International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 04 | April - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

8 Bit Quantized Neural Network

Shyadha Fathima J[1], Tarakeshwari S N[1], Thansika K[1] , P Amudha[2]

[1] Final year BE students, [2]Head of the Department & Professor, Department of CSE, School of Engineering, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore

22ueo052@avinuty.ac.in, 22ueo059@avinuty.ac.in, 22ueo060@avinuty.ac.in, amudha_cse@avinuty.ac.in

Abstract - Efficient deployment of deep neural networks on edge devices is constrained by limited computational resources, memory, and energy availability. While full-precision (FP32) models provide high accuracy, their resource demands hinder practical edge implementation. This paper evaluates 8-bit integer (INT8) quantization using Quantization-Aware Training (QAT) on the ResNet-18 architecture with the CIFAR-10 dataset. By modeling quantization effects during training, the approach reduces the accuracy loss typically associated with lower precision. Experimental results show that the INT8 model achieves 94.77% accuracy, closely matching the FP32 baseline of 94.79%, while significantly improving efficiency in terms of model size and computation. These findings demonstrate that INT8 quantization offers an effective trade-off between performance and resource utilization for real-time edge AI applications.

***KeyWords*:** Edge Computing, Model Quantization, Quantization-Aware Training, ResNet-18, Low-Precision Neural Networks, Embedded AI.

1. INTRODUCTION

The rapid expansion of edge computing and Internet of Things (IoT) technologies has significantly influenced how artificial intelligence models are deployed. Performing inference closer to the data source helps reduce latency, conserve bandwidth, and enhance data privacy. Despite these advantages, implementing deep neural networks (DNNs) on edge devices is challenging due to constraints in processing power, memory capacity, and energy resources. Most modern architectures rely on 32-bit floating-point (FP32) representation, which leads to substantial model size and computational cost, limiting their suitability for resource-limited platforms.

Conventional model development primarily prioritizes accuracy, often overlooking hardware efficiency. This results in increased inference time and power consumption when deployed outside high-performance environments. To overcome these limitations, techniques such as pruning, knowledge distillation, and quantization have been introduced. Among these, quantization is particularly effective as it lowers precision to reduce both memory footprint and computational complexity.

In this work, 8-bit integer (INT8) quantization is explored due to its ability to retain near-original accuracy while significantly improving efficiency. Quantization-Aware Training (QAT) is employed to integrate quantization effects during training, allowing the model to adapt to reduced precision and maintain performance.
This study utilizes the ResNet-18 architecture on the CIFAR-10 dataset to compare FP32 and INT8 models across key metrics such as accuracy and computational efficiency. The objective is to demonstrate that INT8 quantization, when combined with QAT, provides an effective balance between performance and resource utilization, making it suitable for real-time edge deployment.

The organization of the rest of this paper is as follows: Section 2 explores the previous work taken in this area and Section 3 describes the proposed system. Section 4 discusses the results of the proposed system and Section 5 provides conclusion for this paper and suggests avenues for future work

2. RELATED WORK

The optimization of deep neural networks for edge deployment has become an important area of research due to the increasing demand for efficient, low-power artificial intelligence systems. As modern applications require real-time inference on resource-constrained devices, reducing model complexity while maintaining accuracy has emerged as a major challenge.

Zhang et al. [1] proposed a hardware accelerator using mixed-precision quantization with dynamic bit-width allocation across layers. Their approach achieves a 3.2× speedup and 45% power reduction with minimal accuracy loss, demonstrating the effectiveness of precision-aware dataflow.

Shen et al. [2] introduced a post-training quantization framework that combines layer-wise calibration with an error feedback mechanism. Their method achieves a 2.6× reduction in model size while maintaining near full-precision accuracy, making it suitable for latency-sensitive applications.

Gholami et al. [3] provided a comprehensive review of quantization techniques, reporting that 8-bit quantization achieves 3–4× compression with minimal accuracy loss, while 4-bit quantization offers 6–8× compression with moderate degradation. Their study emphasizes hardware-aware and mixed-precision trends in modern deployment.

Li et al. [4] proposed BRECQ, a post-training quantization framework that reconstructs neural networks block-by-block to minimize quantization error without full retraining. The approach enables ultra-low bit-width quantization (down to INT2) while maintaining accuracy comparable to quantization-aware training and significantly reduced model deployment time and computational cost. Jin et al. [5] suggested AdaBits, an adaptive quantization framework that dynamically adjusts bit-widths for weights and activations.
Through joint training and the Switchable Clipping Level (S-CL) technique, their method maintains strong accuracy even at very low precisions, while enabling a single model to flexibly adapt to diverse hardware constraints. This highlights the

effectiveness of adaptive bit-width quantization in achieving efficient deployment without retraining

Nagel et al. [6] introduced a data-free quantization approach that applies weight equalization and bias correction to minimize quantization errors without requiring retraining or calibration data. The method achieves near full-precision accuracy with 8-bit inference while significantly reduced deployment complexity and demonstrated the effectiveness of simple post-processing techniques for efficient model quantization.

Banner et al. [7] recommended an 8-bit training framework for deep neural networks that preserves accuracy through quantization-aware techniques. The approach reduces memory and computation costs while achieving performance comparable to full-precision models and demonstrated the practicality of low-precision training.

Jacob et al. [8]projected a quantization framework that enables neural network inference using integer-only arithmetic by jointly quantizing weights and activations to 8-bit precision. The approach incorporates a co-designed training scheme to preserve accuracy and achieving up to 4× memory reduction, improved latency–accuracy trade-off on mobile hardware. This demonstrated efficient deployment of deep models on resource-constrained devices.

Krishnamoorthi [9] proposed a comprehensive post-training quantization framework for deep convolutional networks that utilizes per-channel weight quantization and per-layer activation quantization to achieve efficient 8-bit inference. The approach reduces model size by up to 4× and achieves 2–3× speedup with minimal accuracy loss (within ~2% of floating-point models) exhibited the effectiveness of practical quantization techniques for real-world deployment.

He et al. [10] offered a deep residual learning framework that introduces skip connections to ease the training of very deep neural networks. The approach enables networks up to 152 layers to be trained effectively, achieving a 3.57% top-5 error on ImageNet and significantly improving accuracy with increased depth, demonstrating the effectiveness of residual mapping for optimizing deep architectures.

3. SYSTEM ARCHITECTURE

The proposed system is structured to support efficient training and evaluation of reduced-precision neural networks for edge deployment. It consists of four key stages: dataset preparation, preprocessing, baseline FP32 model training, and INT8 quantization using Quantization-Aware Training (QAT).This modular pipeline enables systematic comparison of models based on accuracy, computational cost, and memory efficiency, facilitating analysis of performance trade-offs in resource-constrained environments.

Fig 1 illustrates the proposed system and helps understand the pipeline process.

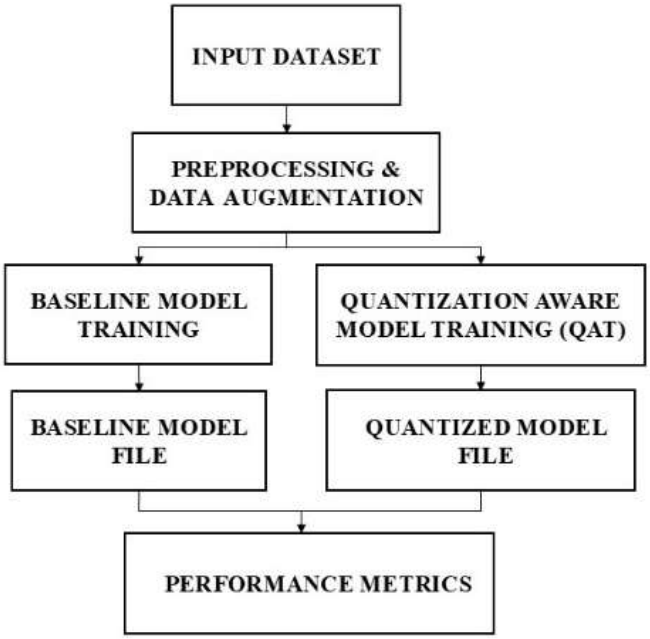

Fig - 1: Block Diagram

A. Data Collection

The data collection component focuses on acquiring a standardized dataset for training and evaluating deep neural network models under different precision settings. In this work, the CIFAR-10 dataset [11] is selected due to its widespread use in image classification benchmarking and its suitability for lightweight architectures. The dataset contains 60,000 labeled RGB images categorized into 10 distinct classes, including vehicles, animals, and everyday objects, providing sufficient diversity for model evaluation.

The dataset is accessed using the TorchVision library, which enables efficient downloading and loading into the training pipeline. It is divided into 50,000 training images and 10,000 testing images, ensuring a clear separation between model learning and evaluation phases. Each image has a resolution of 32×32 pixels, making it computationally efficient for experimentation, particularly in edge-oriented scenarios. Fig 2 shows sample images from the CIFAR-10 dataset, illustrating the diversity and complexity of the classes used for training and evaluation.

Using CIFAR-10 ensures consistency and reproducibility across experiments. Its standardized format allows fair comparison between FP32 and INT8 models, eliminating variability that could arise from custom datasets while enabling reliable performance assessment. Furthermore, its balanced class distribution supports unbiased training behavior and meaningful benchmarking across multiple quantization experiments and deployment conditions. It also enables faster convergence during training and simplified preprocessing for repeated experimental evaluations.

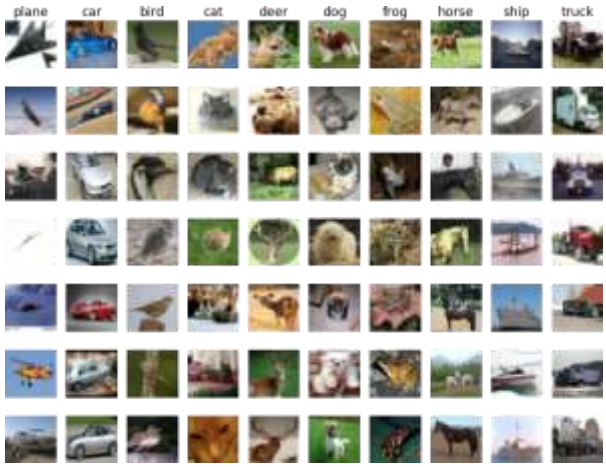

Fig - 2: CIFAR-10 Dataset Samples

B. Preprocessing

Preprocessing plays a crucial role in preparing input data for stable and efficient neural network training. Variations in raw image data, such as differences in pixel intensity distribution, can negatively impact model convergence if not properly handled.

The preprocessing pipeline includes normalization, where pixel values are scaled to a standardized range to improve numerical stability during training. Images are maintained at a fixed size of 32×32 pixels to ensure compatibility with the ResNet-18 architecture. Additionally, data augmentation techniques such as random horizontal flipping and cropping are applied to increase variability in the training data and enhance model generalization.

Batching is also implemented, grouping data samples into mini-batches to enable parallel processing and efficient GPU utilization. These preprocessing steps collectively ensure that the dataset is well-conditioned, allowing both the baseline FP32 and INT8 quantized models to learn robust feature representations and achieve consistent performance.

C. FP32 Model

The FP32 model serves as the reference framework for evaluating the impact of quantization on model performance. In this study, the ResNet-18 architecture is employed due to its efficient design and strong performance on image classification tasks. It consists of 18 convolutional layers with residual connections that facilitate gradient propagation, enabling stable training even in deeper networks.

The model is trained using full-precision 32-bit floating-point (FP32) representation on the CIFAR-10 dataset. Training is conducted using stochastic gradient descent (SGD) with a momentum factor of 0.9 to accelerate convergence. An initial learning rate of 0.1 is applied, along with a cosine annealing scheduler to gradually reduce the learning rate over time. The model is trained for 100 epochs with a batch size of 128, ensuring sufficient exposure to the dataset for optimal learning. Cross-entropy loss is used as the objective function for classification, while weight decay is incorporated as a regularization technique to prevent overfitting. During training, the model learns hierarchical feature representations, capturing both low-level textures and high-level semantic patterns.

The FP32 model achieves an accuracy of approximately 94.79% on the CIFAR-10 test set, establishing a strong benchmark for comparison. This performance represents the upper bound against which the INT8 quantized model is evaluated. By analyzing differences in accuracy, computational complexity, and memory usage, the baseline model provides a critical foundation for understanding the trade-offs introduced by reduced precision.

D. 8-Bit Quantization (INT8)

The INT8 quantization stage focuses on improving model efficiency while preserving accuracy through the use of Quantization-Aware Training (QAT). Unlike post-training quantization, which applies precision reduction after training, QAT integrates quantization effects directly into the training process. This allows the model to adapt to reduced numerical precision and maintain performance.

In this approach, both weights and activations are represented using 8-bit integers, resulting in approximately 4× reduction in memory usage compared to FP32 models. During training, fake quantization modules simulate integer operations, while observer mechanisms track activation ranges to determine appropriate scaling factors and zero-points. This ensures accurate mapping between floating-point and integer representations.

The quantized model is initialized from the pretrained FP32 baseline and fine-tuned for several epochs under quantization constraints. This refinement step helps recover potential accuracy loss and stabilizes the model under lower precision.

Experimental results indicate that the INT8 model achieves accuracy close to the FP32 baseline, typically around 94.60%, with negligible degradation of 0.19%. In addition to memory savings, the use of integer arithmetic enables faster inference and reduced energy consumption, making the model highly suitable for edge deployment.

Overall, INT8 quantization using QAT provides an effective balance between performance and efficiency, demonstrating its practicality for real-time applications in resource-constrained environments.

4. EVALUATION & RESULT

A. Performance Metrics

The evaluation of the image classification task was carried out using standard performance measures and metrics for quantitative assessment. Accuracy was considered as the primary metric to assess overall correctness of the predictions. Precision measures the proportion of correctly predicted samples within a given class.

Recall measures the ability of the model to capture all relevant samples, and the F1-score provides a balanced harmonic mean between precision and recall. A confusion matrix was also

employed to examine misclassifications across positive and negative sentiment classes.

B. Experimental Setup

The experiments were conducted using the CIFAR-10 dataset, a standard benchmark for image classification tasks. The dataset consists of labelled images across 10 classes, enabling reliable evaluation of model performance.

A ResNet-18 architecture was used as the baseline model and trained using full-precision (FP32). The trained model was then converted to an INT8 version using Quantization-Aware Training (QAT) for comparison. The experiments were performed in a GPU-enabled environment with the following configurations:

- **Hardware:** NVIDIA GPU (Google Colab environment with 16 GB RAM).
- **Software:** Python 3.x, PyTorch, TorchVision, and supporting libraries.
- **FP32 Training Parameters:** Batch size of 128, learning rate of 0.1, momentum of 0.9, weight decay of 0.0001, trained for 100 epochs using a cosine annealing learning rate scheduler.
- **INT8 (QAT) Training Parameters:** The pretrained FP32 model was converted to a quantization-aware model and fine-tuned using QAT. Training was continued for 30 epochs with a reduced learning rate of 1e-3, batch size of 128, and the same optimizer settings. Fake quantization modules and observers were enabled to simulate 8-bit precision for both weights and activations during training.
- **Data Split:** 50,000 images for training and 10,000 images for testing.

C. Evaluation & Result

The performance of the image classification model was evaluated using overall accuracy and a confusion matrix. These metrics help assess how well the ResNet-18 model classifies images across different categories.

Table I presents the comparison between FP32 and INT8 models. The INT8 model achieves accuracy very close to the FP32 baseline, showing only a minimal drop while significantly improving efficiency. Fig 3 and Fig 5illustrates the accuracy and loss during training across epochs of FP32 model and INT8 model respectively. The confusion matrices in Fig. 4 and Fig 6 shows the distribution of correct and incorrect predictions across all classes. Most misclassifications occur between visually similar categories such as animals or vehicles.

Table 1: Comparison of Model Performance

Model	Accuracy	Precision	F1 Score	Recall
FP32 Model	94.79%	94.79%	94.79%	94.78%
8 bit QAT Model	94.60%	94.59%	94.60%	94.59%

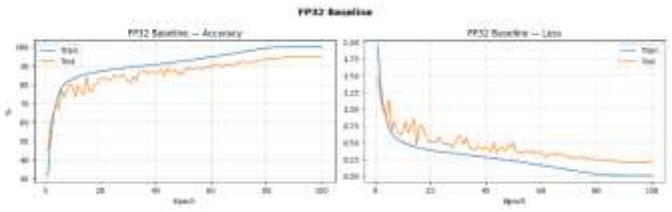

Fig - 3: FP32 Baseline Accuracy and Loss across Epochs

Fig - 4: Confusion Matrix of FP32

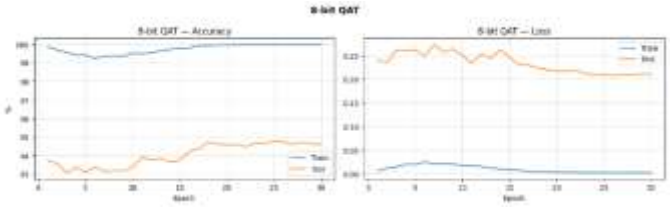

Fig - 5: 8 Bit quantized model Accuracy and Loss across Epochs

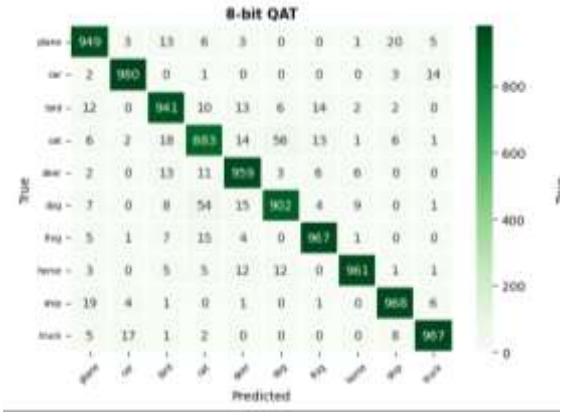

Fig - 6: Confusion Matrix of 8 bit model

5. CONCLUSION

This paper presents a framework for efficient image classification on edge devices by comparing full-precision (FP32) and 8-bit quantized (INT8) models using Quantization-Aware Training (QAT). The system is built on the ResNet-18 architecture and evaluated on the CIFAR-10 dataset. QAT is employed to incorporate quantization effects during training, enabling the model to adapt to reduced precision while maintaining performance. The implementation is carried out using PyTorch in a GPU-enabled environment, ensuring efficient training and evaluation.

Experimental results show that INT8 quantization achieves accuracy very close to the FP32 baseline, with only a slight decrease from 94.79% to 94.60%, while significantly reducing model size and computational overhead. This demonstrates the effectiveness of INT8 in achieving efficient inference without substantial loss in accuracy.

The proposed approach highlights the practicality of deploying quantized models in resource-constrained environments. Future work may explore lower bit-width quantization, hardware-specific optimizations, and advanced compression techniques to further improve efficiency. Overall, the study confirms that QAT-based INT8 quantization enables scalable and high-performance deep learning deployment for real-time edge applications.

REFERENCES

[1] B. Zhao, Y. Li, J. Zuo, W. Zhang, X. Chen, S. Cao and Z. & Jiang, "MPQA: Mixed-Precision Quantization Accelerator for CNN Inference," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2025.

[2] X. Shen, W. Ma, J. Liu, C. Yang, R. W. Q. Ding and J. Gu, "QuartDepth: Post-Training Quantization for Real-Time Depth Estimation on the Edge," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025.

[3] A. Gholami, S. Kim, Z. Dong, Z. Yao, M. Mahoney and K. Keutzer, "A survey of quantization methods for efficient neural network inference," in *In Low-power computer vision*, Chapman and Hall/CRC, 2022, pp. 291-326.

[4] Y. Li, R. Gong, X. Tan, Y. Yang, P. Hu, Q. Zhang, F. Yu, W. Wang and S. Gu, "BRECQ: Pushing the Limit of Post-Training Quantization by Block Reconstruction," 2021.

[5] 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 (CVPR)*, 2020.

[6] M. Nagel, M. van Baalen, T. Blankevoort and M. Welling, "Data-Free Quantization Through Weight Equalization and Bias Correction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[7] R. Banner, I. Hubara, E. Hoffer and D. Soudry, "Scalable Methods for 8-Bit Training of Neural Networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[8] B. Jacob, S. Kligys, B. Chen, M. Zhu, M. Tang, A. Howard, H. Adam and D. 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 (CVPR)*, 2018.

[9] R. Krishnamoorthi, "Quantizing Deep Convolutional Networks for Efficient Inference: A Whitepaper," 2018.

[10] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[11] A. Krizhevsky, "CIFAR-10 (Canadian Institute for Advanced Research) Dataset," [Online]. Available: https://www.cs.toronto.edu/kriz/cifar.html. [Accessed 16 April 2026].