Are Binary Neural Networks Input Bit-Invariant? Optimizing Sensor-Model Systems with Mean Hassanat Distance

Tushar Shinde, Avinash Kumar Sharma
MIDAS (Multimedia Intelligence, Data Analysis and compreSsion) Lab
Indian Institute of Technology Madras, Zanzibar, Tanzania
shinde@iitmz.ac.in

Abstract

Resource-constrained sensing systems must balance data quality and energy consumption, yet the interaction between low-bit sensor input and model architecture remains poorly understood. We present a systematic evaluation of how Binary Neural Networks (BNNs) and FP32 models respond to joint optimization of data volume (via coresets) and input bit-precision (1–8 bit quantization). Our experiments on CIFAR-10 reveal a surprising architectural difference: BNNs maintain near-constant accuracy across all input bit-depths, whereas FP32 models degrade predictably (84.7% at 8-bit to 56.0% at 1-bit). This bit-invariance property enables BNNs to operate with variable-quality sensors without retraining, critical for adaptive sensing scenarios. We identify 75% of training data at 4-bit precision as an optimal operating point, achieving 87.5% resource reduction. At this point, accuracy remains 82.60% for FP32 models and 66.01% for BNNs. Additionally, Mean Hassanat Distance (MHD) between original and compressed weights exhibits strong negative correlations for BNNs (-0.87 to -0.95 PLCC), revealing fundamental differences in how binary networks respond to compression. These findings provide practical guidelines for deploying neural networks in energy-constrained environments with dynamically varying sensor quality, advancing the co-design of sensors and models for efficient machine perception.

1 Introduction and Related Works

Mobile robots navigating disaster zones must process visual data under severe power constraints Chen et al. (2024); Queralta et al. (2020). Autonomous vehicles need to adapt their sensing precision based on weather conditions Yao et al. (2023); Mușat et al. (2021). IoT devices monitoring infrastructure must operate for years on limited batteries Bouguera et al. (2018). These scenarios share a fundamental challenge: how can we maintain acceptable task performance while dramatically reducing the energy cost of sensing?

Current approaches treat sensor design and neural network optimization as separate problems. Sensors are typically designed to capture high-quality data for human perception Mennel et al. (2020), while neural networks are optimized assuming fixed input quality Chen et al. (2024); Liao et al. (2021). This separation misses a critical opportunity-jointly optimizing how we sense and how we process can lead to surprising efficiencies. In this work, we systematically explore what happens when we simultaneously reduce both data volume (through coresets) and data precision (through bit quantization) across different neural architectures.

Our experiments reveal an unexpected finding that challenges conventional wisdom: Binary Neural Networks (BNNs) are essentially blind to input bit-depth. Whether processing 8-bit images or

Learning to Sense (PMLR Proceedings Track) at NeurIPS 2025

1-bit silhouettes, BNNs maintain remarkably stable accuracy (65-69%). Meanwhile, standard FP32 networks suffer catastrophic degradation, dropping from 84.7% to 56.0% accuracy. This discovery has profound implications, a single BNN model can handle sensors of varying quality without modification, enabling truly adaptive sensing systems.

Consider a practical scenario: a drone performing search-and-rescue operations. During daylight with full battery, it captures 8-bit images for maximum detail. As battery depletes or visibility decreases, it seamlessly switches to 4-bit or even 1-bit sensing, extending operational time while maintaining usable performance. With FP32 models, this would require multiple models or accept severe accuracy loss. With BNNs, the same model handles all conditions.

Related Work. The intersection of efficient sensing and deep learning has attracted significant attention. Diamond et al. Diamond et al. (2021) demonstrated that image signal processors designed for human vision discard information valuable for machine perception. LiKamWa et al. (2016) showed that analog processing before digitization can reduce energy by 85% for vision tasks.

Network quantization has progressed from 8-bit implementations Jacob et al. (2018) suitable for mobile deployment to extreme 1-bit networks Courbariaux et al. (2016); Rastegari et al. (2016). While these works demonstrate that binary weights can maintain reasonable accuracy, the interaction between weight quantization and input precision remains unexplored. Hubara et al. (2016) studied quantized networks but focused on weight/activation quantization, not input bit-depth effects.

Coreset selection Mirzasoleiman et al. (2020); Pooladzandi et al. (2022) reduces training data while preserving model performance, but prior work assumes high-quality inputs. The joint optimization of data quantity and quality represents an unexplored dimension.

Model similarity metrics Kornblith et al. (2019); Raghu et al. (2017) traditionally assume continuous weights. Our work reveals these metrics behave fundamentally differently for binary networks, showing strong negative correlations that indicate inverse relationships between weight changes and performance.

Contributions. This work makes the following contributions:

- We propose a systematic framework to evaluate joint optimization of input bit-depth ($k \in \{8, 6, 4, 2, 1\}$) and data volume via coreset selection on FP32 and BNN models, revealing fundamental architectural differences in response to sensor constraints.
- We discover bit-invariance in BNNs, showing they maintain 65-69% accuracy across all input precisions while FP32 models degrade predictably (84.7% to 56.0%), enabling single-model deployment for variable-quality sensors.
- We identify practical operating points for resource-constrained systems, demonstrating that 75% data with 4-bit precision achieves optimal efficiency (87.5% resource reduction) while maintaining usable accuracy.
- We present valuable results showing weight-distance metrics exhibit inverse correlations for BNNs (-0.87 to -0.95 PLCC), advancing understanding of binary network evaluation and highlighting the need for architecture-specific metrics.

2 Proposed Method

We evaluate the impact of joint data-precision optimization on neural network performance, systematically varying input bit-depth and training data volume to understand architectural differences in compression response.

2.1 Problem Description

We model the sensing stage as an *input quantizer* that reduces each RGB channel to k bits by zeroing the least significant $(8 - k)$ bits. Formally, given image tensor $x \in [0, 255]^{H \times W \times C}$, we apply: $x^{(k)} = \left\lfloor \frac{x}{2^{8-k}} \right\rfloor \cdot 2^{8-k}$, followed by normalization.

Let $(\mathcal{X}, \mathcal{Y})$ denote the input-label spaces and P a distribution on $\mathcal{X} \times \mathcal{Y}$. A reference network M_0 with parameters θ_0 implements a measurable map $f_{M_0} : \mathcal{X} \to \Delta^{K-1}$, producing a probability distribution over K classes. We evaluate models on a finite test set $S = \{(x_i, y_i)\}_{i=1}^{n}$ drawn i.i.d. from P. Classification accuracy is defined as:

$$\mathrm{Acc}(M; S) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}\big[\arg\max f_M(x_i) = y_i\big], \tag{1}$$

where $\mathbf{1}$ is the indicator function counting correct predictions. Equation (1) measures task performance, critical for assessing sensor-model systems under low-bit data constraints. This metric evaluates how well models retain classification accuracy when processing quantized sensor inputs, simulating real-world sensor limitations.

We consider two compression modalities applied to M_0: (i) *coreset selection* (data compression) producing M_c, and (ii) *input quantization* (low-bit data capture) producing M_q. Each modality $r \in \mathcal{R} := \{c, q\}$ is represented by an operator:

$$\mathcal{C}_r(\cdot; \lambda) : \mathcal{M} \to \mathcal{M}, \qquad M_{r,\lambda} := \mathcal{C}_r(M_0; \lambda), \tag{2}$$

parametrized by a compression budget $\lambda \in \Lambda_r$ (coreset fraction $\alpha \in [0.01, 0.05, 0.1, 0.25, 0.5, 0.75, 1.0]$ for $r = c$, or input bit-depth $k \in \{8, 6, 4, 2, 1\}$ for $r = q$). Equation (2) formalizes the transformation of the reference model into a compressed model, capturing data reduction or quantization effects. For coreset selection, $\mathcal{C}_c$ trains M_0 on a subset of training data, simulating constrained sensor data acquisition. For quantization, $\mathcal{C}_q$ trains M_0 on k-bit inputs, mimicking low-bit sensor capture.

To quantify compression impact, performance degradation is measured as:

$$\mathrm{ADROP}_{r,\lambda} = \mathrm{Acc}(M_0; S) - \mathrm{Acc}(M_{r,\lambda}; S) \in [0, 1]. \tag{3}$$

Equation (3) computes the accuracy drop between the reference model M_0 and compressed model $M_{r,\lambda}$, directly measuring the impact of compression on task performance. This metric quantifies how sensor constraints (e.g., low-bit data, limited samples) affect downstream tasks, guiding task-driven sensor design.

The primary objective is to investigate whether compression correlates with retained task performance. Let $\mathrm{MSIM}_{r,\lambda} := \mathrm{MSIM}(M_0, M_{r,\lambda})$ denote the similarity between original and compressed models. For each modality r, we analyze the family $\{(\mathrm{MSIM}_{r,\lambda}, \mathrm{ADROP}_{r,\lambda})\}_{\lambda \in \Lambda_r}$ using correlation measures: Pearson (PLCC), Spearman (SRCC), and Kendall's tau (KRCC). The pooled SRCC across all modalities is:

$$\rho_{\mathrm{all}} = \mathrm{Spearman}\Big(\{\mathrm{MSIM}_{r,\lambda}\}_{r \in \mathcal{R}, \lambda \in \Lambda_r}, \{\mathrm{ADROP}_{r,\lambda}\}_{r \in \mathcal{R}, \lambda \in \Lambda_r}\Big). \tag{4}$$

Equation (4) assesses whether higher similarity (lower MSIM) predicts lower accuracy drop, indicating preserved representational geometry. This objective of joint optimization, as strong correlations suggest that sensor compression (e.g., low-bit quantization) can maintain task performance, critical for energy-efficient systems.

2.2 Proposed Model Quality Metrics

To evaluate the quality of compressed neural networks, we propose the Mean Hassanat Distance (MHD) metric to measure alignment with the original model's representational geometry:

$$\mathrm{MHD}(x, y) = \frac{1}{n} \sum_{i=1}^{n} \frac{|x_i - y_i|}{\max(|x_i|, |y_i|) + c}, \quad c = 1.0, \tag{5}$$

where x and y are weight vectors of the original and compressed models, n is the number of weights, and $c = 1.0$ ensures numerical stability. Equation (5) normalizes weight differences by the maximum absolute value, making it robust to scale variations in FP32 models. For BNNs, where weights are ± 1, this normalization may reduce sensitivity to task-relevant changes. Unlike standard metrics (e.g., L1, Jensen-Shannon), MHD balances robustness and sensitivity, making it suitable for evaluating sensor-model systems under low-bit quantization.

MHD is computed by flattening model parameters into vectors and applying Equation (5), comparing M_0 to $M_{r,\lambda}$. We also evaluate standard metrics (L1, L2, Cosine, CKA, Hamming, Jensen-Shannon) to benchmark MHD's performance. High correlations between MHD and accuracy drop (Equation (3)) indicate effective geometry preservation, critical for task-driven sensor design. MHD's simplicity and robustness make it a promising metric for optimizing sensor-model systems in energy-constrained environments, such as mobile devices.

Our method quantifies the impact of low-bit quantization and coreset selection on model performance, with MHD offering a novel metric for sensor-model evaluation. The framework supports energy-efficient sensing in embedded systems, with results for BNNs guiding our metric design. By linking geometric perturbations to task performance, we advance the co-design of sensors and models for real-world applications.

3 Experimental Setup

This section details the experimental setup for evaluating the proposed metric, designed to quantify weight divergence in neural networks under low-bit sensor data and reduced training sets (coresets). The setup focuses on sensor optimization, quantization, and task-driven design, simulating real-world constraints in mobile and embedded systems where efficient data capture and processing are paramount.

Dataset. We conduct experiments on the CIFAR-10 dataset Krizhevsky et al. (2009), which includes 50,000 training and 10,000 test images of size 32×32 across 10 classes. CIFAR-10 is widely used in compression and quantization studies Han et al. (2015); Frankle and Carbin (1810), making it an effective benchmark for evaluating low-bit sensor data processing. Its compact size and moderate complexity facilitate efficient testing of quantization effects, though its low resolution may limit generalization to high-resolution sensor data. Training images are augmented with random horizontal flips and 4-pixel random cropping, and both training and test images are normalized using mean $[0.4914, 0.4822, 0.4465]$ and standard deviation $[0.2470, 0.2435, 0.2616]$. To simulate low-bit sensor capture, input images are quantized to $k \in \{1, 2, 4, 6, 8\}$ bits using the process described in previous section, aligning with focus on low-bit data capture.

Model Architecture. We use a ResNet-18 architecture He et al. (2016), balancing expressivity and computational efficiency, as demonstrated in prior compression studies Mirzasoleiman et al. (2020). For FP32 models, we modify the first convolutional layer to use a 3×3 kernel with stride 1 and padding 1, remove the max-pooling layer, and adjust the fully connected layer to output 10 classes. For Binary Neural Networks (BNNs), we use weights constrained to $\{-1, +1\}$ and apply a sign activation function, reducing computational cost for low-power sensor systems Courbariaux et al. (2016). ResNet-18's residual connections and moderate depth make it ideal for studying the interplay between low-bit sensor data and model weights, supporting joint sensor-model optimization tasks.

Training Implementation Details. Experiments are implemented in PyTorch on a Kaggle environment with an NVIDIA Tesla P100 GPU. FP32 models are trained using SGD with momentum 0.9, weight decay 5×10^{-4}, initial learning rate 0.01, batch size 128, and 10 epochs, following standard CIFAR-10 baselines He et al. (2016). BNN models use Adam with learning rate 0.01 and 20 epochs to accommodate binarization constraints, as binary weights require more iterations to converge Courbariaux et al. (2016). Early stopping with patience of 5 epochs prevents overfitting, enhancing robustness for sensor-model systems. Random seeds are fixed at 42 for reproducibility. These settings prioritize computational efficiency, with emphasis on energy-efficient sensing for mobile and autonomous platforms.

Coreset Setup. Training subsets (coresets) are selected with fractions $[0.75, 0.5, 0.25, 0.1, 0.05, 0.01]$ using random sampling from the CIFAR-10 training set. Random sampling ensures simplicity and reproducibility, though advanced coreset methods Mirzasoleiman et al. (2020) could enhance task-driven selection, a direction for future work. Models are trained from scratch using the above hyperparameters. Coresets simulate constrained data acquisition in sensor systems, such as mobile devices with limited storage.

Evaluation. For each input bit-depth of 8, 6, 4, 2, and 1, we evaluate test accuracy on the CIFAR-10 test set. The input images are first scaled to the 0–255 range, then quantized to retain only the most significant bits corresponding to the chosen bit-depth, effectively simulating low-bit sensor capture. After quantization, the images are re-normalized to the [0, 1] range before being fed to the model. This procedure reduces data precision while preserving task-relevant information. Test accuracy is

Table 1: Test accuracy of FP32 and BNN models under varying input bit-depth (k-bit) and training data fractions (coresets). Each entry shows the accuracy achieved when training with a specific k-bit input and coreset percentage, highlighting how FP32 models degrade with lower bit-depth and reduced data, while BNNs exhibit relative robustness across input precision levels.

Train k-bit	**FP32**							**BNN**						
	100%	**75%**	**50%**	**25%**	**10%**	**5%**	**1%**	**100%**	**75%**	**50%**	**25%**	**10%**	**5%**	**1%**
8-bit	0.8466	0.8304	0.7742	0.6817	0.5356	0.4564	0.2540	0.6765	0.6492	0.5842	0.4932	0.3685	0.3373	0.2389
6-bit	0.8557	0.8262	0.7754	0.6906	0.5454	0.4556	0.2410	0.6616	0.6382	0.5547	0.5130	0.3270	0.3396	0.2136
4-bit	0.8491	0.8260	0.7655	0.6883	0.5238	0.4412	0.2556	0.6774	0.6601	0.5579	0.4526	0.3038	0.3419	0.2372
2-bit	0.7766	0.7351	0.6979	0.5606	0.4260	0.3084	0.2190	0.6889	0.6469	0.5705	0.4683	0.3039	0.3400	0.2748
1-bit	0.5598	0.4996	0.4528	0.2904	0.2301	0.1825	0.1669	0.6568	0.6512	0.5723	0.5155	0.3323	0.3439	0.2335

reported to assess model performance under these quantized inputs, although extending these results to diverse or high-resolution datasets remains a future consideration.

Cross-Bit Testing Protocol. Models trained at bit-depth k_{train} are tested across all bit-depths $k_{test} \in \{1, 2, 4, 6, 8\}$, revealing robustness to variable sensor quality-critical for adaptive sensing scenarios.

Evaluation Metrics. We assess compressed models using classification accuracy, accuracy drop, and similarity metrics. Classification accuracy is the top-1 accuracy on the CIFAR-10 test set, measuring task performance. Accuracy drop is defined as: $\Delta\text{Acc} = \text{Acc}(M_0) - \text{Acc}(M)$, where $\text{Acc}(M_0)$ is the accuracy of the original model (trained on full data), and $\text{Acc}(M)$ is the accuracy of the compressed model (trained on a coreset). Equation (3) quantifies performance degradation due to data reduction, critical for evaluating sensor efficiency in constrained settings. Our proposed Mean Hassanat Distance (MHD) measures weight divergence between the original and compressed models. Equation (5) normalizes weight differences by the maximum absolute value plus a constant c, ensuring robustness to scale variations in FP32 models. For BNNs, where weights are ± 1, the normalization stabilizes comparisons but may fail to capture task-relevant changes, leading to negative correlations.

We also evaluate L1, L2, Cosine, CKA, Hamming, and JS distances Lin (2002) for comprehensive benchmarking. Correlation analysis uses Pearson (PLCC), Spearman (SRCC), and Kendall (KRCC) correlations between similarity metrics and accuracy drop across coreset settings, following best practices Kornblith et al. (2019); Neyshabur et al. (2018). MHD's simplicity and robustness make it a promising metric for FP32 sensor-model systems, though its limitations for BNNs guide future metric design. Visualizations are generated using `seaborn` and `matplotlib` to illustrate neural representation changes.

This setup evaluates low-bit quantization ($k = 1$ to 8) and coreset-based training, simulating real-world sensor constraints. MHD offers a novel, task-driven metric for sensor-model evaluation, with applications in energy-efficient mobile and autonomous systems. Negative correlation results for BNNs provide insights for refining metrics, advancing joint sensor-model optimization.

4 Results and Discussion

We present experimental results evaluating the impact of low-bit input quantization and reduced training data (coresets) on FP32 and Binary Neural Networks (BNNs) using CIFAR-10, with a focus on the proposed Mean Hassanat Distance (MHD) metric. Our experiments simulate constrained data acquisition in mobile and embedded systems. We analyze test accuracies, train-test bit-depth interactions, sensing efficiency, robustness, and correlations to assess model performance and MHD's predictive power, providing insights for energy-efficient sensing.

We evaluate FP32 and BNN models with input bit-depths $k \in \{8, 6, 4, 2, 1\}$ and coreset fractions $\{1.0, 0.75, 0.5, 0.25, 0.1, 0.05, 0.01\}$. Equation (1) measures task performance under low-bit sensor constraints, critical for assessing model effectiveness in task-driven sensor design. Equation (3) captures the impact of sensor constraints on performance, guiding the design of efficient sensors for resource-constrained systems. The MHD metric, measures weight divergence between M_0 and $M_{r,\lambda}$, with x, y as weight vectors, n as weight count, and c ensuring numerical stability. Equation (5) normalizes weight differences to assess representational geometry preservation, but its effectiveness is limited for BNNs due to binary weight constraints (± 1).

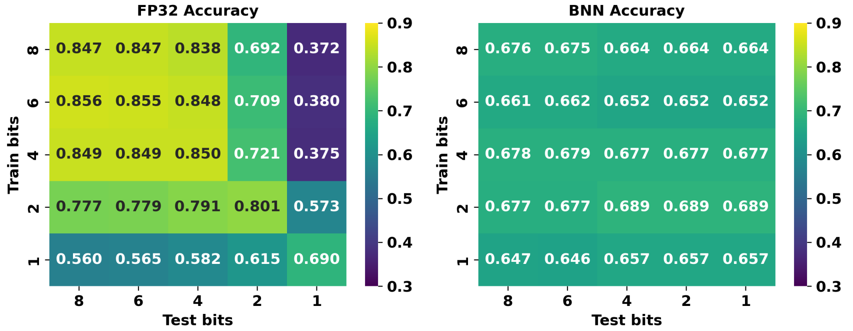

Figure 1: Test accuracy matrices for 100% training data (full coreset). Each cell indicates accuracy for a specific train-test bit-depth combination. The FP32 model (left) shows a strong diagonal pattern, reflecting sensitivity to input bit-depth, whereas the BNN model (right) exhibits nearly uniform values, demonstrating bit-invariance.

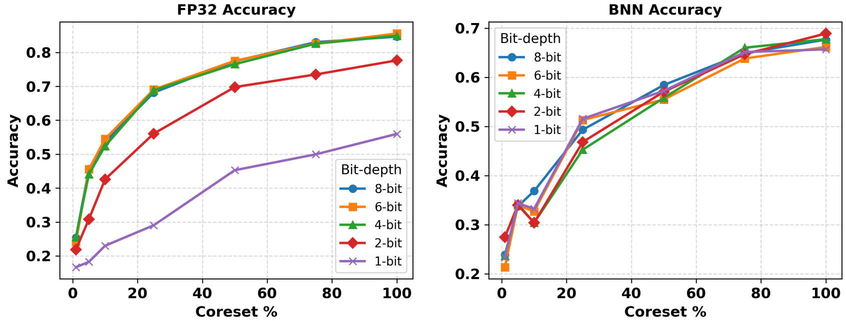

Figure 2: Test accuracy versus coreset percentage for FP32 (left) and BNN (right) models. Each line corresponds to a specific input bit-depth (8-bit to 1-bit). FP32 models display clear separation between bit-depths, indicating sensitivity to input precision, whereas BNN curves largely overlap, demonstrating bit-invariance across training data fractions.

FP32 and BNN Performance Analysis. We analyze train-test bit-depth interactions to assess model robustness to quantization mismatches, critical for real-world sensors with varying precision. Table 1 and Table 3 (Appendix A) reports accuracies for FP32 and BNN models across coreset fractions and train-test bit-depths. Figure 1 visualizes accuracies for 100% coreset as heatmaps, where each cell represents accuracy for a train-test bit-depth pair. We chose heatmaps to highlight patterns in accuracy degradation, with color intensity reflecting performance levels. For FP32 models, the diagonal concentration (e.g., 0.8466 at 8-bit train/test) indicates sensitivity to mismatches (e.g., 0.3716 at 8-bit train/1-bit test), emphasizing the need for aligned sensor-model pipelines. BNNs' uniform coloring (e.g., 0.6765 to 0.6636 for 8-bit training) demonstrates robustness to test-time quantization, making them suitable for low-bit sensor applications. This visualization illustrates how sensor precision impacts performance in real-world systems .The uniform BNN heatmap reveals the bit-invariance property that enables adaptive sensing .

Bit-Depth Impact on Data Efficiency. Table 1 shows test accuracies across bit-depths and coreset fractions. For FP32 models, accuracy decreases with lower bit-depths (e.g., 0.8466 at 8-bit to 0.5598 at 1-bit for 100% data) and smaller coresets (e.g., 0.8466 at 100% to 0.2540 at 1% for 8-bit). BNNs exhibit less sensitivity to bit-depth (e.g., 0.6765 at 8-bit vs. 0.6568 at 1-bit for 100% data-only 3%change versus FP32's 29% drop), but lower baseline accuracy (0.6765 vs. 0.8466 for FP32). Figure 2 plots accuracy versus coreset fraction, with lines for each bit-depth. FP32 models show clear separation by bit-depth (e.g., 8-bit outperforms 1-bit), while BNNs' overlapping curves indicate robustness to input bit-precision variations. This visualization underscores data efficiency challenges in low-bit sensor systems, with focus on constrained data acquisition for embedded platforms.

Sensing Efficiency. We introduce sensing efficiency, defined as Accuracy/Bit, to evaluate performance per bit of sensor data, critical for energy-constrained systems. Figure 3 plots sensing efficiency

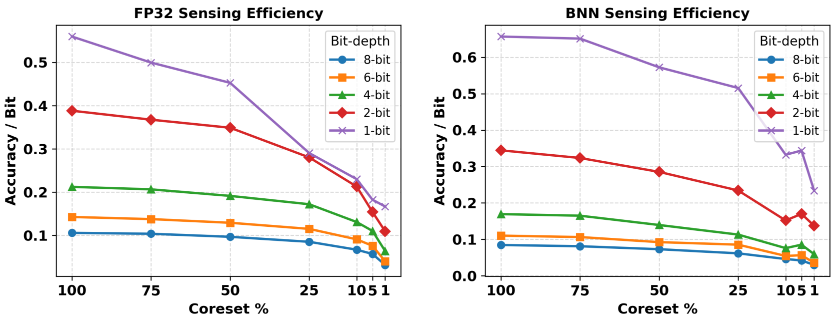

Figure 3: Sensing efficiency (accuracy per input bit) for FP32 (left) and BNN (right) models across varying coreset percentages. FP32 efficiency decreases at lower bit-depths, reflecting sensitivity to input precision, whereas BNNs maintain high efficiency across bit-depths, demonstrating bit-invariance.

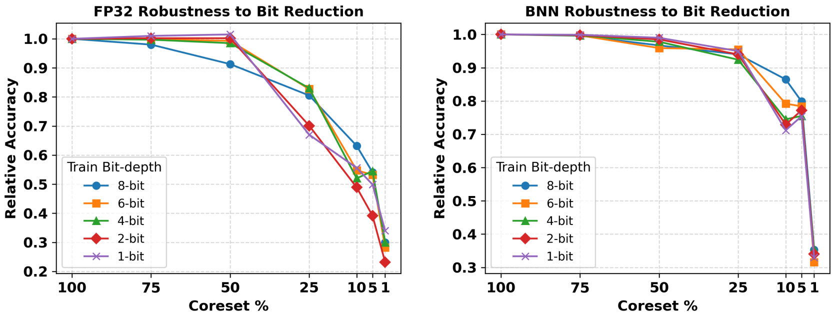

Figure 4: Robustness to input bit reduction across coreset percentages for FP32 (left) and BNN (right) models. Each line represents a training bit-depth (8-bit to 1-bit). FP32 models exhibit decreasing relative accuracy with smaller coresets, while BNNs maintain higher robustness across training data fractions.

versus bit-depth for various coreset fractions, to highlight trends across input bit-depth levels. We chose line plots to emphasize efficiency changes, with distinct line styles for coreset fractions to ensure readability. For FP32 models, efficiency decreases at lower bit-depths (e.g., from 0.106 at 8-bit to 0.560 at 1-bit for 100% data), reflecting sensitivity to input bit-depth variations.Importantly , BNNs maintain high efficiency at low bits (e.g., 0.656 at 1-bit), due to robustness, making them suitable for low-bit sensors. This metric and visualization guiding sensor design for mobile and autonomous platforms.

Robustness to Bit Reduction. We assess robustness to bit reduction by computing relative accuracy, defined as Acc(k)/Acc(8-bit), across coreset fractions. Figure 4 plots relative accuracy versus coreset percentage, for different train bit-depths. FP32 models show decreasing relative accuracy with smaller coresets (e.g., 0.66 at 1-bit for 100% data), indicating sensitivity to combined quantization and data reduction. BNNs maintain higher robustness (e.g., 0.97 at 1-bit), due to binary weight constraints, making them suitable for low-bit sensor systems. This visualization supports robust sensor-model co-optimization.

Correlation Analysis. Table 2 presents Pearson (PLCC), Spearman (SRCC), and Kendall (KRCC) correlations between distance metrics (L1, L2, Cosine, CKA, Hamming, JS, MHD) and accuracy drop (Equation (3)) across coreset settings. For FP32 models, MHD achieves high correlations (e.g., PLCC: 0.8970–0.9831, SRCC/KRCC: 1.0000), outperforming JS (0.7082–0.8475 PLCC), indicating strong predictive power. For BNNs, MHD shows negative correlations (e.g., PLCC: -0.8696 to -0.9473), unlike JS (0.6555–0.9621 PLCC), due to normalization in Equation (5) reducing sensitivity to binary weights (±1). Figure 5 plots absolute PLCC (|PLCC|) versus model layers for each metric, to highlight correlation trends across bit-depths. The negative correlations for BNNs underscore

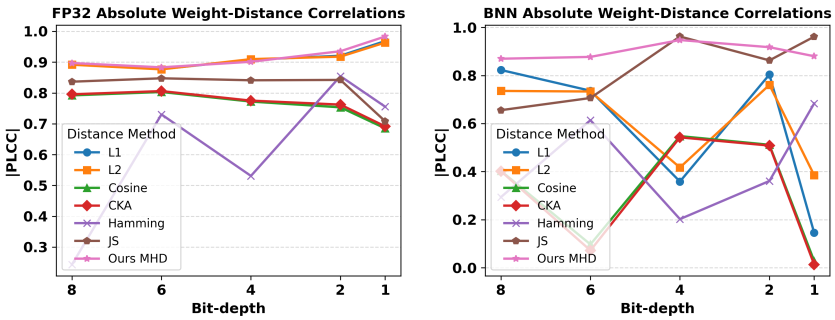

Figure 5: Absolute Pearson correlations (|PLCC|) between weight-distance metrics and model layers for varying input bit-depths. FP32 models (left) show strong positive correlations for the Mean Hassanat Distance (MHD), whereas BNNs (right) exhibit negative correlations, highlighting fundamental differences in how the two architectures respond to compression.

MHD's ability in capturing task-relevant changes in binary architectures, a key failure case guiding future metric design.

4.1 Discussion

BNN bit-invariance fundamentally changes adaptive sensing design.Our results reveal distinct behaviors of FP32 and BNN models under sensor constraints, with implications for task-driven sensor design. FP32 models' sensitivity to quantization and data reduction highlights the need for aligned sensor-model pipelines, while BNNs' robustness supports their use in low-bit systems despite lower baseline accuracy. MHD's high correlations for FP32 models (e.g., PLCC: 0.9831 at 1-bit) position it as a robust metric for predicting performance degradation, outperforming standard metrics like JS. However, its negative correlations for BNNs (e.g., PLCC: -0.9473) suggest that normalization in Equation (5) over-smooths binary weight differences, reducing task-relevant sensitivity. This failure case, underscores the need for BNN-specific metrics. Our framework advances sensor-model co-optimization, with applications in energy-efficient mobile and autonomous platforms.

Limitations and Impact. The use of CIFAR-10 limits generalization to high-resolution sensor data, a direction for future work alongside advanced coreset methods Mirzasoleiman et al. (2020). Limited coreset samples prevented statistical significance testing for correlations, addressable with larger datasets. Our framework and MHD metric provide a foundation for optimizing low-bit sensor systems, with negative BNN results guiding future metric design for energy-efficient embedded applications.

5 Conclusion

In this work, we investigate the interplay of low-bit input precision and reduced training data (coresets) in FP32 and Binary Neural Networks (BNNs) using CIFAR-10, with a focus on the proposed Mean Hassanat Distance (MHD) metric. Our findings demonstrate that FP32 models exhibit high sensitivity to quantization and data reduction, necessitating aligned sensor-model pipelines, while BNNs offer robustness to low-bit inputs, making them ideal for energy-efficient sensor systems despite lower baseline accuracy. The MHD metric achieves strong correlations with accuracy drop in FP32 models (e.g., PLCC: 0.9831 at 1-bit), outperforming standard metrics like JS divergence, but shows negative correlations for BNNs (e.g., PLCC: -0.9473), highlighting a critical limitation due to its normalization scheme. We introduce a novel sensing efficiency metric (Accuracy/Bit) to guide energy-constrained sensor design, supported by comprehensive analyses of accuracy, robustness, and train-test bit-depth interactions. These insights inform the development of efficient, robust sensor-model systems for mobile and autonomous platforms. The negative MHD results for BNNs provide a foundation for designing metrics tailored to binary architectures, to optimize sensing for resource-constrained environments.

Table 2: Correlation coefficients (PLCC, SRCC, KRCC) between various weight-distance metrics and model layers for FP32 (left) and BNN (right) models across input bit-depths. The Mean Hassanat Distance (MHD) exhibits the strongest positive correlations for FP32 and strong negative correlations for BNNs, highlighting fundamental differences in how the two architectures respond to compression and input quantization. Bold values indicate the best-performing metric per model.

Corr. Metric	**Distance Method**	**FP32**					**BNN**				
		8-bit	**6-bit**	**4-bit**	**2-bit**	**1-bit**	**8-bit**	**6-bit**	**4-bit**	**2-bit**	**1-bit**
PLCC	L1	0.8921	0.8772	0.9076	0.9207	0.9683	-0.8227	0.7364	-0.3586	-0.8037	0.1456
	L2	0.8916	0.8765	**0.9097**	0.9177	0.9634	-0.7356	0.7332	-0.4170	-0.7608	0.3851
	Cosine	0.7921	0.8031	0.7720	0.7533	0.6852	-0.4026	0.0971	-0.5476	-0.5115	0.0282
	CKA	0.7956	0.8060	0.7751	0.7622	0.6917	-0.4014	0.0718	-0.5424	-0.5082	0.0133
	Hamming	0.2435	-0.7303	0.5309	0.8544	0.7555	0.2939	0.6137	0.2023	0.3609	0.6835
	JS	0.8365	0.8475	0.8411	0.8423	0.7082	0.6555	0.7062	**0.9621**	0.8627	**0.9613**
	Ours MHD	**0.8970**	**0.8831**	*0.9011*	**0.9353**	**0.9831**	**-0.8696**	**-0.8772**	-0.9473	**-0.9176**	-0.8800
SRCC	L1	1.0000	1.0000	1.0000	1.0000	1.0000	-0.8286	0.6000	-0.4286	-0.4857	0.0857
	L2	1.0000	1.0000	1.0000	1.0000	1.0000	-0.6000	0.5429	-0.3143	-0.6000	0.2571
	Cosine	1.0000	1.0000	1.0000	1.0000	1.0000	-0.3714	-0.0286	-0.4286	-0.3714	0.0857
	CKA	1.0000	1.0000	1.0000	1.0000	1.0000	-0.3714	-0.0286	-0.4286	-0.3714	0.0857
	Hamming	0.0857	-0.7714	0.2000	0.7714	0.6000	-0.0286	0.6000	0.2571	0.2571	0.6000
	JS	0.8117	0.8286	0.7714	0.7714	0.8286	0.4857	0.4857	0.9429	0.7714	0.9429
	Ours MHD	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**-1.0000**	**-0.9429**	**-0.9429**	**-0.9429**	**-0.9429**
KRCC	L1	1.0000	1.0000	1.0000	1.0000	1.0000	-0.7333	0.4667	-0.3333	-0.3333	0.0667
	L2	1.0000	1.0000	1.0000	1.0000	1.0000	-0.4667	0.4667	-0.2000	-0.4667	0.2000
	Cosine	1.0000	1.0000	1.0000	1.0000	1.0000	-0.3333	-0.0667	-0.4667	-0.3333	0.0667
	CKA	1.0000	1.0000	1.0000	1.0000	1.0000	-0.3333	-0.0667	-0.4667	-0.3333	0.0667
	Hamming	0.0667	-0.6000	0.0667	0.6000	0.4667	-0.0667	0.4667	0.2000	0.2000	0.4667
	JS	0.6901	0.7333	0.6000	0.6000	0.7333	0.3333	0.3333	0.8667	0.6000	0.8667
	Ours MHD	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**1.0000**	**-1.0000**	**-0.8667**	**-0.8667**	**-0.8667**	**-0.8667**

Future Work. Future research will explore high-resolution datasets beyond CIFAR-10 to enhance generalization, incorporate advanced coreset selection methods, and develop BNN-specific distance metrics to address MHD's limitations. Statistical significance testing with larger coreset samples will further validate correlation analyses. These directions will strengthen sensor-model co-optimization, enabling scalable, energy-efficient sensing for real-world applications.

References

Taoufik Bouguera, Jean-François Diouris, Jean-Jacques Chaillout, Randa Jaouadi, and Guillaume Andrieux. Energy consumption model for sensor nodes based on lora and lorawan. *Sensors*, 18(7):2104, 2018.

Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. End-to-end autonomous driving: Challenges and frontiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Matthieu Courbariaux, Itay Hubara, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks: Training deep neural networks with weights and activations constrained to+ 1 or-1. *arXiv preprint arXiv:1602.02830*, 2016.

Steven Diamond, Vincent Sitzmann, Frank Julca-Aguilar, Stephen Boyd, Gordon Wetzstein, and Felix Heide. Dirty pixels: Towards end-to-end image processing and perception. *ACM Transactions on Graphics (TOG)*, 40(3):1–15, 2021.

Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. arxiv 2018. *arXiv preprint arXiv:1803.03635*, 1810.

Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks. *Advances in neural information processing systems*, 29, 2016.

Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2704–2713, 2018.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International conference on machine learning*, pages 3519–3529. PMlR, 2019.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images.(2009), 2009.

Fuyou Liao, Feichi Zhou, and Yang Chai. Neuromorphic vision sensors: Principle, progress and perspectives. *Journal of Semiconductors*, 42(1):013105, 2021.

Robert LiKamWa, Yunhui Hou, Julian Gao, Mia Polansky, and Lin Zhong. Redeye: analog convnet image sensor architecture for continuous mobile vision. *ACM SIGARCH Computer Architecture News*, 44(3):255–266, 2016.

Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37 (1):145–151, 2002.

Lukas Mennel, Joanna Symonowicz, Stefan Wachter, Dmitry K Polyushkin, Aday J Molina-Mendoza, and Thomas Mueller. Ultrafast machine vision with 2d material neural network image sensors. *Nature*, 579(7797): 62–66, 2020.

Baharan Mirzasoleiman, Jeff Bilmes, and Jure Leskovec. Coresets for data-efficient training of machine learning models. In *International Conference on Machine Learning*, pages 6950–6960. PMLR, 2020.

Valentina Mușat, Ivan Fursa, Paul Newman, Fabio Cuzzolin, and Andrew Bradley. Multi-weather city: Adverse weather stacking for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2906–2915, 2021.

Behnam Neyshabur, Zhiyuan Li, Srinadh Bhojanapalli, Yann LeCun, and Nathan Srebro. Towards understanding the role of over-parametrization in generalization of neural networks. *arXiv preprint arXiv:1805.12076*, 2018.

Omead Pooladzandi, David Davini, and Baharan Mirzasoleiman. Adaptive second order coresets for data-efficient machine learning. In *International Conference on Machine Learning*, pages 17848–17869. PMLR, 2022.

Jorge Pena Queralta, Jussi Taipalmaa, Bilge Can Pullinen, Victor Kathan Sarker, Tuan Nguyen Gia, Hannu Tenhunen, Moncef Gabbouj, Jenni Raitoharju, and Tomi Westerlund. Collaborative multi-robot search and rescue: Planning, coordination, perception, and active vision. *Ieee Access*, 8:191617–191643, 2020.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. *Advances in neural information processing systems*, 30, 2017.

Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *European conference on computer vision*, pages 525–542. Springer, 2016.

Shanliang Yao, Runwei Guan, Xiaoyu Huang, Zhuoxiao Li, Xiangyu Sha, Yong Yue, Eng Gee Lim, Hyungjoon Seo, Ka Lok Man, Xiaohui Zhu, et al. Radar-camera fusion for object detection and semantic segmentation in autonomous driving: A comprehensive review. *IEEE Transactions on Intelligent Vehicles*, 9(1):2094–2128, 2023.

A Train-Test Analysis

Table 3 provides detailed accuracies for FP32 and BNN models across coreset fractions and varying train-test bit-depths. The results highlight FP32's sensitivity to quantization mismatches (e.g., 8-bit train/1-bit test yields 0.3716 at 100% data) versus BNNs' robustness (e.g., 0.6636 at 8-bit train/1-bit test), supporting the design of robust sensor-model systems for energy-efficient applications.

Table 3: Test accuracy of FP32 and BNN models across varying coreset percentages and input bit-precision. Each row corresponds to a specific training bit-depth (k-bit), and columns show accuracy for different testing bit-depths. BNNs exhibit consistent performance across test bit-depths, demonstrating bit-invariance, whereas FP32 models show accuracy degradation as input precision decreases.

Coreset %	**Train k-bit**	**FP32**					**BNN**				
		8-bit	**6-bit**	**4-bit**	**2-bit**	**1-bit**	**8-bit**	**6-bit**	**4-bit**	**2-bit**	**1-bit**
100%	8-bit	0.8466	0.8472	0.8383	0.6916	0.3716	0.6765	0.6755	0.6636	0.6636	0.6636
	6-bit	0.8557	0.8554	0.8476	0.7086	0.3803	0.6615	0.6616	0.6517	0.6517	0.6517
	4-bit	0.8491	0.8492	0.8501	0.7205	0.3751	0.6785	0.6794	0.6774	0.6774	0.6774
	2-bit	0.7766	0.7792	0.7911	0.8009	0.5732	0.6772	0.6772	0.6889	0.6889	0.6889
	1-bit	0.5598	0.5653	0.5815	0.6146	0.6898	0.6469	0.6463	0.6568	0.6568	0.6568
75%	8-bit	0.8304	0.8300	0.8212	0.6702	0.3217	0.6492	0.6473	0.6386	0.6386	0.6386
	6-bit	0.8262	0.8256	0.8192	0.6903	0.3951	0.6406	0.6382	0.6283	0.6283	0.6283
	4-bit	0.8260	0.8257	0.8215	0.7039	0.4150	0.6569	0.6587	0.6601	0.6601	0.6601
	2-bit	0.7351	0.7377	0.7495	0.7711	0.5538	0.6493	0.6478	0.6469	0.6469	0.6469
	1-bit	0.4996	0.5032	0.5221	0.5997	0.6583	0.6473	0.6473	0.6512	0.6512	0.6512
50%	8-bit	0.7742	0.7741	0.7651	0.6308	0.3571	0.5842	0.5818	0.5695	0.5695	0.5695
	6-bit	0.7754	0.7755	0.7682	0.6447	0.3295	0.5569	0.5547	0.5385	0.5385	0.5385
	4-bit	0.7655	0.7664	0.7657	0.6542	0.3758	0.5506	0.5495	0.5579	0.5579	0.5579
	2-bit	0.6979	0.7007	0.7150	0.7338	0.5314	0.5728	0.5736	0.5705	0.5705	0.5705
	1-bit	0.4528	0.4564	0.4616	0.4812	0.5873	0.5724	0.5720	0.5723	0.5723	0.5723
25%	8-bit	0.6817	0.6819	0.6815	0.5484	0.3096	0.4932	0.4938	0.4933	0.4933	0.4933
	6-bit	0.6906	0.6921	0.6896	0.5631	0.3232	0.5110	0.5130	0.5003	0.5003	0.5003
	4-bit	0.6883	0.6895	0.6876	0.5990	0.3648	0.4519	0.4523	0.4526	0.4526	0.4526
	2-bit	0.5606	0.5618	0.5781	0.6228	0.4121	0.4574	0.4597	0.4683	0.4683	0.4683
	1-bit	0.2904	0.2945	0.3154	0.4071	0.5400	0.4972	0.4965	0.5155	0.5155	0.5155
10%	8-bit	0.5356	0.5351	0.5331	0.4457	0.2712	0.3685	0.3687	0.3610	0.3610	0.3610
	6-bit	0.5454	0.5464	0.5454	0.4563	0.2972	0.3269	0.3270	0.3316	0.3316	0.3316
	4-bit	0.5238	0.5247	0.5299	0.4500	0.3040	0.2956	0.2983	0.3038	0.3038	0.3038
	2-bit	0.4260	0.4312	0.4515	0.5014	0.3553	0.2951	0.2970	0.3039	0.3039	0.3039
	1-bit	0.2301	0.2352	0.2511	0.3342	0.4317	0.3332	0.3329	0.3323	0.3323	0.3323
5%	8-bit	0.4564	0.4578	0.4592	0.3876	0.2482	0.3373	0.3377	0.3399	0.3399	0.3399
	6-bit	0.4556	0.4559	0.4549	0.3839	0.2304	0.3411	0.3396	0.3248	0.3248	0.3248
	4-bit	0.4412	0.4400	0.4393	0.4031	0.2712	0.3408	0.3391	0.3419	0.3419	0.3419
	2-bit	0.3084	0.3130	0.3351	0.3795	0.2509	0.3294	0.3339	0.3400	0.3400	0.3400
	1-bit	0.1825	0.1826	0.1866	0.2194	0.2876	0.3328	0.3311	0.3439	0.3439	0.3439
1%	8-bit	0.2540	0.2558	0.2584	0.2499	0.1219	0.2389	0.2378	0.2322	0.2322	0.2322
	6-bit	0.2410	0.2402	0.2426	0.2350	0.1286	0.2120	0.2136	0.1982	0.1982	0.1982
	4-bit	0.2556	0.2553	0.2565	0.2606	0.1644	0.2160	0.2218	0.2372	0.2372	0.2372
	2-bit	0.2190	0.2189	0.2304	0.2655	0.1966	0.2591	0.2607	0.2748	0.2748	0.2748
	1-bit	0.1669	0.1685	0.1764	0.2041	0.2326	0.2161	0.2205	0.2335	0.2335	0.2335