sum_c b_{\ell,c} = A_b$ exactly. This layer-wise allocation gives more bits to sensitive channels while preserving the target average precision A_b for each layer.

3.3 COD-GGN TASK-AWARE FREQUENCY ALLOCATION

Block-frequency representation and reconstruction cost. For deep activations, we partition the spatial feature map into local blocks and apply an orthonormal 2-D DCT to each block. Coefficients at the same within-block frequency coordinate are grouped into a frequency band and share one bit width. Let $\mathcal{T}$ denote the block-DCT transform and $\boldsymbol{b}_\ell$ the band-wise bit assignment. The quantized activation is reconstructed as

$$\widehat{\boldsymbol{A}}_\ell = \mathcal{T}^{-1}(Q_{\boldsymbol{b}_\ell}(\mathcal{T}(\boldsymbol{A}_\ell))), \tag{5}$$

where $\mathcal{T}$ is the block-DCT transform.

For frequency band k and candidate bit tier b, let $u_{i,k}$ denote the DCT coefficient from calibration sample i. We estimate the reconstruction cost by

$$C_{k,b}^{\mathrm{SSE}} = \min_r \sum_i \left(Q_{b,r}(u_{i,k}) - u_{i,k}\right)^2. \tag{6}$$

where r indexes candidate clipping ranges. The minimum over r selects the clipping range that gives the smallest reconstruction error for each band and bit tier. This SSE term measures coefficient distortion without task weighting; the COD-aware cost introduced below accounts for the task sensitivity of different frequency bands. These SSE values provide a reconstruction-only reference for comparing candidate bit tiers before task sensitivity is incorporated. Each band is evaluated independently, so candidate tiers share a common reconstruction criterion.

COD-aware curvature and frequency cost. Reconstruction error weights all DCT coefficient errors equally, although COD is more sensitive to errors near object boundaries, uncertain predictions, and small foreground regions. From the full-precision teacher probability map $\boldsymbol{p} = \sigma(\boldsymbol{z})$, we define the output-space task metric

$$\begin{aligned} \Omega &= \mathrm{Norm}(R(\boldsymbol{p})[1 + \alpha B(\boldsymbol{p})]U(\boldsymbol{p}))\,, \\ B(\boldsymbol{p}) &= \frac{1}{|\mathcal{Q}|}\sum_{q\in\mathcal{Q}} \left|\mathrm{AvgPool}_q(\boldsymbol{p}) - \boldsymbol{p}\right|, \\ U(\boldsymbol{p}) &= \eta + (1-\eta)\,4\boldsymbol{p}(1-\boldsymbol{p}). \end{aligned} \tag{7}$$

Here R balances foreground and background mass, B captures multi-scale boundary variation, and U emphasizes uncertain predictions while retaining a nonzero floor. Together, these factors separate global mass balance from local task structure: R prevents large regions from dominating, while B and U highlight boundary changes and uncertain predictions. Their product is normalized independently for each image, so the probe represents relative task importance rather than output magnitude.

Using a Rademacher probe, we map this output-space metric to the activation at layer ℓ through a vector–Jacobian product (VJP). This produces a positive-semidefinite COD-aware generalized Gauss–Newton (COD-GGN) curvature surrogate without constructing a full Hessian (Martens, 2020). The probe response and the quantization error are expressed in the same block-DCT basis. For frequency band k and candidate bit tier b, we define the task-aware cost and its fusion with the reconstruction cost as

$$\begin{aligned} C_{k,b}^{\mathrm{task}} &= \sum_i (g_{i,k})^2 (\boldsymbol{e}_{i,k}^{(b)})^2, \\ C_{k,b} &= (1-\beta)\frac{C_{k,b}^{\mathrm{SSE}}}{\sum_j C_{j,b_{\mathrm{ref}}}^{\mathrm{SSE}}} + \beta\frac{C_{k,b}^{\mathrm{task}}}{\sum_j C_{j,b_{\mathrm{ref}}}^{\mathrm{task}}}. \end{aligned} \tag{8}$$

Here $g_{i,k}$ and $\boldsymbol{e}_{i,k}^{(b)}$ are the transformed probe response and quantization error for sample i and band k. A larger probe response makes the same quantization error more costly and therefore favors a higher bit tier. Reference-tier normalization keeps the two terms comparable, while β controls the influence of task sensitivity. With $\mathcal{D}_{\ell,k}(b) = C_{k,b}$ in Eq. (2), the Lagrangian relaxation gives $b_k = \arg\min_b(C_{k,b} + \lambda b)$. We then greedily correct the assignments so that their average equals A_b exactly.

The SSE term measures coefficient distortion, whereas the task term increases the penalty for errors in COD-sensitive directions. Setting $\beta = 0$ recovers a reconstruction-only frequency allocator; larger β shifts the allocation toward task-sensitive bands. After correction, the frozen band assignments are passed to the shared reconstruction procedure below. All task costs are computed during calibration and discarded after the assignments are frozen, so the selected quantizers introduce no additional inference-time COD computation.

3.4 Boundary-aware reconstruction

After allocations are frozen, we reconstruct the model block by block using the existing Hessian-based reconstruction objective (Hwang et al., 2026), AdaRound weight variables (Nagel et al., 2020), and QDrop (Wei et al., 2022). This pipeline is shared by matched baselines and AxisQ; COD-GGN affects only frequency calibration, while the boundary-aware extension is evaluated separately.

To make reconstruction boundary-aware, we reweight the gradient and error statistics before the existing Hessian fitting procedure. For calibration image i at block ℓ, let $\boldsymbol{r}_{\ell,i}$ be the absolute teacher-loss gradient and $\boldsymbol{d}_{\ell,i}$ its absolute deviation from the full-precision output. With the teacher probability map $\boldsymbol{p}_{\ell,i}$ resized to the block grid, we use

$$\begin{aligned} \boldsymbol{w}_{\ell,i} &= 1 + \alpha_{\mathrm{rec}} \left|\mathrm{AvgPool}_{k_\ell}(\boldsymbol{p}_{\ell,i}) - \boldsymbol{p}_{\ell,i}\right|, \\ \widetilde{\boldsymbol{r}}_{\ell,i} &= \boldsymbol{w}_{\ell,i} \odot \boldsymbol{r}_{\ell,i}, \qquad \widetilde{\boldsymbol{d}}_{\ell,i} = \boldsymbol{w}_{\ell,i} \odot \boldsymbol{d}_{\ell,i}. \end{aligned} \tag{9}$$

The weight map is expanded to the flattened block statistics. Larger local prediction variation therefore gives boundary-sensitive responses more weight in the curvature fit. Applying the same map to the gradient and error vectors preserves their correspondence with the original reconstruction statistics. AdaRound, QDrop, and final-logit BCE reconstruction remain unchanged. Because the weights come from the teacher prediction, no additional segmentation labels are required.

4 EXPERIMENTS

This section evaluates AxisQ across various COD models and benchmarks under different quantization settings. We describe the experimental setup, present comparisons with existing methods, and examine the contributions of individual components.

4.1 EXPERIMENTAL SETUP

Models and quantization settings. We quantize the Swin-based CFRN model (Song et al., 2025; Liu et al., 2021a) and CamoFormer-P (Yin et al., 2024; Wang et al., 2022) at input resolution 384×384. Models A and B in Figure 1 denote CFRN and CamoFormer-P, respectively. The main comparison evaluates both models at W2A2 and W4A4, with additional W2A3, W2A4, and W3A3 results on CFRN. These settings allow us to evaluate AxisQ's performance across different precision levels. Patch-embedding activations remain at A8 for every method and precision setting.

Datasets and evaluation metrics. Evaluation uses the standard CAMO, CHAMELEON, COD10K, and NC4K test sets (Le et al., 2019; Skurowski et al., 2018; Fan et al., 2020; Lv et al., 2021). We report S-measure S_α (Fan et al., 2017) for structural similarity and weighted F-measure F_β^w (Margolin et al., 2014) for spatially weighted precision and recall. Mean E-measure E_ϕ (Fan et al., 2018) assesses local and global alignment, averaged across thresholds. Maximum F-measure F_β^m is the highest F-score across thresholds, while MAE measures the mean absolute pixel-wise prediction error. Higher is better except for MAE. Macro metrics give equal weight to each dataset. Macro quality averages S_α, F_β^w, E_ϕ, F_β^m, and $1 - \mathrm{MAE}$ with equal weights across metrics and datasets.

Implementation details. All calibration samples are drawn from the training set. The calibration pool contains 128 images, processed with batch size 4. We use 64 paired image–mask samples with batch size 2 to estimate channel-domain Fisher costs, and 16 images to estimate deep frequency and COD-GGN costs. Block reconstruction uses 1,024 images, an effective batch size of 32, and 20,000 optimization iterations per block. QDrop is applied with probability 0.5. We set the channel allocation spread to $\gamma = 0.30$ and use 12×12 DCT blocks with eight candidate clipping ratios. For COD-GGN, we use pooling sizes $\mathcal{Q} = \{3, 7, 15\}$, boundary weight $\alpha = 2$, uncertainty floor $\eta = 0.1$, and cost mixing weight $\beta = 0.10$. The output weights are normalized to mean one per image and capped at 8. Both frequency cost terms are normalized at the reference tier $b_{\mathrm{ref}} = 2$. These settings, the stage split, and the reconstruction length remain fixed across precision budgets.

Compared methods. We compare with LS-ViT (Hwang et al., 2026), BRECQ (Li et al., 2021), AdaRound+QDrop (Nagel et al., 2020; Wei et al., 2022), FIMA-Q (Wu et al., 2025a), and APHQ-ViT (Wu et al., 2025b) on both models, and additionally with RepQ-ViT (Li et al., 2023) and COD-TDQ (Li et al., 2026) on CFRN. These methods cover calibration-based quantization, adaptive rounding, and curvature-based reconstruction. All comparator results use local COD adaptations. Reconstruction-based methods use 20,000 optimization iterations per block; RepQ-ViT and COD-TDQ use no reconstruction. COD-TDQ retains full-precision attention matrix multiplications; its W2A2 row is a local four-level affine extension, not an author-released W2A2 result. FP32 results provide the full-precision reference for each model.

4.2 QUANTITATIVE RESULTS

AxisQ delivers substantial gains under W2A2 quantization across both backbones, as shown in Table 1. It leads all 20 dataset–metric comparisons on CFRN and 19 on CamoFormer-P. On CAMO, its weighted F-measure exceeds the best competing result by 0.082 on CFRN and 0.125 on CamoFormer-P. The consistent improvements across models and evaluation measures support the effectiveness of AxisQ in preserving segmentation quality under a tight bit budget.

The advantage remains consistent at W4A4. At the reported three-decimal precision, AxisQ achieves the best quantized result in 33 of the 40 dataset–metric comparisons and ties for the best in the remaining seven. This shows that its benefit extends to higher-precision quantization, where

Table 1: Quantitative comparison on CFRN and CamoFormer-P across four COD benchmarks at W4A4 and W2A2. FP32 denotes the full-precision reference. Bold indicates the best quantized results within each model and precision setting.

Method	CAMO					CHAMELEON					COD10K					NC4K				
	$S_\alpha\uparrow$	$F_\beta^w\uparrow$	$E_\phi\uparrow$	$F_\beta^m\uparrow$	MAE↓	$S_\alpha\uparrow$	$F_\beta^w\uparrow$	$E_\phi\uparrow$	$F_\beta^m\uparrow$	MAE↓	$S_\alpha\uparrow$	$F_\beta^w\uparrow$	$E_\phi\uparrow$	$F_\beta^m\uparrow$	MAE↓	$S_\alpha\uparrow$	$F_\beta^w\uparrow$	$E_\phi\uparrow$	$F_\beta^m\uparrow$	MAE↓
CFRN																				
FP32	.876	.843	.932	.877	.042	.912	.874	.959	.898	.020	.870	.794	.937	.835	.022	.888	.849	.939	.880	.030
W4A4																				
RepQ-ViT	.676	.608	.750	.656	.099	.701	.622	.774	.672	.067	.676	.549	.779	.599	.062	.718	.651	.783	.691	.072
COD-TDQ	.813	.723	.864	.795	.070	.861	.759	.906	.825	.040	.802	.650	.864	.737	.038	.836	.746	.883	.816	.052
LS-ViT	.860	.816	.916	.855	.048	.906	.857	.951	.887	.023	.850	.754	.915	.802	.026	.875	.824	.925	.862	.035
BRECQ	.864	.823	.919	.861	.047	.907	.857	**.956**	.883	.022	.859	.769	.923	.815	.024	.881	.834	.930	.870	.033
AdaRound+QDrop	.863	.823	.917	.861	.047	.907	.860	.951	.886	.022	.859	.771	.923	.815	.024	.881	.836	.929	.871	.033
FIMA-Q	.865	.825	.920	.863	.046	.907	.861	.951	.888	.022	.859	.772	.923	.816	.024	.881	.836	.929	.871	.033
APHQ-ViT	.866	.827	.921	.864	.046	.907	.859	.951	.886	.022	.859	.772	.923	.817	**.024**	.882	.837	.930	.872	**.033**
AxisQ (ours)	**.869**	**.832**	**.924**	**.870**	**.044**	**.909**	**.865**	**.956**	**.893**	**.021**	**.861**	**.775**	**.925**	**.820**	**.024**	**.883**	**.838**	**.932**	**.873**	**.033**
W2A2																				
COD-TDQ (ext.)	.261	.162	.244	.218	.614	.239	.133	.236	.171	.662	.249	.082	.234	.112	.648	.258	.138	.240	.185	.629
RepQ-ViT	.386	.106	.485	.218	.247	.400	.085	.511	.171	.215	.421	.061	.534	.111	.175	.399	.093	.503	.185	.224
BRECQ	.702	.559	.764	.641	.113	.779	.625	.839	.709	.066	.734	.521	.803	.607	.059	.787	.659	.841	.736	.072
AdaRound+QDrop	.713	.583	.749	.668	.102	.803	.681	.859	.750	.052	.759	.580	.806	.654	.045	.805	.700	.845	.767	.060
FIMA-Q	.729	.612	.767	.694	.097	.804	.686	.851	.756	.048	.766	.596	.819	.668	.044	.814	.716	.856	.779	.057
APHQ-ViT	.743	.631	.785	.707	.093	.823	.712	.874	.776	.047	.774	.608	.828	.679	.042	.818	.723	.861	.785	.056
LS-ViT	.746	.645	.794	.725	.089	.826	.722	.873	.786	.045	.782	.624	.836	.693	.040	.824	.735	.867	.794	.054
AxisQ (ours)	**.798**	**.727**	**.861**	**.787**	**.071**	**.851**	**.766**	**.924**	**.809**	**.036**	**.793**	**.654**	**.862**	**.713**	**.036**	**.834**	**.760**	**.890**	**.810**	**.048**
CamoFormer-P																				
FP32	.873	.827	.926	.868	.046	.898	.845	.941	.884	.026	.860	.770	.920	.814	.025	.887	.837	.931	.872	.032
W4A4																				
LS-ViT	.861	.810	.914	.853	.050	.892	.833	.935	.876	.029	.850	.748	.909	.798	.027	.879	.822	.923	.862	.035
BRECQ	.858	.805	.911	.849	.051	.889	.828	.935	.870	.029	.846	.742	.906	.790	.028	.877	.820	.922	.859	.035
AdaRound+QDrop	.863	.813	.914	.857	.049	.891	.833	.935	.876	.028	.849	.748	.908	.797	.027	.879	.823	.923	.862	**.034**
FIMA-Q	.863	.812	.914	.857	.050	.892	.833	.937	.876	.028	.849	.748	.909	.797	.027	.879	.822	.923	.862	.035
APHQ-ViT	.862	.810	.914	.853	.049	.891	.833	.937	**.876**	**.028**	.850	.750	.908	.798	**.027**	.880	.823	.924	.862	.035
AxisQ (ours)	**.867**	**.817**	**.921**	**.858**	**.047**	**.893**	**.834**	**.938**	**.876**	**.028**	**.851**	**.751**	**.913**	**.800**	**.027**	**.881**	**.825**	**.925**	**.864**	**.034**
W2A2																				
BRECQ	.574	.350	.654	.462	.177	.606	.354	.659	.464	.136	.606	.302	.683	.408	.117	.653	.425	.714	.542	.137
AdaRound+QDrop	.576	.347	.608	.456	.164	.630	.398	.652	.513	.108	.645	.367	.712	.468	**.092**	.698	.499	.745	.602	.110
FIMA-Q	.581	.362	.629	.462	.167	.642	.412	.692	.507	.110	.640	.363	.709	.468	.102	.698	.501	.751	.607	.116
APHQ-ViT	.593	.371	.635	.469	.163	.637	.404	.690	.499	.113	.645	.368	.711	.480	.100	.704	.505	.755	.617	.114
LS-ViT	.598	.375	.631	.484	.179	.636	.394	.671	.507	.134	.632	.340	.687	.470	.122	.696	.479	.736	.610	.131
AxisQ (ours)	**.671**	**.500**	**.727**	**.596**	**.146**	**.716**	**.524**	**.771**	**.625**	**.104**	**.671**	**.419**	**.734**	**.539**	.101	**.735**	**.560**	**.788**	**.675**	**.106**

reconstruction-based methods already perform closely. Although the margins are smaller than at W2A2, AxisQ maintains a consistently strong position across both models and all four datasets.

The two backbones show different sensitivity to low-bit quantization. Their FP32 mean weighted F-measures are relatively close, but CamoFormer-P suffers a larger drop under the W2A2 LS-ViT baseline. AxisQ raises mean weighted F-measure over this baseline by approximately 0.104 on CamoFormer-P, compared with 0.045 on CFRN. This larger recovery on the more affected backbone further demonstrates AxisQ's ability to alleviate severe quantization degradation across models.

Table 2 further demonstrates the applicability of AxisQ across different precision budgets on CFRN. With shared allocation hyperparameters, it improves 94 of the 100 dataset–metric comparisons against precision-matched baselines, including all 20 comparisons at W2A2, W3A3, and W4A4. With weights fixed at two bits, mean weighted F-measure gains remain positive as activation precision increases from two to four bits. These results support using the same allocation strategy across different combinations of weight and activation precision.

4.3 Ablation studies and allocation analysis

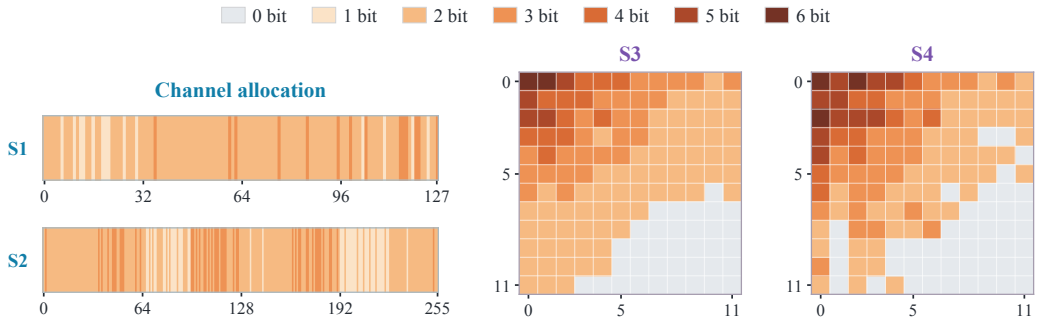
Table 3 suggests that the allocation domain should be chosen according to feature depth. The mixed-domain no-GGN variant improves macro weighted F-measure by 0.025 over the stronger frequency-only control, exceeding that control's 0.013 gain over the baseline. This gap is consistent with the complementary roles of channel sensitivity in shallow features and frequency structure in deep features. The accompanying reduction in MAE supports an overall improvement in prediction quality.

Table 2: AxisQ gains across precisions.

Setting	ΔS_α	ΔF_β^w	ΔE_ϕ	ΔF_β^m	ΔMAE	Wins
W2A2	+.025	+.045	+.042	+.030	-.009	20/20
W2A3	+.006	+.009	+.008	+.008	-.002	18/20
W2A4	+.002	+.004	+.001	+.005	.000	16/20
W3A3	+.006	+.011	+.010	+.008	-.002	20/20
W4A4	+.008	+.015	+.008	+.012	-.003	20/20

Table 3: W2A2 quantization comparison.

Method	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$E_\phi \uparrow$	$F_\beta^m \uparrow$	MAE$\downarrow$
Baseline	.795	.681	.842	.749	.057
Channel-only	.799	.688	.848	.749	.055
Frequency-only	.802	.694	.854	.758	.054
AxisQ no GGN	.813	.719	.875	.774	.050
AxisQ	**.819**	**.727**	**.884**	**.780**	**.048**



(a) Shallow channels (b) Deep frequency bands

Figure 4: Actual CFRN W2A2 bit assignments in the first attention-output projection of each encoder stage. Channels retain their original order; frequency maps use DCT coordinates. Every shown layer averages two bits. Gray indicates suppressed, zero-bit bands.

The full configuration improves structural agreement and prediction alignment while reducing pixel-wise error. These gains suggest improvements in both the overall structure of the predicted masks and their pixel-level accuracy. The improvements over no-GGN favor the complete design. Boundary-weighted reconstruction shows a similar pattern in the separate CFRN W2A4 comparison on CHAMELEON, supporting the relevance of boundary-focused reconstruction to overall mask quality.

Figure 4 visualizes CFRN's frozen W2A2 bit assignments. Most shallow channels retain two bits, with selected channels adjusted to one or three bits. Deep layers show greater variation across frequency bands: lower frequencies generally receive more precision, while part of the high-frequency region is suppressed with zero-bit assignments. Each displayed layer maintains an exact two-bit average. The differences between the S3 and S4 maps reflect layer-specific calibration costs, allowing each deep layer to distribute precision differently within the same frequency domain. This pattern illustrates how AxisQ redistributes a limited precision budget across different feature components, using channels in shallow stages and local frequencies in deep stages.

5 Conclusion

We presented AxisQ, a depth-adaptive post-training quantization framework for camouflaged object detection. Motivated by the changing characteristics of hierarchical features, AxisQ allocates activation precision across shallow channels and deep local-frequency bands under a fixed per-layer average-bit budget. Empirical-Fisher channel costs and COD-aware frequency costs guide this allocation toward components important for foreground and boundary prediction. We also formulated an auxiliary boundary-aware Hessian approximation that incorporates boundary-weighted gradient and error statistics into reconstruction. Experiments across COD models, benchmarks, and precision settings demonstrate the effectiveness of the mixed-domain allocation. This work highlights the importance of adapting both the allocation domain and its sensitivity measure to preserve task-relevant information in low-bit hierarchical COD models.

Ethics statement

This work studies low-bit quantization for camouflaged object detection using publicly available benchmarks. It involves no human-subject studies or collection of new personal data. Although efficient COD can support applications such as wildlife monitoring, its potential use in surveillance and military applications warrants consideration.

Reproducibility statement

We provide the implementation and configuration files for AxisQ, along with environment requirements and instructions for quantization and evaluation. These materials document the calibration data, precision settings, and reconstruction procedure, and include commands for evaluating the models on all four COD benchmarks.

AI use statement

We used generative AI tools to assist with code implementation and debugging, experimental analysis, and manuscript preparation, including English editing and LaTeX formatting. Reported results are based on experimental measurements and cited sources. The authors take responsibility for the accuracy and originality of the final manuscript and accompanying code.

References

Ron Banner, Yury Nahshan, and Daniel Soudry. Post training 4-bit quantization of convolutional networks for rapid-deployment. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Yifu Ding, Haotong Qin, Qinghua Yan, Zhenhua Chai, Junjie Liu, Xiaolin Wei, and Xianglong Liu. Towards accurate post-training quantization for vision transformer. In *Proceedings of the 30th ACM International Conference on Multimedia*, pp. 5380–5388, 2022. doi: 10.1145/3503161.3547826.

Zhen Dong, Zhewei Yao, Amir Gholami, Michael W. Mahoney, and Kurt Keutzer. HAWQ: Hessian AWare quantization of neural networks with mixed-precision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 293–302, 2019.

Zhen Dong, Zhewei Yao, Daiyaan Arfeen, Amir Gholami, Michael W. Mahoney, and Kurt Keutzer. HAWQ-V2: Hessian aware trace-weighted quantization of neural networks. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Deng-Ping Fan, Ming-Ming Cheng, Yun Liu, Tao Li, and Ali Borji. Structure-measure: A new way to evaluate foreground maps. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4548–4557, 2017.

Deng-Ping Fan, Cheng Gong, Yang Cao, Bo Ren, Ming-Ming Cheng, and Ali Borji. Enhanced-alignment measure for binary foreground map evaluation. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pp. 698–704, 2018. doi: 10.24963/ijcai.2018/97.

Deng-Ping Fan, Ge-Peng Ji, Guolei Sun, Ming-Ming Cheng, Jianbing Shen, and Ling Shao. Camouflaged object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2777–2787, 2020.

Xiaobin Hu, Shuo Wang, Xuebin Qin, Hang Dai, Wenqi Ren, Donghao Luo, Ying Tai, and Ling Shao. High-resolution iterative feedback network for camouflaged object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 881–889, 2023. doi: 10.1609/aaai.v37i1.25167.

Hyunha Hwang, Xuan Truong Nguyen, and Hyuk-Jae Lee. Least-squares hessian based block reconstruction for low-bit post-training quantization of vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 33588–33597, 2026.

Frederik Kunstner, Lukas Balles, and Philipp Hennig. Limitations of the empirical Fisher approximation for natural gradient descent. *arXiv preprint arXiv:1905.12558*, 2019. URL https://arxiv.org/abs/1905.12558.

Trung-Nghia Le, Tam V. Nguyen, Zhiqiang Nie, Minh-Triet Tran, and Akihiro Sugimoto. Anabranch network for camouflaged object segmentation. *Computer Vision and Image Understanding*, 184:45–56, 2019. doi: 10.1016/j.cviu.2019.04.006.

Tianqi Li, Wenyu Fang, Xin He, Xue Geng, Xu Cheng, and Yun Liu. When W4A4 breaks camouflaged object detection: Token-group dual-constraint activation quantization. *arXiv preprint arXiv:2604.16855*, 2026.

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 *International Conference on Learning Representations*, 2021.

Zhikai Li, Junrui Xiao, Lianwei Yang, and Qingyi Gu. RepQ-ViT: Scale reparameterization for post-training quantization of vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17227–17236, 2023.

Yang Lin, Tianyu Zhang, Peiqin Sun, Zheng Li, and Shuchang Zhou. FQ-ViT: Post-training quantization for fully quantized vision transformer. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, pp. 1173–1179, 2022.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021a.

Zhenhua Liu, Yunhe Wang, Kai Han, Wei Zhang, Siwei Ma, and Wen Gao. Post-training quantization for vision transformer. In *Advances in Neural Information Processing Systems*, volume 34, 2021b.

Yunqiu Lv, Jing Zhang, Yuchao Dai, Aixuan Li, Bowen Liu, Nick Barnes, and Deng-Ping Fan. Simultaneously localize, segment and rank the camouflaged objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11591–11601, 2021.

Ran Margolin, Lihi Zelnik-Manor, and Ayellet Tal. How to evaluate foreground maps? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2014.

James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020. URL https://jmlr.org/papers/v21/17-678.html.

Haiyang Mei, Ge-Peng Ji, Ziqi Wei, Xin Yang, Xiaopeng Wei, and Deng-Ping Fan. Camouflaged object segmentation with distraction mining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8772–8781, 2021.

Markus Nagel, Rana Ali Amjad, Mart van Baalen, Christos Louizos, and Tijmen Blankevoort. Up or down? adaptive rounding for post-training quantization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 7197–7206, 2020.

Youwei Pang, Xiaoqi Zhao, Tian-Zhu Xiang, Lihe Zhang, and Huchuan Lu. Zoom in and out: A mixed-scale triplet network for camouflaged object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2160–2170, 2022.

Przemysław Skurowski, Hassan Abdulameer, Jakub Błaszczyk, Tomasz Depta, Adam Kornacki, and Przemysław Kozieł. Animal camouflage analysis: CHAMELEON database. Unpublished manuscript, 2018. URL https://www.polsl.pl/rau6/chameleon-database-animal-camouflage-analysis/.

Ze Song, Xudong Kang, Xiaohui Wei, Jinyang Liu, Zheng Lin, and Shutao Li. Continuous feature representation for camouflaged object detection. *IEEE Transactions on Image Processing*, 34: 5672–5685, 2025. doi: 10.1109/TIP.2025.3602657.

Yujia Sun, Shuo Wang, Chenglizhao Chen, and Tian-Zhu Xiang. Boundary-guided camouflaged object detection. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, pp. 1335–1341, 2022. doi: 10.24963/ijcai.2022/186.

Yu-Shan Tai and An-Yeu (Andy) Wu. MPTQ-ViT: Mixed-precision post-training quantization for vision transformer. *arXiv preprint arXiv:2401.14895*, 2024. URL https://arxiv.org/abs/2401.14895.

Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. PVT v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3):415–424, 2022. doi: 10.1007/s41095-022-0274-8.

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

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9686–9695, 2025b.

Bowen Yin, Xuying Zhang, Deng-Ping Fan, Shaohui Jiao, Ming-Ming Cheng, Luc Van Gool, and Qibin Hou. CamoFormer: Masked separable attention for camouflaged object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Zhihang Yuan, Chenhao Xue, Yiqi Chen, Qiang Wu, and Guangyu Sun. PTQ4ViT: Post-training quantization for vision transformers with twin uniform quantization. In *European Conference on Computer Vision*, pp. 191–207, 2022.

Yijie Zhong, Bo Li, Lv Tang, Senyun Kuang, Shuang Wu, and Shouhong Ding. Detecting camouflaged object in frequency domain. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4504–4513, 2022.

A EXPECTATION OF THE COD-GGN PROBE

COD-GGN uses teacher-derived output weights to account for foreground–background balance, boundaries, and prediction uncertainty during frequency allocation. Here we explain how a random probe transfers these weights to layer activations using a vector–Jacobian product (VJP) via first-order backpropagation, without explicitly forming the Jacobian or Hessian.

For a fixed calibration image, let $\boldsymbol{J}_\ell$ be the Jacobian of the $H \times W$ output logits with respect to the flattened activation at layer ℓ. The model and the nonnegative teacher weights Ω in Equation 7 are held fixed; the expectation is over independent Rademacher entries of $\boldsymbol{\epsilon}$, each equally likely to be -1 or $+1$. Thus $\mathbb{E}[\boldsymbol{\epsilon}] = 0$ and $\mathbb{E}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^\top] = \boldsymbol{I}$. Defining $\boldsymbol{v} = \operatorname{diag}(\Omega)^{1/2}\boldsymbol{\epsilon}/\sqrt{HW}$ and $\boldsymbol{g}_\ell = \boldsymbol{J}_\ell^\top \boldsymbol{v}$ gives

$$\mathbb{E}[\boldsymbol{g}_\ell \boldsymbol{g}_\ell^\top] = \boldsymbol{J}_\ell^\top \mathbb{E}[\boldsymbol{v}\boldsymbol{v}^\top]\boldsymbol{J}_\ell \tag{10}$$

$$= \frac{1}{HW}\boldsymbol{J}_\ell^\top \operatorname{diag}(\Omega)\boldsymbol{J}_\ell. \tag{11}$$

Because $\Omega \geq 0$, the expected matrix is positive semidefinite. After expressing $\boldsymbol{g}_\ell$ in the orthonormal block-DCT basis, its component-wise squares estimate the diagonal curvature in that basis. These sensitivities weight the coefficient errors in Equation 8, making errors in task-sensitive directions more costly. The allocation cost uses this diagonal approximation and omits cross-coordinate curvature interactions.

B DETAILED GAINS OVER MATCHED BASELINES

Table 4 expands the CFRN comparison in Table 2 into per-dataset gains across five precision settings. Each entry reports AxisQ minus the precision-matched baseline, exposing variation hidden by the macro averages. Gains are positive for the first four metrics and negative for MAE.

Table 4: Per-dataset metric changes of AxisQ relative to the precision-matched baseline, rounded to three decimal places.

Setting	Dataset	ΔS_α	ΔF_β^w	ΔE_ϕ	ΔF_β^m	ΔMAE
W2A2	CAMO	+.052	+.082	+.067	+.062	-.017
	CHAMELEON	+.025	+.044	+.051	+.023	-.009
	COD10K	+.011	+.030	+.026	+.020	-.003
	NC4K	+.010	+.025	+.023	+.016	-.005
W2A3	CAMO	+.017	+.026	+.019	+.024	-.005
	CHAMELEON	+.001	+.001	+.006	.000	.000
	COD10K	+.002	+.004	+.002	+.004	-.001
	NC4K	+.003	+.004	+.005	+.004	-.001
W2A4	CAMO	+.006	+.009	+.009	+.009	-.003
	CHAMELEON	.000	+.002	-.004	+.004	+.001
	COD10K	+.001	+.003	+.001	+.005	.000
	NC4K	.000	+.001	+.001	+.001	.000
W3A3	CAMO	+.016	+.023	+.019	+.020	-.005
	CHAMELEON	+.003	+.007	+.007	+.004	-.001
	COD10K	+.004	+.008	+.008	+.006	-.001
	NC4K	+.002	+.004	+.005	+.002	-.001
W4A4	CAMO	+.009	+.017	+.009	+.014	-.004
	CHAMELEON	+.003	+.008	+.005	+.005	-.002
	COD10K	+.010	+.022	+.010	+.018	-.003
	NC4K	+.008	+.014	+.007	+.011	-.003

Gains peak at W2A2 and generally shrink as activation precision rises with two-bit weights, consistent with allocation being most useful under tight budgets. Yet W4A4 gives larger weighted F-

measure gains than W2A4 on every dataset, showing that activation precision alone does not explain the benefit. CAMO leads in four settings and COD10K at W4A4.

Weighted F-measure improves in all 20 dataset–precision pairs, while MAE decreases in 16. These gains generally combine better foreground prediction with lower pixelwise error, although the small MAE increase on CHAMELEON at W2A4 shows that improvement is not uniform across metrics.

C Fixed hyperparameters and quantization scope

Table 5 summarizes the calibration, reconstruction, and allocation settings shared by the main mixed-domain experiments. The stated activation precision is an exact per-layer average over eligible channels or frequency bands, allowing individual units to receive different bit widths. The reconstruction component study follows the separate protocol in Appendix D.

Fisher allocation covers eligible non-frequency Conv/Linear paths, including the decoder; DCT allocation covers frequency-eligible stage 3–4 Linear layers. Zero-bit bands are suppressed and patch-embedding activations remain A8. Unwrapped operations follow the reconstruction baseline.

Table 5: Configuration used for the main mixed-domain runs.

Item	Value
Input resolution	384×384
Calibration / reconstruction images	128 / 1,024
Fisher image–mask pairs / frequency images	64 / 16
Reconstruction iterations	20,000 per block
Reconstruction method	Least-squares Hessian + AdaRound + QDrop
QDrop probability / temperature	0.5 / 20
Channel / DCT encoder stages	1–2 / 3–4
Channel allocation spread γ	0.30
DCT block / bands	12×12 / 144
Deep bit tiers	$\{0, 2, 3, 4, 5, 6\}$
Frequency clip candidates	8
COD-GGN blend β	0.10
Boundary kernels / weight	$\{3, 7, 15\}$ / 2.0
Uncertainty floor / curvature cap	0.10 / 8.0
Frequency normalization reference	2 bits for every precision
Patch embedding activation	8 bits for every precision

D Boundary-reconstruction component protocol

Table 6 uses a separate CFRN W2A4 protocol. Both runs restore the same calibration checkpoint and use identical adaptive iteration counts across 35 blocks (153,800 iterations total). Their shared teacher BCE uses pixel weights $1 + 10|\,\mathrm{AvgPool}_9(\boldsymbol{p}) - \boldsymbol{p}|$, normalized by the sum within each image. Only the variant adds the local Hessian weighting in Equation 9, with $\alpha_{\mathrm{rec}} = 2$ and kernels 7/5/3 for patch embedding/stages 1/2.

Table 6: Historical CFRN W2A4 reconstruction on CHAMELEON, with and without local boundary weights.

Boundary weights	$S_\alpha \uparrow$	$F_\beta^w \uparrow$	$E_\phi \uparrow$	$F_\beta^m \uparrow$	MAE$\downarrow$
Without	.897	.841	.942	.878	.027
With	.901	.844	.946	.878	.025

Table 7: AxisQ latency on CamoFormer-P under CUDA Graph replay (RTX 5090). Speedup is relative to FP32.

Implementation	Latency (ms)$\downarrow$	FPS$\uparrow$	Speedup$\uparrow$
FP32	17.13	58.4	1.00$\times$
W4A4 (integer)	15.95	62.7	1.07$\times$
W4A4 (FP16)	**11.28**	**88.6**	**1.52**$\times$

E W4A4 DEPLOYMENT PERFORMANCE

Deployment gains. With CUDA Graph replay (Table 7), our fused W4A4 implementation runs CamoFormer-P at 88.6 FPS (11.28 ms), achieving 3.33× and 1.52× speedups over the original PyTorch quantization simulation and FP32 reference, respectively. These correspond to latency reductions of 70.0% and 34.1%. The packed-W4 integer-GEMM prototype reaches 62.7 FPS, a 2.36× speedup over the simulation. Both reuse the saved AxisQ bit assignments and quantization parameters without recalibration or reconstruction.

Accuracy after deployment. On all 250 CAMO test images, the fused implementation achieves $F_\beta^w = 0.8164$ and $S_\alpha = 0.8670$, compared with 0.8179 and 0.8679 for the original quantized reference, respectively. The packed-W4 integer-GEMM prototype achieves $F_\beta^w = 0.8191$ and $S_\alpha = 0.8691$. Both implementations retain comparable segmentation accuracy. Metrics come from fresh checkpoint replays; floating-point casts and summation order cause small numerical differences.

Implementation and timing. The fused path caches quantized weights and fuses quantization/dequantization (Q/DQ), retaining DCT/IDCT and using FP16 GEMM. The packed-W4 path adapts the COD-TDQ Triton backend (Li et al., 2026): weights are unpacked into INT8 registers for Tensor Core products with INT32 accumulation. It covers all 205 Linear layers and 76 pointwise convolutions, with FP16 outputs and remaining operators. A4 denotes the per-layer average activation budget, including mixed tiers.

Measurements use PyTorch 2.8.0, CUDA 12.8, Triton 3.4.0, and driver 580.76.05 on an RTX 5090 (32 GB), with cuDNN autotuning. All implementations use batch size 1, 384×384 inputs, and disabled TF32; FPS is the reciprocal of latency. We average five rounds of 100 complete forwards after 30 warm-up calls using synchronized CUDA events. Timing includes channel gathering and padded computation, Q/DQ, DCT/IDCT, the decoder, and all prediction heads. Inputs reside on the GPU; I/O, preprocessing, postprocessing, compilation, graph capture, and offline weight packing are excluded.