Performance Analysis of Post-Training Quantization for CNN-based Conjunctival Pallor Anemia Detection

Sebastián A. Cruz Romero[1] and Wilfredo E. Lugo Beauchamp[1]

Computer Science and Engineering, University of Puerto Rico at Mayagüez, Mayagüez, Puerto Rico, USA
sebastian.cruz6@upr.edu, wilfredo.lugo1@upr.edu

Abstract. Anemia is a widespread global health issue, particularly among young children in low-resource settings. Traditional methods for anemia detection often require expensive equipment and expert knowledge, creating barriers to early and accurate diagnosis. To address these challenges, we explore the use of deep learning models for detecting anemia through conjunctival pallor, focusing on the CP-AnemiC dataset, which includes 710 images from children aged 6–59 months. The dataset is annotated with hemoglobin levels, gender, age, and other demographic data, enabling the development of machine learning models for accurate anemia detection. We use the MobileNet architecture as a backbone, known for its efficiency in mobile and embedded vision applications, and fine-tune our model end-to-end using data augmentation techniques and a cross-validation strategy. Our model implementation achieved an accuracy of 0.9313, a precision of 0.9374, and an F1 score of 0.9773, demonstrating strong performance on the dataset. To optimize the model for deployment on edge devices, we performed post-training quantization, evaluating the impact of different bit-widths (FP32, FP16, INT8, and INT4) on model performance. Preliminary results suggest that while FP16 quantization maintains high accuracy (0.9250), precision (0.9370) and F1 score (0.9377), more aggressive quantization (INT8 and INT4) leads to significant performance degradation. Overall, our study supports further exploration of quantization schemes and hardware optimizations to assess trade-offs between model size, inference time, and diagnostic accuracy in mobile healthcare applications.

Keywords: computer-aided diagnosis, anemia detection, convolutional neural network, post-training quantization

1 Introduction

Anemia is a widespread global health concern that mostly affects women and children in low- and middle-income countries (LMICs); symptoms like exhaustion and weakened immune systems may have an impact on a child's development. [1, 3] Standard diagnostic methods for anemia involve measuring blood hemoglobin

(Hb) levels, which require specialized equipment and personnel; resources often limited in rural and underserved areas. [1, 2] According to the World Health Organization, anemia affects over 40% of children aged 6 to 59 months and 37% of pregnant women globally, with the highest rates observed in LMICs with limited access to healthcare. [3] Given these challenges, there is a growing interest in non-invasive, portable diagnostic tools for early anemia detection to enable preventive interventions. [4, 2]

One non-invasive approach explores computer-aided diagnostic (CAD) systems capable of analyzing physiological indicators such as changes in pigmentation in conjunctiva pallor as a diagnostic indicator for anemia due to its direct relationship with Hb levels and ease of use. [5] Among pallor assessment areas (nail beds, palms, tongue), the conjunctiva is considered particularly sensitive for detecting anemia, due to its minimal intervening tissue layers and direct vascular access. [5, 4] However, the feasibility of conjunctival pallor-based diagnostic tools for anemia detection, especially in resource-limited settings, remains a growing area of research interest without clear consensus on optimal implementation approaches. [6]

1.1 Prior Work on Anemia Detection with CNNs

With advancements in artificial intelligence (AI) and deep learning (DL), non-invasive anemia diagnosis has gained significant attention as an alternative to traditional clinical methods [4]. Deep learning-based computer-aided diagnosis (CAD) models have been developed to identify anemia status or estimate hemoglobin (Hb) levels from pallor-based features. However, many of these models rely on proprietary datasets, limiting generalization due to a lack of data diversity [4, 7]. The recently introduced CP-AnemiC dataset addresses some of these limitations by providing a publicly available, large-scale dataset focused on conjunctival pallor for anemia detection in children. It includes diverse samples collected from ten regions in Ghana, improving model robustness and generalization [7].

However, most state-of-the-art (SOTA) models tested are computationally intensive, requiring significant memory and processing power. As shown in Table 1, CNN-based architectures commonly used in CAD demand substantial computational resources, making them impractical for real-world deployment in resource-limited settings.

1.2 Quantization for Model Compression

Deploying high-computation models in resource-limited environments remains a major challenge [8]. These models often require billions of parameters, leading to increased storage and inference costs compared to traditional algorithms [9]. To address this issue, quantization has emerged as a key technique for reducing model size while maintaining high predictive performance. Quantization approximates floating-point values from neural network weights using lower bit-width representations, significantly reducing memory and computational overhead.

Model	Size (MB)	Total Parameters	FLOPs	Image Input Size
AlexNet (2012)	238	62,378,344	2.27B	227x227
VGGNet (2014)	528	138,423,208	30.94B	224x224
ResNet (2015)	102	25,557,032	4.1B	224x224
MobileNet (2017)	17	4,221,000	0.569B	224x224
ViT (2021)	330	86,567,656	17.6B	224x224
ConvNeXt (2022)	338	88,591,464	15.4B	224x224

Table 1. Computational complexity and storage requirements of CNN architectures commonly used in computer-aided diagnosis. Versions of models are the following: VGG16, ResNet18, ViT B16, ConvNeXt Base, and MobileNet-224-1.0

By converting 32-bit floating-point (FP32) values into lower-precision formats such as 8-bit integer (INT8), quantization minimizes storage requirements and speeds up inference by enabling efficient integer-based computations. This technique enhances the feasibility of deploying deep learning models on edge devices with constrained resources [10, 11]. The balance between model efficiency and performance is critical for practical implementation in real-world anemia detection systems.

While this conversion introduces minor approximation errors, several quantization strategies help manage the precision-accuracy trade-off. Post-Training Quantization (PTQ), a widely used method, applies quantization after model training without requiring additional labeled data, making it computationally economical, though at a slight cost to accuracy. In contrast, Quantization-Aware Training (QAT) integrates quantization into the training process, allowing the model to adapt and retain higher accuracy, albeit with increased computational demand during training. [10, 11] Quantization can follow either uniform or non-uniform schemes; uniform quantization assigns values evenly across intervals, offering simplicity and speed, while non-uniform quantization tailors interval sizes to the data distribution, though at higher computational complexity. PTQ can be applied in static or dynamic forms. Post-training dynamic quantization reduces the bit representation of weights and activations during inference, decreasing computational load and memory usage. This differs from post-training static quantization, which uses a calibration dataset to precompute quantization parameters, including scaling factors for weights and activations, prior to deployment. [10, 11].

However, while certain studies report that quantized models might retain a degree of diagnostic accuracy comparable to full-precision models, more exploration is needed to confirm the consistency of these outcomes in specific diagnostic contexts, such as conjunctival pallor-based anemia detection. In this paper, we introduce a CNN-based classifier with the MobileNet [12] model for conjunctival pallor-based anemia detection that has comparable performance to state-of-the-art models used for anemia detection. This project builds on the CP-AnemiC dataset, using it to explore the feasibility of employing PTQ to

compare inference performance and execution time at FP32, FP16, INT8, and INT4 bit representations.

2 Methodology

2.1 Dataset

The CP-AnemiC dataset is a large-scale, publicly available dataset created to address the challenges in anemia detection through conjunctival pallor analysis. It includes 710 conjunctival images from children aged 6–59 months, collected from ten healthcare facilities in Ghana between January and June 2022. Of these, 424 images (60%) are labeled as anemic and 286 (40%) as non-anemic, based on the WHO threshold of hemoglobin (Hb) levels below 11 g/dL for anemia diagnosis. This diversity is aimed at enhancing the generalizability of models trained on it. The mean participant age is 31.58 months, with 306 females (43%) and 404 males (57%). Each image is annotated with Hb levels, age, gender, collection site, and remarks from laboratory assessments. The dataset also provides demographic analyses of Hb concentration by age and gender, highlighting lower Hb levels in anemic participants.

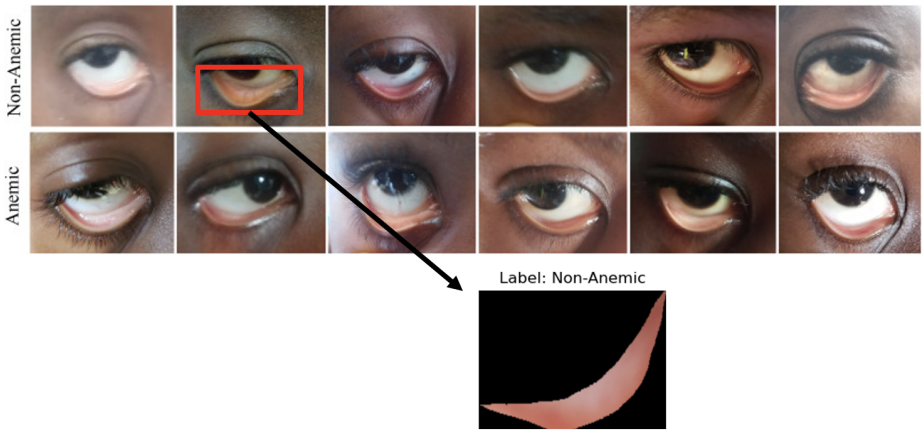

Fig. 1. Sample images of conjunctival pallor with an example region-of-interest from the CP-AnemiC dataset. The first row represents images from non-anemic patients, while the second row represents images from anemic patients.

2.2 Experimental Setting

We utilize the MobileNet [12] architecture as the backbone for our anemia classification model, fine-tuning the weights from a pre-trained model on the ImageNet [15] dataset, as illustrated in Figure 2. Our experimental setup follows the

Patient Class	**Anemic**	**Non-anemic**	**Total**
Patients	424	286	710
Female	174	132	306
Male	250	154	404
Age (months)	31.04 ± 17.02	32.31 ± 16.46	31.58 ± 16.78
Anemia Diagnosis for Age 6–59 months			
Anemia Classification	**Anemic**	**Non-anemic**	
Hemoglobin Levels	<11 g/dL	≥ 11 g/dL	

Table 2. A patient-level characteristics summary of the dataset

approach outlined by Appiahene et al. [7], which employs CNN-based anemia detection alongside data augmentation techniques, including random horizontal flips, rotations, shifts, and scaling, to enhance model generalization.

MobileNet is particularly well-suited for this task due to its lightweight design and computational efficiency, making it ideal for deployment in resource-constrained environments. As shown in Table 1, traditional CNN architectures such as AlexNet, VGGNet, ResNet, ViT, and ConvNeXt require significantly larger storage and perform a higher number of floating-point operations (FLOPs), leading to increased inference time and higher computational costs. In contrast, MobileNet employs depthwise separable convolutions, drastically reducing the number of parameters and FLOPs while maintaining competitive accuracy.

A 5-fold cross-validation strategy was implemented, where four folds were used for training and one for testing. The model weights were randomly initialized at the start of each fold, and training was performed with a batch size of 32 on an NVIDIA GeForce RTX 4090 GPU. We used Binary Cross-Entropy as the loss function and optimized the model with the Adam optimizer, setting the learning rate to 10^{-4}. The fully connected layer was modified to output a binary classification target, where 1 represents an anemic class and 0 represents a non-anemic class, using a sigmoid activation function for the output. Training was conducted end-to-end for a maximum of 150 epochs, with early stopping applied if there was no improvement in F1 score for 10 consecutive epochs during the validation step. We compute model evaluation performance metrics as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{1}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{AUC} = \text{Recall}(\text{TPR}) - (1 - \text{Precision})(\text{FPR}) \tag{4}$$

where TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives, respectively.

2.3 Quantization and Inference

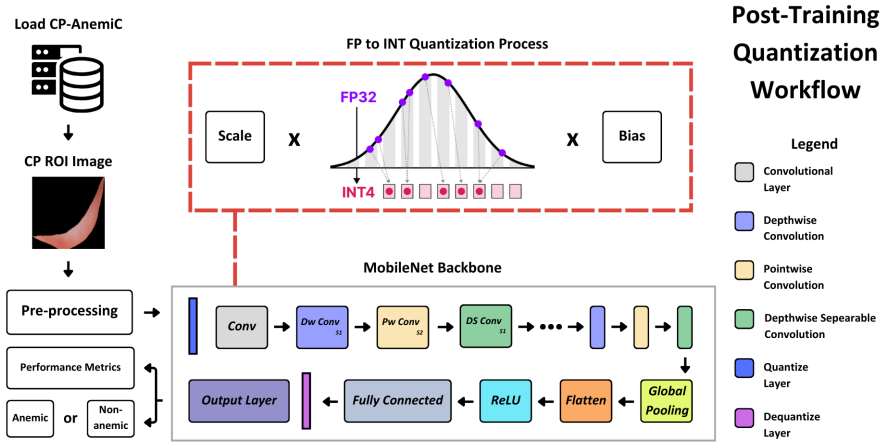

Fig. 2. MobileNet architecture used as a backbone for our anemia detection task. A conjunctival pallor region-of-interest is used as input and the model outputs a numerical representation of anemic and non-anemic classes. The model is quantized for certain layers to the bit-width representation that is selected.

We perform post-training quantization (PTQ) to optimize our fine-tuned MobileNetV2 model by reducing the bit-width of model weights and activations. The model weights are converted into the Open Neural Network Exchange (ONNX) [13] format, a universal representation that ensures compatibility across frameworks and optimization tools. This conversion involves exporting the PyTorch model and mapping functional operators to ONNX equivalents while preserving the computational graph. ONNX facilitates backend-specific optimizations by standardizing model representation. In our workflow, NVIDIA's ModelOpt [16] offers quantization configurations with TensorRT [14] as the backend to dynamically select the optimal precision for operators in each layer based on computational efficiency and accuracy requirements. TensorRT applies these optimizations layer-by-layer, leveraging Tensor Cores for FP16 operations and integer arithmetic units for INT8 and INT4 operations. Dequantization is performed as needed to ensure compatibility between mixed-precision layers. For FP16 conversion, each 32-bit floating-point value (x) is truncated to 16 bits by retaining a reduced number of mantissa bits, represented as:

$$\text{FP16 Value} = \text{Round}\left(\frac{x}{2^k}\right) \cdot 2^k \tag{5}$$

where,

– k determines the precision range.

For INT8 and INT4 quantization, activations are mapped to integer ranges using scale factors derived from calibration data. We used ModelOpts INT8 default configuration to enable 8-bit precision for weights and activations. Weight quantization is performed per-channel, while activation quantization adopts a per-tensor approach. The MaxCalibrator algorithm determines scale factors by computing the maximum absolute value across tensors, ensuring robust mapping from FP32 to INT8. This configuration prioritizes minimal accuracy degradation while achieving significant computational efficiency.

$$q_x = \text{Round}\left(\frac{x}{s_x}\right), \quad s_x = \frac{\max(|x|)}{2^{b-1}-1} \tag{6}$$

where,

– x represents the tensor (weights or activations),
– b is the number of bits (e.g., b=8 for INT8 or b=4 for INT4).

For INT4 quantization, we employed the Advanced Weight Quantization (AWQ) configuration, which utilizes block-wise quantization for weights with block sizes of 128 elements. Activations are excluded from quantization to reduce precision loss. AWQ is used with the "awq_lite" method, iteratively adjusting scaling parameters (α) in small steps to minimize quantization error. This approach achieves extreme compression, targeting environments with strict memory and computational constraints.

$$\alpha^{(t+1)} = \alpha^{(t)} - \eta \cdot \nabla_\alpha \mathcal{L}(w, q_w), \tag{7}$$

where,

– η is the step size,
– $\mathcal{L}(w, q_w)$ is the loss function measuring quantization error.

3 Results

3.1 Anemia Detection

The model was initialized to train over 150 epochs through 5-fold cross validation due to the limited dataset size. Validation F1 score was monitored to save model weights optimal for running inference at different bit-widths. The highest validation F1 score of 0.9773 was observed at 26 epochs with a validation accuracy of 96.88% and precision of 97.13% as observed in Table 4. However, to avoid overfitting, early stopping was triggered after 26 epochs indicating model convergence, as further F1 score improvements were minimal.

Fold	Loss	Accuracy ↓	Precision	Recall	F1 Score	AUC Score
74	0.2269	0.9229	0.9252	0.9531	0.9383	0.9667
98	0.2157	0.9208	0.9305	0.9402	0.9331	0.9680
100	0.2448	0.9160	0.9370	0.9267	0.9257	0.9636
99	0.2314	0.9122	0.8869	0.9372	0.9090	0.9759
80	0.2090	0.9104	0.9170	0.9293	0.9219	0.9705
103	0.2023	0.9092	0.9312	0.9229	0.9249	0.9755
86	0.2394	0.9090	0.9160	0.9271	0.9208	0.9710
101	0.2230	0.9076	0.9598	0.8934	0.9225	0.9767
71	0.2663	0.9021	0.9071	0.9307	0.9176	0.9519
78	0.2495	0.9000	0.9244	0.9155	0.9166	0.9573
91	0.2521	0.8988	0.9085	0.9252	0.9132	0.9614
79	0.2356	0.8979	0.9046	0.9286	0.9127	0.9633
65	0.2660	0.8972	0.9329	0.8979	0.9115	0.9591
76	0.2808	0.8958	0.9107	0.9162	0.9072	0.9464
85	0.2973	0.8951	0.9047	0.9191	0.9061	0.9527
92	0.2322	0.8951	0.9057	0.9213	0.9097	0.9677
73	0.2531	0.8938	0.9204	0.9063	0.9104	0.9571
75	0.2741	0.8935	0.9265	0.9098	0.9147	0.9585
84	0.2487	0.8931	0.8993	0.9221	0.9071	0.9624
96	0.2725	0.8924	0.9124	0.9144	0.9064	0.9588
93	0.2823	0.8903	0.9084	0.9045	0.9053	0.9653
89	0.3117	0.8882	0.9086	0.9010	0.9021	0.9559
88	0.2817	0.8854	0.8947	0.9079	0.8992	0.9504
87	0.2823	0.8851	0.8797	0.9345	0.9001	0.9526
82	0.2719	0.8813	0.9103	0.8968	0.9001	0.9502

Table 3. Training performance of the anemia classification model across cross-validation folds. Each fold represents a unique dataset split, ordered by accuracy.

3.2 Performance Comparison at Different Quantization Levels

FP32 achieves the best performance, with a loss of 0.2141, accuracy of 93.13%, precision of 93.74%, recall of 95.00%, F1 score of 0.9428, and AUC score of 0.9657 as shown in Table 5. FP16 follows closely, showing only a slight drop in performance with a loss of 0.2149, accuracy of 92.50%, and similar precision, recall, and F1 scores, maintaining a strong AUC of 0.9654. In contrast, INT8 quantization leads to a substantial decrease in accuracy (71.25%) and AUC (90.05%), with a much higher loss of 0.7441, indicating that the model's performance suffers despite lower computational requirements. The INT4 quantization results in a drastic performance drop, with a loss of 2.3136, accuracy of just 43.13%, and poor precision (20.00%) and recall (1.00%).

3.3 Quantized Layers for Integer Arithmetic

Table 6 compares memory consumption and execution latency across different quantization levels, illustrating key trade-offs. FP16 achieves substantial memory

Fold	Loss	Accuracy ↓	Precision	Recall	F1 Score	AUC Score
12	0.0857	0.9688	0.9713	0.9722	0.9705	0.9978
13	0.1033	0.9688	0.9773	0.9773	0.9773	0.9923
11	0.1225	0.9531	0.9565	0.9659	0.9602	0.9920
10	0.1273	0.9453	0.9268	0.9747	0.9481	0.9910
9	0.1846	0.9315	0.8970	1.0000	0.9453	1.0000
8	0.3033	0.8768	0.9147	0.8728	0.8915	0.9497
7	0.3598	0.8162	0.8481	0.8302	0.8382	0.9189
6	0.3852	0.7858	0.7981	0.8589	0.8239	0.9160
5	0.4081	0.7822	0.7570	0.9025	0.8181	0.8923
3	0.5210	0.7813	0.7964	0.8160	0.7993	0.8048
4	0.4401	0.7537	0.7365	0.8927	0.8017	0.8865
1	0.5704	0.6696	0.6731	0.8441	0.7435	0.7548
2	0.6277	0.6677	0.6700	0.8615	0.7458	0.6965
0	0.6060	0.6530	0.6696	0.8242	0.7292	0.6981

Table 4. Validation performance of the anemia classification model across cross-validation folds. Each fold represents a unique dataset split, ordered by accuracy.

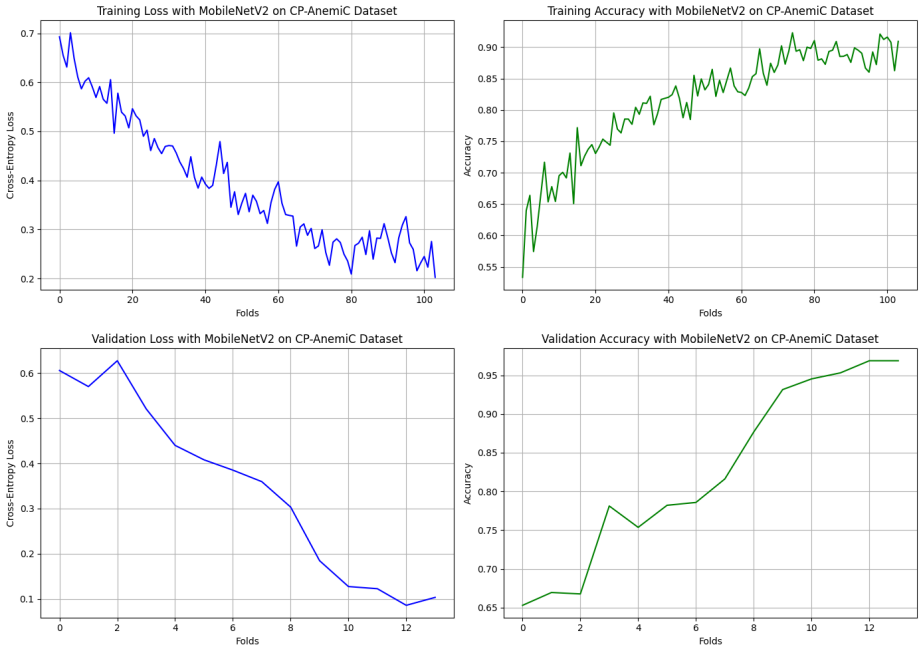

Fig. 3. Training and validation performance of our fine-tuned MobileNet for anemia detection. Early stopping halted training at epoch 26 after 10 epochs without F1 score improvement, preventing overfitting.

Bit-width	Loss	Accuracy	Precision	Recall	F1 Score	AUC Score
FP32	0.2141	0.9313	0.9374	0.9500	0.9428	0.9657
FP16	0.2149	0.9250	0.9370	0.9400	0.9377	0.9654
INT8	0.7441	0.7125	0.7697	0.7519	0.7607	0.9005
INT4	2.3136	0.4313	0.2000	0.0100	0.0196	0.6387

Table 5. Inference performance comparison of the model at different quantization levels. Higher bit-widths (FP32, FP16) maintain higher accuracy, while lower bit-widths (INT8, INT4) show performance degradation.

reduction (4.61 MB) while also yielding the fastest execution time (37.4 ms). INT8, despite reducing numerical precision, results in an unexpected model size increase (9.24 MB) and the highest execution latency (91.9 ms). We hypothesize that post-training quantization schemes for integer arithmetic cause significant memory increase due to additional quantization parameters stored alongside weights, which introduce overhead during inference [17, 18].

Bit-width	Model Size	Execution Time
FP32	9.13 MB	48.6 ms ± 235 μs
FP16	4.61 MB	37.4 ms ± 1.01 ms
INT8	9.24 MB	91.9 ms ± 1.92 ms
INT4	17.75 MB	49.5 ms ± 1.34 ms

Table 6. Memory consumption and execution time across different quantization levels.

INT4, while achieving extreme weight compression, paradoxically results in the largest model size (17.75 MB). This can be attributed to inefficient hardware handling of sub-byte precision, requiring extra padding and metadata storage to align with memory accesses [19, 20]. Furthermore, INT4 exhibits execution time (49.5 ms) comparable to FP32 (48.6 ms), indicating that ultra-low precision formats can introduce computational inefficiencies, particularly when not natively supported by the hardware.

The computational graph shown in Figure 4 illustrates a post-training quantization architecture that consists of operations for quantization, dequantization, convolution, batch normalization, clipping, and further processing. Quantize Linear operations convert data into lower precision format using a scale factor and zero point. Similarly, weights are also quantized before the first convolution operation takes the dequantized values and processes them. Batch Normalization is applied, adjusting the activations using learned parameters, and a Clip operation represents a ReLU activation. The repeated pattern of Quantized $\rightarrow$ Dequantize $\rightarrow$ Compute ensures compatibility with hardware accelerators that support low-precision arithmetic.

Table 7 summarizes the key layers affected by INT8 and INT4 quantization, detailing bit-width, quantization methods, and the range of activation maxima

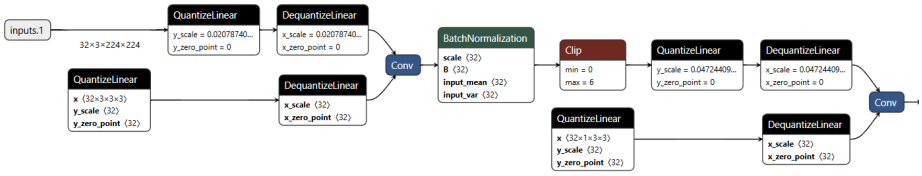

Fig. 4. Computational graph visualization of MobileNet's operations after INT8 post-training quantization, representing a portion of the CNN with quantized weights and activations. Diagram used NETRON's model visualizer API [21].

(amax). Notably, per-axis quantization in INT8 retains finer control over dynamic range, while block-wise INT4 quantization results in increased activation variation, potentially degrading numerical stability.

Layer Name	Bit-width	Quantization Method	amax Range
features.0.0	INT8	Per-axis	[0.0039, 1.4840]
features.1.conv.0.0	INT8	Per-axis	[0.0036, 2.6928]
features.10.conv.1.0	INT8	Per-axis	[0.0103, 0.6126]
features.0.0	INT4	Block-wise	[0.0005, 1.4840]
features.1.conv.0.0	INT4	Block-wise	[0.0014, 2.6928]
features.10.conv.1.0	INT4	Block-wise	[0.0035, 0.5219]

Table 7. Quantization summary for key layers. Comparison of bit-widths, quantization methods, and activation maximum (amax) ranges for selected layers.

4 Conclusion and Future Work

This study underscores the potential of lightweight architectures such as MobileNet for anemia detection through conjunctival pallor analysis, demonstrating that deep learning models can achieve high predictive accuracy while remaining computationally efficient. Our MobileNet-based model outperformed larger architectures such as ResNet50, VGG16, and ViT, achieving state-of-the-art performance on the CP-AnemiC dataset with an F1 score of 0.943 and an accuracy of 93.13% during inference without quantization. This highlights the efficacy of MobileNet's depthwise separable convolutions, which significantly reduce computational complexity without sacrificing predictive power.

While larger CNN architectures have shown promise in medical imaging tasks, their high computational overhead and memory demands make them im-

practical for deployment in resource-constrained environments such as point-of-care diagnostics and mobile health applications. As demonstrated in Table 8, MobileNet achieves comparable or superior performance to these models while requiring significantly lower memory and computation, making it a viable solution for real-world applications in remote and low-resource settings.

Model	Model Size (MB)	Accuracy	Precision	Recall	F1-Score
ResNet50	98	**84.79%**	**0.852**	0.830	**0.837**
VGG16	528	80.56%	0.807	0.785	0.799
DenseNet121	**33**	79.58%	0.790	0.788	0.786
ViT	330	84.08%	0.833	**0.835**	0.833
ConvNeXt Base	338	75.63%	0.749	0.739	0.740
MobileNet (FP32)	9.13	**93.13%**	**93.74**	**0.953**	**0.943**
MobileNet (FP16)	**4.61**	92.50%	0.937	0.940	0.934
MobileNet (INT8)	9.24	71.25%	0.770	0.752	0.761
MobileNet (INT4)	17.75	43.13%	0.200	0.010	0.020

Table 8. Final performance summary of models tested on CP-AnemiC and different MobileNet quantization levels.

To facilitate real-time inference on edge devices, we explored post-training quantization techniques to optimize model efficiency. Our results indicate that FP16 quantization effectively reduces model size (4.61 MB) while maintaining high accuracy (92.50%) and an F1 score of 0.934, making it an ideal balance between computational efficiency and predictive performance. Performance degradation was evident with more aggressive quantization techniques. INT8 quantization, while achieving a notable reduction in computational complexity, exhibited a substantial drop in accuracy (71.25%), limiting its effectiveness for clinical applications. INT4 quantization, in contrast, led to a severe degradation in performance (43.13% accuracy, F1 score of 0.020), rendering it unsuitable for real-world deployment. Preliminary results of model performance across different quantization bit-widths show that aggressive quantization can severely degrade the model's predictive capabilities.

Beyond standard quantization methods, future work includes systematically achieving full integer arithmetic within layer-by-layer computation. Furthermore, the impact of these optimizations on inference latency will be studied, particularly on edge devices such as the NVIDIA Jetson Xavier NX and TX2 NX due to their small form factor and low-power consumption. Using TensorRT for operator conversion backend, we aim to exploit its ability to select optimal precision per layer, leveraging mixed-precision arithmetic to enhance execution speed while maintaining diagnostic integrity. These methods can further enhance MobileNet's deployment feasibility by dynamically adjusting precision based on computational constraints while minimizing accuracy loss.

Acknowledgment This work is supported by the Center for Research & Development at the University of Puerto Rico at Mayagüez and the NSF-EPSCoR Center for the Advancement of Wearable Technologies, NSF grant OIA-1849243.

References

1. Chaparro, C.M., Suchdev, P.S.: Anemia epidemiology, pathophysiology, and etiology in low- and middle-income countries. Ann. N. Y. Acad. Sci. 1450(1), 15–31 (2019). doi:10.1111/nyas.14092
2. Garcia-Casal, M.N., Dary, O., Jefferds, M.E., Pasricha, S.R.: Diagnosing anemia: Challenges selecting methods, addressing underlying causes, and implementing actions at the public health level. Ann. N. Y. Acad. Sci. 1524(1), 37–50 (2023). doi:10.1111/nyas.14996
3. World Health Organization: Anaemia (2023). https://www.who.int/news-room/fact-sheets/detail/anaemia
4. An, R., Huang, Y., Man, Y., Valentine, R.W., et al.: Emerging point-of-care technologies for anemia detection. Lab Chip 21(10), 1843–1865 (2021). doi:10.1039/d0lc01235a
5. Sheth, T.N., Choudhry, N.K., Bowes, M., Detsky, A.S.: The relation of conjunctival pallor to the presence of anemia. J. Gen. Intern. Med. 12(2), 102–106 (1997). doi:10.1046/j.1525-1497.1997.00014.x
6. Merid, M.W., Chilot, D., Alem, A.Z.: An unacceptably high burden of anaemia and its predictors among young women (15–24 years) in low and middle income countries; setback to SDG progress. BMC Public Health 23, 1292 (2023). doi:10.1186/s12889-023-16187-5
7. Dapena, G., et al.: CP-AnemiC: A conjunctival pallor dataset and benchmark for anemia detection in children. Pattern Recognit. Lett. (2024). doi:10.1016/j.patrec.2024.03.019
8. Sharma, R., Alharbi, A., Alsarhan, A., et al.: Internet of Intelligent Things: A convergence of embedded systems, edge computing and machine learning. J. King Saud Univ. Comput. Inf. Sci. (2024). doi:10.1016/j.jksuci.2024.03.022
9. Canziani, A., Paszke, A., Culurciello, E.: An Analysis of Deep Neural Network Models for Practical Applications. CoRR abs/1605.07678 (2016). http://arxiv.org/abs/1605.07678
10. Nagel, M., Fournarakis, M., Amjad, R.A., et al.: A White Paper on Neural Network Quantization. CoRR abs/2106.08295 (2021). https://arxiv.org/abs/2106.08295
11. Gholami, A., Kim, S., Dong, Z., et al.: A Survey of Quantization Methods for Efficient Neural Network Inference. arXiv preprint (2021). https://arxiv.org/abs/2103.13630
12. Howard, A.G., Zhu, M., Chen, B., et al.: MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. arXiv preprint (2017). https://arxiv.org/abs/1704.04861
13. ONNX Runtime developers: ONNX Runtime (2021). https://onnxruntime.ai/
14. NVIDIA Corporation: TensorRT Developer Guide (2024). https://docs.nvidia.com/deeplearning/tensorrt/pdf/TensorRT-Developer-Guide.pdf
15. Deng, J., Dong, W., Socher, R., et al.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conf. Comput. Vis. Pattern Recognit., pp. 248–255. IEEE (2009)

16. NVIDIA: TensorRT Model Optimizer: PyTorch Quantization Guide (2024). https://nvidia.github.io/TensorRT-Model-Optimizer/index.html
17. Jacob, B., Kligys, S., Chen, B., et al. Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.
18. Krishnamoorthi, R. Quantizing Deep Convolutional Networks for Efficient Inference: A Whitepaper. arXiv preprint arXiv:1806.08342, 2018.
19. Banner, R., Nahshan, Y., & Soudry, D. Post Training 4-bit Quantization of Convolutional Networks for Rapid Deployment. Advances in Neural Information Processing Systems (NeurIPS), 2019.
20. Wu, S., Xu, J., Dai, X., et al. Integer Quantization for Deep Learning Inference: Principles and Empirical Study. International Conference on Learning Representations (ICLR), 2020.
21. Roeder, L. Netron: Visualizer for Neural Network, Deep Learning, and Machine Learning Models. Available at: https://www.lutzroeder.com/ai