Adaptive Quantization of CNNs for Spectrogram-Based Foreground Activity Classification

Ashitabh Misra, Madhav Agrawal, Tomoyoshi Kimura, Jinyang Li*, Avaljot Singh, Arham Jain, Tarek Abdelzaher
Department of Computer Science
University of Illinois Urbana Champaign
{misra8, madhav5, tkimura4, jinyang7, avaljot2, arhamj3, zaher}@illinois.edu

***Abstract*—Adaptive Quantization (AQ) enables mixed-precision computation in convolutional neural networks (CNNs) without retraining, reducing model size and inference cost for resource-constrained deployments. Existing AQ methods, however, typically assign a single bit-width to the entire input. While adequate for many vision tasks with approximate translation invariance, this assumption is ill-suited for spectrogram-based representations, where different frequency regions carry distinct semantic meaning. A single bit-width for the entire input can therefore misallocate precision, over-allocating bits for noise-dominant frequencies and under-allocating them for informative foreground signals. We propose FREQQUANT, an adaptive quantization framework for CNNs operating on spectrogram inputs for foreground activity classification. FREQQUANT incorporates input signal statistics into the quantization optimization to derive a preference metric that assigns frequency-band-specific bit-widths on a per-layer basis. This enables precision to be allocated where it most impacts performance without increasing computational cost. Experiments on vehicle and human activity classification demonstrate consistent improvements over state-of-the-art AQ methods across multiple CNN architectures under comparable compute budgets.**

***Index Terms*—Adaptive quantization, mixed precision, CNNs, spectrograms, edge inference**

I. INTRODUCTION

Deploying Convolutional Neural Networks (CNNs) on resource-constrained edge devices requires model compression to meet strict memory and latency budgets. Among compression techniques such as pruning [1], [2], sparsification, and knowledge distillation [3], [4], quantization has proven particularly effective. Quantization reduces model size by using low-precision weights and activations [5]–[9]. While Transformers have become dominant for many tasks [10], CNNs remain a popular choice of architecture for real-time signal processing on edge devices, particularly for applications involving waveform modalities [11].

Waveform time-series signals, including audio, seismic, accelerometer, and gyroscope data, are common inputs for edge applications such as voice assistants and activity recognition [11], [12]. These signals are typically transformed into spectrogram-based time-frequency representation and processed by CNNs for classification. We focus on *foreground activity classification* — tasks in which the target of interest produces structured, high-energy signals (e.g., vehicle detection, human activity recognition) that stand out against ambient noise. This is distinct from detecting faint or low-energy targets, which our approach does not address. While quantization has been studied extensively for image and video tasks, less attention has been given to time-frequency representations such as spectrograms. We find that directly applying existing quantization methods to spectrograms yields sub-optimal accuracy.

Existing Techniques. Conventional quantization methods convert full-precision (32-bit) DNNs into fixed lower bit-width representations, such as 8-bit or 4-bit weights and activations. Edge devices operate under dynamic conditions: battery-powered systems may need to reduce computation under low power, while latency requirements may tighten under heavy system load. This makes static quantization impractical since switching bit-widths requires retraining. To address this challenge, *Adaptive Quantization* (AQ) techniques [6], [7], [13]–[15] enable a single model to operate across multiple bit allocations *without retraining*. Despite their flexibility, we identify two limitations when applying existing AQ methods to spectrogram-based representations.

First, we observe that existing AQ techniques exhibit significant accuracy variance across different bit allocations for foreground activity detection tasks — including Vehicle Classification (VC) and Human Activity Recognition (HAR) (Section III, Table I). This instability forces users to sample and evaluate numerous bit allocation configurations to identify one that meets their resource constraints and accuracy requirements. Beyond being computationally expensive, such sampling becomes infeasible when sample data is limited or unavailable during deployment.

Second, existing AQ techniques apply the same bit-width across the entire input when quantizing a layer's feature maps

*Corresponding author: Jinyang Li (jinyang7@illinois.edu)

and weights. For vision tasks, this is reasonable since semantic content can appear anywhere in the image (translation invariance). However, spectrograms have an inherent structure along the frequency axis: different frequency regions exhibit varying signal-to-noise ratios (SNR), with some containing structured information and others dominated by noise. An instance of this is shown in Figure 2 where region B is dominated by noise, and region A consists of a structured signal. A single bit-width therefore over-allocates bits to noisy regions while under-allocating to informative ones.

Together, these observations motivate *a frequency-aware, spatially adaptive quantization scheme* that allocates bits *non-uniformly* across the spectrogram to stabilize accuracy across bit-allocation choices and removes the need for costly deployment-time bit-search.

This Work. FREQQUANT addresses the limitations through band-wise mixed-precision quantization across the frequency axis: within each convolutional layer, frequency bands are quantized at bit-widths determined by their SNR, allocating fewer bits to noisy (low SNR) bands and more bits to informative (high SNR) ones. During training, for each layer and frequency band, FREQQUANT learns a probability distribution over bit-widths. At inference, the learned distribution serves two purposes: (1) a greedy algorithm uses it to select a bit allocation for a given resource budget (referred to as *data-free bit allocation*); and (2) when test data is available, sampling from this distribution yields higher average accuracy than uniform sampling (referred to as *sampling-based bit allocation*).

During training, FREQQUANT optimizes a three-term loss: (1) cross-entropy for task accuracy; (2) a weighted distillation loss in which a low-precision student prioritizes matching a high-precision teacher on high-SNR frequency bands; and (3) a weighted bit-cost term that encourages lower precision on low-SNR bands. The weights for the distillation and bit-cost terms are derived from the energy and entropy of each frequency band and decay with layer depth as spectral structure diminishes in deeper layers (Section V). At inference, FREQQUANT casts bit-width allocation as a Multiple-Choice Knapsack Problem (MCKP) and solves it greedily using the learned bit-width preferences (Section IV-C).

FREQQUANT outperforms state-of-the-art AQ techniques on VC and HAR tasks with higher accuracy and lower variance, achieving up to 65.47% (sampling-based) and 65.57% (data-free) additive accuracy improvements.

Main Contributions:

- We introduce FREQQUANT, a novel adaptive quantization technique for waveform modalities that addresses two key limitations of existing methods: high accuracy variance across quantization schemes and underutilization of spectral properties.
- We propose a spectral-saliency-guided training method with importance-weighted distillation and bit cost losses, and a greedy inference-time bit allocator (Section V).
- We show that FREQQUANT achieves state-of-the-art accuracy on ResNet-18 and ResNet-34 for time-series tasks (VC and HAR) while maintaining low bit-width under varying resource constraints.

II. RELATED WORK

Quantization techniques for Convolutional Neural Networks (CNNs) generally fall into two categories: Post-training quantization (PTQ) and Quantization-aware training (QAT). PTQ techniques quantize pre-trained full-precision models to lower precision, often including a calibration step for fine-tuning [16]. QAT techniques train quantized models from scratch. As FREQQUANT is an AQ technique performing QAT, we focus on Quantization Aware Training and Adaptive Quantization.

A. Quantization Aware Training

Numerous QAT techniques exist for low-precision CNNs [17]. DoReFa [18] stochastically quantizes parameter gradients to low precision, demonstrating improved performance over deterministic quantization for backpropagation. LQ-Nets [19] introduce the notion of a learnable quantizer instead of a fixed quantizer. PACT [20] introduces a learnable activation clipping parameter α that improves gradient flow and reduces quantization error. Unlike FREQQUANT, the mentioned techniques return a model with fixed bit-width allocation that cannot be changed post-training restricting its applicability to dynamically changing resource constraints.

B. Adaptive Quantization

Adaptive Quantization (AQ) enables models to adjust quantization schemes at runtime without retraining. In the vision domain, AdaBits [6] establishes a foundational AQ framework by combining Scale Adjusted Training with PACT [20]. To address feature distribution shifts across precisions, Any-Precision [15] utilizes switchable Batch Normalization (BN), while Bit-Mixer [7] employs transitional BN. Finally, RobustQuant [13] minimizes quantization error across different bit-widths by enforcing uniform weight distributions through Kurtosis regularization. Unlike FREQQUANT, the aforementioned AQ techniques require evaluating multiple bit-allocations at deployment, whereas FREQQUANT selects a configuration data-free without any post-training search.

III. MOTIVATION

We identify two challenges in applying AQ to waveform inference: accuracy variance and signal underutilization.

Accuracy Variance. Although AQ techniques allow inference across arbitrary quantization schemes, they exhibit high performance variability. Figure 1 illustrates the accuracy distributions for AdaBits, RobustQuant, and Bit-Mixer on the Moving Object Dataset (MOD) [21], [22] and PAMAP2 [23] using ResNet-18. We observe significant performance instability, with accuracy variances reaching up to $\sim 10\%$ within identical bit-width constraints. For example, AdaBits fluctuates between 44.94% and 68.25% on MOD in the 5.5–6.5 bitwidth range. This instability is particularly pronounced at lower precisions; for RobustQuant on PAMAP2, the accuracy spread in the low-bit regime ($27.77 \pm 15.03\%$) is markedly wider

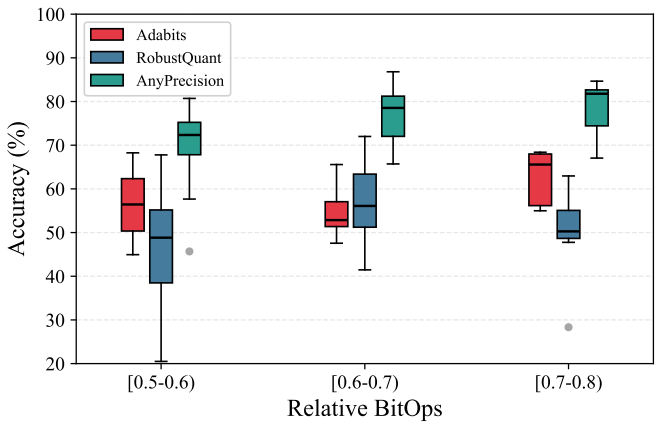

Fig. 1. We show the variation in accuracy for different Relative BitOps ranges (defined in Section VI) on the MOD dataset. The set of bit-widths used for bit-allocation is $2, 4, 8$. We observe that high variance in accuracy is observed across different adaptive quantization techniques under different resource budget.

than in higher bitwidths ($36.81 \pm 11.14\%$). Such high variance implies that randomly selecting a bit allocation under resource constraints often yields suboptimal results. FREQQUANT addresses this in two ways: its saliency-guided training yields lower variance across bit allocations when calibration data is available, and data-free bit allocation eliminates the need for random selection entirely when it is not (Section IV-C).

Underutilization of Signal Characteristics.

Vision-based AQ techniques typically assume spatial invariance regarding object's location, justifying the use of a single bit-width convolution. However, waveform modalities exhibit distinct spectral properties that challenge this assumption. Figure 2 displays the spectrogram of a moving object from the MOD dataset. We observe that a significant portion of the spectrum (200–800 Hz) is dominated by noise rather than signal. Furthermore, higher frequency bands exhibit more uniform distributions, permitting aggressive Higher frequency bands exhibit more uniform distributions, permitting aggressive quantization [13], while lower frequency ranges require higher bit-widths — a pattern observed across VC and HAR. Using a single bit-width therefore leads to suboptimal memory utilization and accuracy.

IV. OVERVIEW

We study frequency-band-wise mixed-precision quantization for waveform-based modalities. Each input sample $x \in \mathbb{R}^{F \times T}$ is represented as a spectrogram with F frequency bins and T time steps. We partition the frequency axis into G contiguous, non-overlapping frequency bands.

Let L denote the total number of convolutional layers and let $\mathcal{K} \subset \mathbb{Z}^+$ denote the set of candidate bit-widths (e.g., $\mathcal{K} = \{2, 4, 8\}$). For each layer $l \in \{1, \ldots, L\}$ and frequency band $g \in \{1, \ldots, G\}$, a bit-width $b_g^{(l)} \in \mathcal{K}$ is assigned. We denote by $\mathbf{b}^{(l)} = [b_1^{(l)}, \ldots, b_G^{(l)}]$ the bit-width configuration at layer l, and by $\mathbf{b} = \{\mathbf{b}^{(l)}\}_{l=1}^{L}$ the complete network-wide configuration.

A. Problem Formulation

The goal is to select a bit-width configuration $\mathbf{b}$ that maximizes performance under a resource constraint. Let $V(\mathbf{b})$ denote a value function estimating the predictive performance of the model under configuration $\mathbf{b}$, and let $C(\mathbf{b})$ denote the corresponding resource cost (e.g., latency, energy, or memory). Given a budget τ, the inference-time optimization problem is:

$$\mathbf{b}^* = \arg\max_{\mathbf{b}} V(\mathbf{b}) \quad \text{s.t.} \quad C(\mathbf{b}) \leq \tau. \tag{1}$$

There are two key challenges associated with this problem. First, it is an instance of the Multiple-Choice Knapsack Problem (MCKP) where each layer-band pair constitutes a class with exactly one bit-width selected. Second, the budget τ varies across deployments and is unknown at training time, requiring a single model to support diverse constraints. These challenges motivate learning constraint-agnostic sensitivity information across layers and frequency bands.

B. Learning Budget-Agnostic Bit-Width Importance

To address the challenges of inference-time bit-width selection, we learn structured, budget-agnostic information about the *relative preference* of different bit-widths across layers and frequency bands. For each layer-band pair, we maintain a set of unnormalized preference scores $\mu_g^{(l)}$ over the candidate bit-widths in $\mathcal{K}$. Training proceeds by minimizing the expected loss over both the data distribution and sampled bit-width configurations:

$$\min_{\theta, \boldsymbol{\mu}} \mathbb{E}_{(x,y)\sim\mathcal{D}} \mathbb{E}_{\mathbf{b}\sim p(\mathbf{b}|\boldsymbol{\mu})} \left[\mathcal{L}(f_\theta(x; \mathbf{b}), y)\right], \tag{2}$$

where $p(\mathbf{b} \mid \boldsymbol{\mu}) = \prod_{l,g} p(b_g^{(l)} \mid \mu_g^{(l)})$ and $\mathcal{L}$ combines the task loss, a distillation loss, and a bit-width-dependent cost term. This joint training procedure optimizes both the model parameters θ and the preference scores $\boldsymbol{\mu}$, enabling the model to capture layer- and band-specific sensitivity to quantization.

In Section V, we describe in detail how FREQQUANT constructs these distributions and the corresponding loss functions from spectral saliency, connecting the learned bit-width preferences to concrete training objectives.

C. Inference-Time Bit-Width Selection

After training, the learned preference scores μ are fixed. They define the categorical distributions $\beta = \text{softmax}(\mu)$, which are then used at inference time to select bit-widths.

These distributions can be used in two ways:

1) *Sampling-based allocation*: We can also sample bit allocations directly from β, which often produces better configurations than uniform or random selection. If sampling does not yield a configuration within the target budget range, we fall back to the nearest valid allocation, preferring the lower one.
2) *Data-free bit allocation*: Given a deployment-specific budget τ, we compute a bit allocation by solving the optimization problem in Eq. (1).

The selected bit-widths define the value function $V(\mathbf{b})$. To satisfy a resource constraint, we use a greedy approximation to the inference-time optimization problem. We first assign each layer-band pair with the $\arg\max_k \beta_{g,k}^{(l)}$, which yields the maximum value of $V(\mathbf{b})$ found by our method. If $C(\mathbf{b}) > \tau$, we enumerate all valid transitions from the current assignment (e.g., 8→4, 8→2) and sort them by $\beta \cdot 1/\text{cost-saving}(\text{current}, \text{target})$. We then apply the highest-ranked transition and repeat this process until $C(\mathbf{b}) \leq \tau$ or all layers reach the minimum bit-width.

V. Training Methodology of FreqQuant

In this section, we describe how FreqQuant learns frequency-band-wise bit-width importance in a budget-agnostic manner. Our goal is to jointly learn model parameters and bit-width importance across layers and frequency bands, enabling adaptation to diverse inference-time constraints.

As introduced in Section IV, inference-time bit-width selection is posed as a constrained optimization problem over a value function that depends on a bit-width configuration. Rather than committing to a specific resource budget during training, we seek to learn budget-agnostic bit-width importance that captures the relative sensitivity of each layer and frequency band to quantization.

At a high level, in Eq. (2), training is formulated as Joint Quantization [6], i.e., a joint optimization problem in which the model is exposed to a diverse set of mixed-precision configurations. By optimizing the expected task loss under randomly sampled bit-width assignments, the model learns *relative precision preference* of each bit-width to each layer-band pair rather than a single fixed configuration. This learned preference is used to guide inference-time bit allocation under different deployment constraints.

Further, we introduced a preference parameter $\boldsymbol{\mu}$ to represent relative bit-width importance across layers and frequency bands (Eq. 2). We make this notion concrete by specializing $\boldsymbol{\mu}$ into a categorical parameterization over candidate bit-widths: the unnormalized scores $\mu_g^{(l)}$ are converted into a probability distribution $\boldsymbol{\beta}_g^{(l)} = \text{Gumbel-softmax}(\mu_g^{(l)})$, which is used to sample bit-widths during training. Gumbel-Softmax [24] provides a differentiable approximation to categorical sampling. During training, we use Gumbel-Softmax in soft mode.

A. Parameterization of Bit-Width Importance

For each convolutional layer $l \in \{1, \ldots, L\}$ and frequency band $g \in \{1, \ldots, G\}$, we define the distribution

$$p(b_g^{(l)} = k \mid \boldsymbol{\mu}_g^{(l)}) = \beta_{g,k}^{(l)}, \quad k \in \mathcal{K}, \tag{3}$$

are the normalized probabilities derived from the learned preference scores $\mu_g^{(l)}$. These parameters capture the *relative importance of each bit-width* for band g at layer l:

- Higher probability mass on larger bit-widths indicates higher sensitivity to quantization. The respective layer-band corresponds to higher task-relevance (Section V-B).
- Concentration on smaller bit-widths indicates robustness to aggressive quantization.

Crucially, $\boldsymbol{\beta}$ does not prescribe a single bit-width; instead, it represents flexible preferences that can later be adapted to different resource constraints. Figure 3 illustrates this behavior where the lower band (containing structured signal) converges toward 8-bit precision, while the upper band (noise-dominated) settles on 4-bit.

B. Spectral Saliency Estimation

While the bit-width importance parameters $\boldsymbol{\beta}$ define *what* precision configurations the model can prefer, they do not encode any notion of which frequency bands are more relevant to the task. To guide the learning of $\boldsymbol{\beta}$ toward allocating higher precision to informative spectral regions, we introduce a data-driven prior that quantifies the task relevance of each frequency band. We refer to this prior as *spectral saliency*.

The design of spectral saliency is motivated by two observations in the context of waveform-based classification. First, frequency components with highly structured temporal patterns are more sensitive to quantization noise and therefore benefit from higher precision, whereas high-entropy components are more robust to aggressive quantization [13]. Second, low-energy frequency components tend to contribute minimally to task performance and can often be quantized more aggressively without degrading accuracy. For instance, as shown in Figure 2, bands in A region are more sensitive to quantization than bands in B region. Together, these observations suggest that frequency bands with high energy and low entropy should be prioritized during mixed-precision learning.

Based on these observations, we define a bin-level spectral saliency measure that combines energy and entropy. For each frequency bin $j \in \{1, \ldots, F\}$, we compute

$$i_j = E_j \times (1 - \mathcal{H}_j), \tag{4}$$

where $E_j \in [0, 1]$ denotes the normalized energy of bin j, and $\mathcal{H}_j \in [0, 1]$ denotes the normalized Shannon entropy of that bin across T time steps. Entropy is normalized by $\log T$ to ensure comparable scaling. By construction, $i_j \in [0, 1]$, with higher values indicating greater spectral saliency.

Since mixed-precision decisions are made at the frequency-band level, we aggregate bin-level saliency within each band. For frequency band $g \in \{1, \ldots, G\}$, the band-level saliency is defined as $I_g = \sum_{j \in g} i_j$.

To enable comparison across bands, we normalize band-level saliency to obtain a relative saliency score

$$\tilde{I}_g = \frac{I_g}{\sum_{k=1}^{G} I_k + \epsilon}, \tag{5}$$

where ϵ is a small constant for numerical stability. The resulting scores satisfy $\tilde{I}_g \in [0, 1]$ and $\sum_{g=1}^{G} \tilde{I}_g = 1$.

We emphasize that spectral saliency is not a learned quantity and does not directly determine bit-width assignments. Instead, it serves as an external importance prior that modulates the training signal used to learn the bit-width importance parameters $\boldsymbol{\beta}$. In the following subsections, we describe how spectral saliency is incorporated into the training objective through saliency-guided loss functions that bias learning toward preserving information in spectrally informative regions.

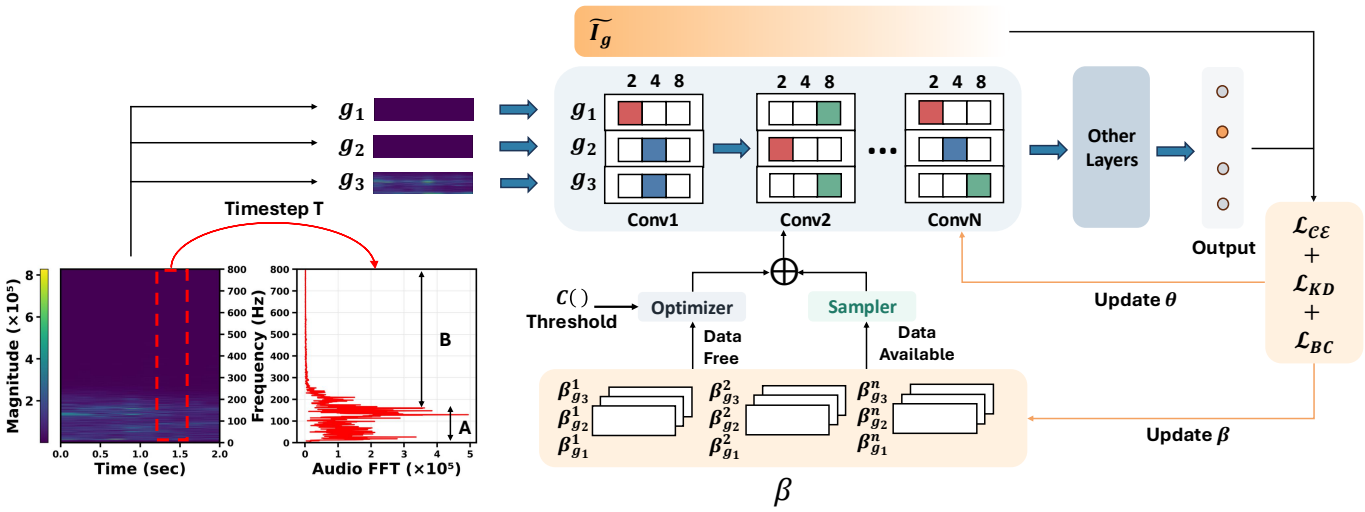

Fig. 2. Overview of the FREQQUANT framework. **Left:** Input spectrogram partitioned into G frequency bands with computed spectral saliency $\tilde{I}_g$. **Center:** Each convolutional layer maintains per-band bit-width preference distributions $\beta_g^{(l)}$ over candidate bit-widths $\{2, 4, 8\}$. **Right:** Training optimizes the combined loss ($\mathcal{L}_{\text{CE}} + \mathcal{L}_{\text{KD}} + \mathcal{L}_{\text{BC}}$) to update both model parameters θ and preference parameters β. The influence of $\tilde{I}_g$ on the loss reduces as depth increases. At inference, bit allocation proceeds either via sampling (data-available) or greedy optimization (data-free).

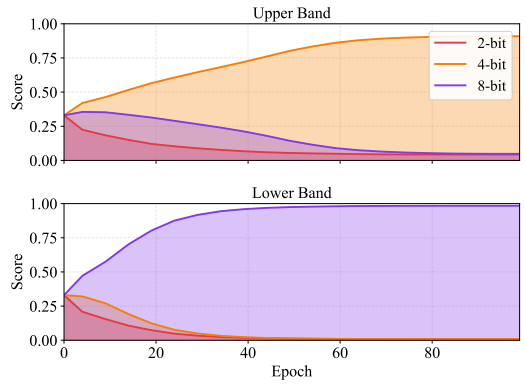

Fig. 3. Evolution of learned bit-width preferences β during training of ResNet-18 for the third convolution layer, we observe similar patterns for most layer. Here, $G = 2$. The upper band (high-frequency) converges toward lower precision (4-bit), while the lower band (low frequency), which contains more structured signal content, maintains higher preference for 8-bit precision.

C. Weighted Distillation Loss

To promote robustness under mixed-precision execution, we incorporate a distillation objective that aligns the representations produced by quantized executions with those of a high-precision reference. Unlike standard distillation, which treats all frequency bands equally, we design a saliency-guided distillation loss that emphasizes fidelity in spectrally informative regions while suppressing noise-dominated bands.

Specifically, for each layer l and frequency band g, we compute the discrepancy between the student (quantized) and teacher (full-precision) feature maps. The weighted distillation loss is defined as

$$\mathcal{L}_{\text{KD}} = \sum_{l=1}^{L} \sum_{g=1}^{G} \phi_g(l) \cdot \left\| \mathbf{T}^{(l,g)} - \mathbf{S}^{(l,g)} \right\|_2^2, \tag{6}$$

where $\mathbf{T}^{(l,g)}$ and $\mathbf{S}^{(l,g)}$ denote the teacher and student feature maps for band g at layer l, respectively. Since the teacher and student share the same network parameters and differ only in numerical precision, minimizing this discrepancy is equivalent to minimizing quantization-induced error.

The weighting function $\phi_g(l)$ controls the relative importance of matching the teacher in band g at layer l. We define

$$\phi_g(l) = \frac{G \cdot m_g \cdot \left[\alpha(l)\, \tilde{I}_g + (1 - \alpha(l))\, \frac{1}{G}\right]}{\sum_{j=1}^{G} m_j \cdot \left[\alpha(l)\, \tilde{I}_j + (1 - \alpha(l))\, \frac{1}{G}\right] + \epsilon}, \tag{7}$$

where $\tilde{I}_g$ is the relative spectral saliency defined in Section V-B. Here, $m_g = \frac{\tilde{I}_g}{\tilde{I}_g + \epsilon}$ acts as a saliency gate, and $\alpha(l) = \cos\left(\frac{\pi l}{2L}\right)$ is a depth-dependent decay factor.

Intuitively, this formulation interpolates between saliency-weighted distillation in early layers and uniform distillation in deeper layers. In shallow layers, where spectral structure is preserved, the loss emphasizes bands with higher saliency. In deeper layers, where representations become increasingly abstract, the weighting gradually becomes uniform.

The proposed weighting scheme is designed to satisfy the following properties:

- **Zero-saliency suppression**: If all bands have zero saliency, the distillation loss vanishes; i.e., if $\tilde{I}_g = 0$ for all g, then $m_g = 0$, yielding $\mathcal{L}_{\text{KD}} = 0$.
- **Band-specific suppression**: Bands with zero saliency do not contribute to the loss; i.e., if $\tilde{I}_{g_1} = 0$ while other bands have nonzero saliency, then $m_{g_1} \approx 0$, and consequently $\phi_{g_1}(l) \approx 0$.
- **Uniform fallback**: If all bands have equal saliency, the loss reduces to standard (unweighted) distillation; i.e., if $\tilde{I}_g = \frac{1}{G}$ for all g, then the weighting becomes $\phi_g(l) = \frac{1}{G}$.

- **Depth decay**: The effect of spectral saliency diminishes with increasing layer depth; i.e., as l increases, $\alpha(l) \to 0$, causing the weights to become uniform across bands.

By incorporating spectral saliency and depth-aware weighting, the distillation loss biases learning toward preserving information in critical frequency regions and early stages, without enforcing uniform precision across the spectrum.

D. Weighted Bit-Width Cost Loss

While the weighted distillation loss promotes representational fidelity under quantization, it does not minimize bit allocation. To bias learning toward compact mixed-precision configurations, we introduce a saliency-guided bit-width cost loss that penalizes expected precision usage while allowing higher precision in spectrally informative regions.

For each layer l and frequency band g, we define the expected bit-width as $B_g^{(l)} = \sum_{k \in \mathcal{K}} \beta_{g,k}^{(l)} \cdot k$, where $\beta_{g,k}^{(l)}$ are the bit-width importance parameters introduced in Section V-A. The weighted bit-width cost loss is then defined as

$$\mathcal{L}_{\text{BC}} = \sum_{l=1}^{L} \sum_{g=1}^{G} \left[(1 - c_g(l)) \cdot B_g^{(l)} + c_g(l) \cdot \left(B_{\max} - B_g^{(l)} \right) \right], \tag{8}$$

where $B_{\max} = \max_{k \in \mathcal{K}} k$ and $c_g(l) \in [0, 1]$ controls the trade-off between compression and capacity for band g at layer l.

When $c_g(l) = 0$, the loss encourages minimizing the expected bit-width, favoring aggressive compression. When $c_g(l) = 1$, the loss instead encourages maximizing precision for that band. Thus, $c_g(l)$ determines whether a band is pushed toward lower or higher precision during training.

We define the coefficient function $c_g(l)$ as

$$c_g(l) = \alpha(l) \cdot \gamma \cdot m_g \cdot \frac{\tilde{I}_g}{\max_j \tilde{I}_j + \epsilon}, \tag{9}$$

where $\tilde{I}_g$ is the relative spectral saliency, $m_g = \frac{\tilde{I}_g}{\tilde{I}_g + \epsilon}$ is the saliency gate, and $\alpha(l)$ is the depth decay (Section V-C). The scalar γ is a variance-based gate defined as

$$\sigma^2 = \frac{1}{G} \sum_{g=1}^{G} \left(\tilde{I}_g - \frac{1}{G} \right)^2, \qquad \gamma = \frac{\sigma^2}{\sigma^2 + \eta}, \tag{10}$$

where $\eta \ll 1$ is introduced to ensure numerical stability.

Unlike distillation, where equal saliency across bands implies equal importance, for bit allocation equal saliency indicates the absence of a discriminative signal. In this case, the model should default to compression rather than arbitrarily allocating higher precision. The variance gate γ detects whether saliency meaningfully distinguishes bands and disables saliency-driven allocation when it does not.

The coefficient function $c_g(l)$ satisfies the following properties:

- **Compression by default**: When saliency provides no discriminative signal, the loss encourages minimal precision; i.e., if all $I_g = 0$ or all I_g are equal, then $m_g = 0$ or $\gamma = 0$, yielding $c_g(l) = 0$ for all bands.

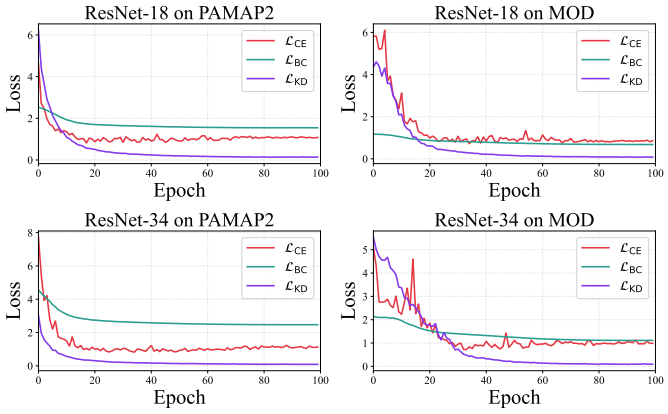

Fig. 4. Training loss curves (scaled) for FREQQUANT across architectures (ResNet-18, ResNet-34) and datasets (PAMAP2, MOD). All three loss components—cross-entropy ($\mathcal{L}_{\text{CE}}$), bit cost ($\mathcal{L}_{\text{BC}}$), and knowledge distillation ($\mathcal{L}_{\text{KD}}$)—converge smoothly, demonstrating stable joint optimization.

- **Saliency-proportional allocation**: Bands with higher relative saliency are pushed toward higher precision; i.e., for the most salient band in early layers when saliency varies across bands, $c_g(l) \approx 1$, pushing its expected bit-width toward $B_{\max}$.
- **Depth decay**: As layer depth increases, precision allocation becomes increasingly conservative regardless of saliency; i.e., as l increases, $\alpha(l) \to 0$, causing $c_g(l) \to 0$ for all bands.

Together with the weighted distillation loss, the proposed bit-width cost loss shapes the bit-width importance parameters $\boldsymbol{\beta}$ to reflect both task relevance and efficiency considerations, while remaining independent of any specific inference-time resource budget. Figure 4 shows the convergence of all three loss components during training.

VI. RESULTS

Existing adaptive quantization (AQ) techniques often struggle to achieve high accuracy on foreground activity classification tasks. As discussed in Section IV-C, deployment scenarios vary depending on the availability of deployment data and

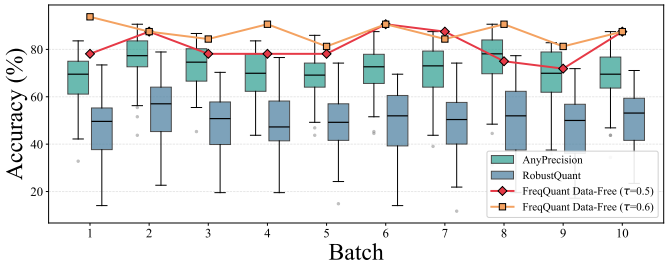

Fig. 5. Batch-wise accuracy comparison between baseline AQ methods (AnyPrecision, RobustQuant) using random bit allocation and FREQQUANT using data-free preference-based allocation (Section IV-C) at two resource thresholds ($\tau = 0.5$, $\tau = 0.6$). FREQQUANT achieves consistently higher accuracy without requiring deployment data for bit-width selection.

TABLE I
MEAN ACCURACY (%) AND VARIANCE ACROSS DIFFERENT RELATIVE BITOPS RANGES ON MOD AND PAMAP2 DATASETS USING RESNET-18/34. DF-FREQQUANT USES DATA-FREE BIT ALLOCATION

		MOD						PAMAP2					
		0.5–0.6		0.6–0.7		0.7–0.8		0.5–0.6		0.6–0.7		0.7–0.8	
		Variance	Mean	Variance	Mean	Variance	Mean	Variance	Mean	Variance	Mean	Variance	Mean
ResNet-18	Adabits	7.94	56.55	6.31	56.58	5.14	61.76	9.41	58.21	9.49	59.16	6.20	58.52
	BitMixer	4.86	68.02	10.38	65.62	2.94	72.69	3.92	67.08	7.42	67.72	5.02	71.74
	RobustQuant	14.16	43.58	8.35	56.22	9.03	50.64	15.03	27.77	14.59	38.82	11.14	36.81
	AnyPrecision	11.23	72.27	6.15	77.69	6.56	79.75	7.09	72.24	4.71	75.62	10.32	71.98
	FREQQUANT	5.11	82.51	0.99	84.34	0.99	84.34	**1.82**	**80.70**	1.32	79.94	1.85	80.59
	DF-FREQQUANT	**0.00**	**84.86**	**0.00**	**84.86**	**0.00**	**84.86**	-	70.47	**0.00**	**81.07**	**0.00**	**81.07**
ResNet-34	Adabits	8.36	64.58	6.26	68.21	5.03	65.51	11.60	50.52	10.47	48.99	14.55	40.60
	BitMixer	12.01	54.68	13.81	56.26	12.43	62.19	9.52	55.63	11.58	62.71	13.54	64.93
	RobustQuant	13.44	54.16	9.65	56.10	6.45	60.82	9.27	15.86	7.67	30.57	6.77	28.72
	AnyPrecision	4.17	77.94	5.61	76.22	3.36	82.64	13.09	62.88	4.72	70.29	6.46	70.80
	FREQQUANT	4.23	82.69	0.58	83.84	0.58	83.84	1.56	81.33	1.56	81.33	1.56	81.33
	DF-FREQQUANT	**0.00**	**83.99**	**0.00**	**83.99**	**0.00**	**83.99**	**0.00**	**81.43**	**0.00**	**81.43**	**0.00**	**81.43**

computational resources. We evaluate FREQQUANT under two realistic deployment scenarios:

1) **Data-Driven Deployment with Sampling (Scenario 1)** – The deployment environment allows the collection of some sample data and sufficient computation. Multiple quantization schemes can be evaluated on this data to identify the best scheme for deployment.
2) **Data-Free Deployment with Preference-Based Allocation (Scenario 2)** – The deployment environment has no access to data and limited computational resources. A quantization scheme must be selected without evaluating the model on deployment data.

For our evaluation, we use two datasets from distinct domains: MOD for VC [21], [22] and PAMAP2 [23] for HAR. MOD dataset contains audio and seismic data for classifying six vehicle types and one human walking activity. PAMAP2 dataset captures 18 human activities via accelerometers, gyroscopes, and magnetometers. We report results across Relative BitOps ranges $[0.5, 0.6)$, $[0.6, 0.7)$, and $[0.7, 0.8)$, where $\texttt{BitOps}(f) = b_w \cdot b_a \cdot |f| \cdot w_f \cdot h_f / s_f^2$ [25] where b_w, b_a the bit-widths of weights and activations, $|f|$ the filter cardinality, w_f, h_f, s_f the spatial width, height, and stride, normalized by the total BitOps of the INT8 model.

Models are trained on 8:1:1 splits (training:validation:test), for 100 epochs and we test the model with the highest validation accuracy. We use AdamW with an initial learning rate of 0.0001 and weight decay of 0.05, while β parameters start at 0.001. A cosine learning rate scheduler with decay rate 0.2 is applied. Expected loss for each batch is the mean over eight sampled bit allocations. For baselines, $p(b^{(l)}) = \frac{1}{|\mathcal{K}|}$, whereas for FREQQUANT, the bit allocation for band g and layer l is sampled from $p(b_g^{(l)} = k \mid \boldsymbol{\mu}_g^{(l)})$. We set $G = 2$ across all our experiments. The precision of the teacher model for $\mathcal{L}_{\text{KD}}$ is set to 16 bits.

A. Scenario 1: Data-Driven Deployment with Sampling

In this scenario, we assume the deployment environment allows us to collect some data and run multiple quantization schemes to identify the best one. Here, FREQQUANT samples bit-widths for all layer-band pairs from the learned distribution, $p(b_g^{(l)} = k \mid \boldsymbol{\mu}_g^{(l)}) = \beta_{g,k}^{(l)}$ where $k \in \mathcal{K}$.

To evaluate the effectiveness of each method, we sample multiple quantization schemes within the user-defined threshold and measure their Top-1 accuracy. We report both the mean and variance of Top-1 accuracy across these samples, which reflects the reliability of each technique in identifying a good deployment configuration. Lower variance implies fewer configurations are needed to reach high accuracy, reducing computational cost and improving deployment efficiency.

Table I summarizes the results for bit-width options $\mathcal{K} = \{2, 4, 8\}$. FREQQUANT consistently achieves higher mean Top-1 accuracy and the lowest variance compared to baseline AQ techniques such as AdaBits, Bit-Mixer, AnyPrecision, and RobustQuant in most cases. For example, for ResNet-18 on MOD at a $[0.5, 0.6)$ relative BitOps range, FREQQUANT achieves a mean accuracy of 82.51% with variance 5.11%, versus AnyPrecision's 72.27% mean accuracy and 11.23% variance. The lower variance demonstrates that FREQQUANT is more robust in selecting effective bit allocations, making the deployment process more stable and computationally efficient.

B. Scenario 2: Data-Free Deployment with Preference-Based Allocation

When no deployment data or computational resources are available, baseline techniques typically select a quantization scheme randomly. FREQQUANT, in contrast, performs a preference-based data-free allocation. Here, preference values act as a surrogate value function, and a greedy knapsack-style algorithm selects a quantization scheme that maximizes expected accuracy under a resource constraint. (Section IV-C).

Table I and Figure 5 show that FREQQUANT consistently outperforms random allocation, providing higher accuracy without relying on deployment data or heavy computation. The performance gap is particularly noticeable when resources are extremely constrained, demonstrating that preference-guided allocation is a viable alternative to random selection.

C. Ablation

Here, we evaluate key design choices in FREQQUANT's training procedure.

We observe that Gumbel-Softmax in hard mode with $\mathcal{L}_{\text{KD}}$ and $\mathcal{L}_{\text{BC}}$, performs significantly below the soft-mode baseline. This suggests that discrete sampling prematurely constrains the optimization landscape resulting in poor performance. When training in hard mode *without* $\mathcal{L}_{\text{KD}}$ and $\mathcal{L}_{\text{BC}}$, the model achieves $80.25 \pm 3.40\%$ accuracy. In this configuration, the learned β parameters converge toward 8-bit precision, reducing the adaptivity and maximizing accuracy for a single configuration. For knowledge distillation, a 16-bit teacher outperforms the 32-bit variant, consistent with [26].

We observe that exponential temperature schedules consistently outperform cosine decay. For depth decay $\alpha(l)$, cosine decay performs better than exponential decay. The exponential variant causes a rapid reduction in the contribution of $\tilde{I}_g$, which correlates with degraded performance. This indicates that a gradual attenuation of intermediate guidance is important for stable training and effective representation learning.

VII. CONCLUSION

Through extensive evaluation, we demonstrate the practical benefits of domain-specific compression. Specialized quantization for waveform modalities significantly improves low-bit inference accuracy under varying resource budgets. Our method, FREQQUANT, achieves higher accuracy on real-time sensing tasks using common CNNs such as ResNet-18 and ResNet-34. We introduce data-free bit allocation in the context of adaptive quantization. FREQQUANT provides up to 65.47% additive accuracy improvement with sampling-based allocation and 65.57% with data-free allocation.

VIII. ACKNOWLEDGMENT

Research reported in this paper was sponsored in part by DEVCOM ARL under Cooperative Agreement W911NF-172-0196, NSF CNS 20-38817, and the Boeing Company.

REFERENCES

[1] Z. Liu, M. Sun, T. Zhou, G. Huang, and T. Darrell, "Rethinking the value of network pruning," in *ICLR*, 2019.

[2] P. Molchanov, A. Mallya, S. Tyree, I. Frosio, and J. Kautz, "Importance estimation for neural network pruning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[3] G. E. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *ArXiv*, vol. abs/1503.02531, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:7200347

[4] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *Int. J. Comput. Vision*, vol. 129, no. 6, p. 1789–1819, jun 2021. [Online]. Available: https://doi.org/10.1007/s11263-021-01453-z

[5] Z. Liu, Y. Wang, K. Han, S. Ma, and W. Gao, "Instance-aware dynamic neural network quantization," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 12 434–12 443.

[6] Q. Jin, L. Yang, and Z. Liao, "Adabits: Neural network quantization with adaptive bit-widths," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2146–2156.

[7] A. Bulat and G. Tzimiropoulos, "Bit-mixer: Mixed-precision networks with runtime bit-width selection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5188–5197.

[8] J. Yang, X. Shen, J. Xing, X. Tian, H. Li, B. Deng, J. Huang, and X.-s. Hua, "Quantization networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[9] A. Gholami, S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, and K. Keutzer, "A survey of quantization methods for efficient neural network inference," *CoRR*, vol. abs/2103.13630, 2021. [Online]. Available: https://arxiv.org/abs/2103.13630

[10] S. Islam, H. Elmekki, A. Elsebai, J. Bentahar, N. Drawel, G. Rjoub, and W. Pedrycz, "A comprehensive survey on applications of transformers for deep learning tasks," *Expert Syst. Appl.*, vol. 241, no. 122666, p. 122666, May 2024.

[11] N. Dua, S. N. Singh, and V. B. Semwal, "Multi-input CNN-GRU based human activity recognition using wearable sensors," *Computing*, vol. 103, no. 7, pp. 1461–1478, Jul. 2021.

[12] N. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. Campbell, "A survey of mobile phone sensing," *IEEE Commun. Mag.*, vol. 48, no. 9, pp. 140–150, Sep. 2010.

[13] M. Shkolnik, B. Chmiel, R. Banner, G. Shomron, Y. Nahshan, A. M. Bronstein, and U. C. Weiser, "Robust quantization: One model to rule them all," *CoRR*, vol. abs/2002.07686, 2020. [Online]. Available: https://arxiv.org/abs/2002.07686

[14] K. Xu, Q. Feng, X. Zhang, and D. Wang, "Multiquant: Training once for multi-bit quantization of neural networks," in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, ser. IJCAI-2022. International Joint Conferences on Artificial Intelligence Organization, Jul. 2022. [Online]. Available: http://dx.doi.org/10.24963/ijcai.2022/504

[15] H. Yu, H. Li, H. Shi, T. S. Huang, and G. Hua, "Any-precision deep neural networks," *arXiv preprint arXiv:1911.07346*, 2019.

[16] B. Rokh, A. Azarpeyvand, and A. Khanteymoori, "A comprehensive survey on model quantization for deep neural networks in image classification," *ACM Trans. Intell. Syst. Technol.*, vol. 14, no. 6, Nov. 2023. [Online]. Available: https://doi.org/10.1145/3623402

[17] D. Zhang, J. Yang, D. Ye, and G. Hua, "Lq-nets: Learned quantization for highly accurate and compact deep neural networks," 2018. [Online]. Available: https://arxiv.org/abs/1807.10029

[18] S. Zhou, Z. Ni, X. Zhou, H. Wen, Y. Wu, and Y. Zou, "Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients," *CoRR*, vol. abs/1606.06160, 2016. [Online]. Available: http://arxiv.org/abs/1606.06160

[19] D. Zhang, J. Yang, D. Ye, and G. Hua, "Lq-nets: Learned quantization for highly accurate and compact deep neural networks," *CoRR*, vol. abs/1807.10029, 2018. [Online]. Available: http://arxiv.org/abs/1807.10029

[20] J. Choi, Z. Wang, S. Venkataramani, P. I. Chuang, V. Srinivasan, and K. Gopalakrishnan, "PACT: parameterized clipping activation for quantized neural networks," *CoRR*, vol. abs/1805.06085, 2018. [Online]. Available: http://arxiv.org/abs/1805.06085

[21] J. Li, Y. Chen, T. Kimura, T. Wang, R. Wang, D. Kara, Y. Hu, L. Wu, W. A. Hanafy, A. Souza, P. Shenoy, M. Wigness, J. Bhattacharyya, J. Kim, G. Wang, G. Kimberly, J. Eckhardt, D. Osipychev, and T. Abdelzaher, "Acies-os: A content-centric platform for edge ai twinning and orchestration," in *2024 33rd International Conference on Computer Communications and Networks (ICCCN)*. IEEE, 2024, pp. 1–1.

[22] S. Liu, T. Kimura, D. Liu, R. Wang, J. Li, S. Diggavi, M. Srivastava, and T. Abdelzaher, "Focal: Contrastive learning for multimodal time-series sensing signals in factorized orthogonal latent space," in *Advances in Neural Information Processing Systems*, 2023.

[23] A. Reiss, "PAMAP2 Physical Activity Monitoring," UCI Machine Learning Repository, 2012, DOI: https://doi.org/10.24432/C5NW2H.

[24] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," 2017. [Online]. Available: https://arxiv.org/abs/1611.01144

[25] X. Huang, Z. Shen, S. Li, Z. Liu, H. Xianghong, J. Wicaksana, E. Xing, and K.-T. Cheng, "Sdq: Stochastic differentiable quantization with mixed precision," in *International Conference on Machine Learning*. PMLR, 2022, pp. 9295–9309.

[26] S.-I. Mirzadeh, M. Farajtabar, A. Li, N. Levine, A. Matsukawa, and H. Ghasemzadeh, "Improved knowledge distillation via teacher assistant," 2019. [Online]. Available: https://arxiv.org/abs/1902.03393