Efficient Land-Cover Image Classification via Mixed Bit-Precision Quantization

Tushar Shinde, Ahmed Silima Vuai

Indian Institute of Technology Madras Zanzibar

Introduction

- Deep neural networks (DNNs) are highly effective for land cover classification using remote sensing data.
- Deploying DNNs on edge devices is challenging due to resource constraints; uniform quantization helps but may reduce accuracy due to varying layer sensitivity.

Motivation

- Not all layers contribute equally to a model's accuracy — uniform quantization may either waste compression potential or harm performance.
- A layer-aware, adaptive quantization strategy is needed to: 1) Minimize model size, 2) Preserve classification performance, and 3) Enable efficient deployment on edge devices (e.g., drones).

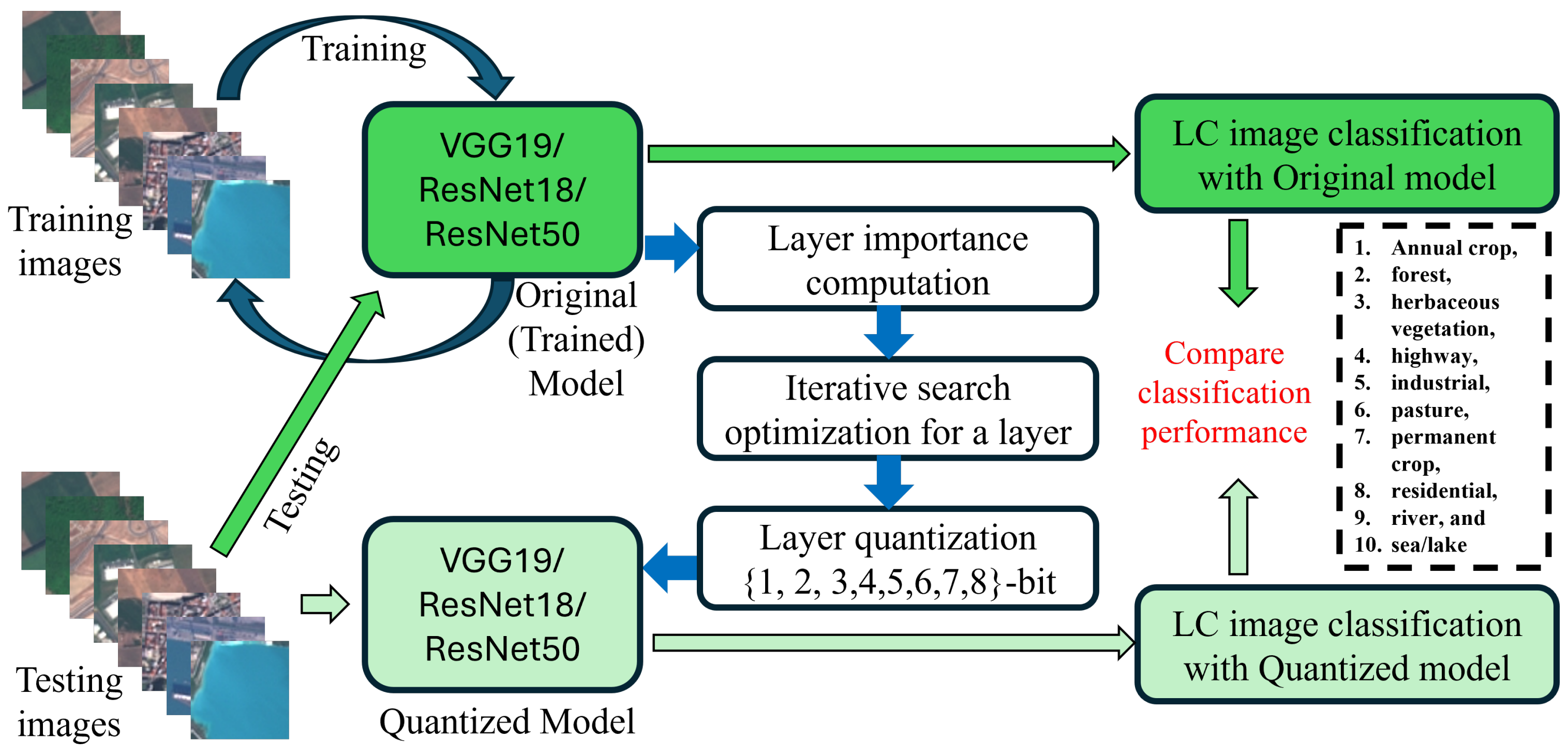
Objectives

The present study investigates the following objectives:

- Design an adaptive layer-wise quantization strategy that assigns different bit-widths to neural network layers based on their importance.
- Reduce model size and computational load while maintaining high classification accuracy for land-cover image analysis.

Method

The proposed method introduces an adaptive quantization framework for compressing deep neural networks (DNNs) while preserving performance in land-cover classification.



Training
Training images
VGG19/ ResNet18/ ResNet50
Original (Trained) Model
Testing
Testing images
VGG19/ ResNet18/ ResNet50
Quantized Model
Layer importance computation
Iterative search optimization for a layer
Layer quantization {1, 2, 3,4,5,6,7,8}-bit
LC image classification with Original model
Compare classification performance
LC image classification with Quantized model
1. Annual crop,
2. forest,
3. herbaceous vegetation,
4. highway,
5. industrial,
6. pasture,
7. permanent crop,
8. residential,
9. river, and
10. sea/lake

Figure 1. The framework of the proposed approach

Layer Importance Computation

To balance bit-precision and accuracy, we compute the importance of each layer using three metrics:

- **Normalized Parameter Proportion:**

$$N_P(l) = \frac{\text{Number of parameters in layer } l}{\text{Total number of parameters in the model}}$$

- **Normalized Zero-Order Entropy:**

$$N_E(l) = \frac{\text{Zero-order entropy of parameters in layer } l}{\text{Bit-precision of the quantized model}}$$

- **Normalized Variance:**

$$N_V(l) = \log\left(e - 1 + \frac{\text{Variance of parameters in layer } l}{\max_k \text{Variance of parameters in layer } k}\right)$$

The overall **Importance Score** for each layer is computed as:

$$\text{Importance}(l) = w_P \cdot N_P(l) + w_E \cdot (1 - N_E(l)) + w_V \cdot (1 - N_V(l))$$

Here, w_P, w_E, and w_V are weights assigned to each component. Layers with higher parameter count, entropy, or variance tend to be more important. Empirically, the initial and final layers of deep neural networks often rank highest in importance.

Layer-wise Bit-Width Selection Algorithm

- Each layer's bit-width $b(l)$ is selected based on its importance score to balance compression and accuracy.
- Search begins from low bit-width (e.g., 1-bit) and increases only if model accuracy drops beyond a threshold.
- A layer-specific margin $T_{\text{margin}}(l) = T_{\text{margin}}^{\text{init}} \cdot \text{Importance}(l)$ controls acceptable accuracy loss.
- The method ensures minimal model size while preserving accuracy for edge deployments.

Experimental Setup

Dataset: We used the EuroSAT dataset comprising 27,000 Sentinel-2 images (64×64 pixels, 13 spectral bands). It includes 10 LULC classes such as forest, crop, and river. Data augmentation techniques (flips, transforms) were applied.

Platform: All experiments were conducted on Kaggle using NVIDIA Tesla P100/G4 GPUs with PyTorch and supporting Python libraries.

Models: VGG19, ResNet18, and ResNet50 were trained from scratch for 120 epochs using SGD, cross-entropy loss, and a learning rate scheduler.

Evaluation: An 80/20 train-test split was used. Only post-training quantization was applied (no quantization-aware training).

Hyperparameters: Layer importance weights were set equally ($w_P = w_E = w_V = 1/3$). Bit-width quantization used 8-bit as baseline, with an initial threshold margin of 0.5.

Results and Analysis

We evaluated our adaptive quantization method on VGG19, ResNet18, and ResNet50 using the EuroSAT dataset. The technique effectively preserved model accuracy while significantly reducing bit-width.

Table 1. Model accuracy and Average bit-width comparison for different DNN architectures at fixed and adaptive quantization.

Dataset	VGG19	ResNet18	ResNet50
Original (32-bit)	96.06%	94.78%	94.61%
Fixed (8-bit) Q	96.04%	94.70%	94.70%
Fixed (7-bit) Q	96.07%	94.67%	93.24%
Fixed (6-bit) Q	96.13%	94.63%	93.81%
Fixed (5-bit) Q	95.96%	94.43%	93.83%
Fixed (4-bit) Q	95.52%	90.09%	87.50%
Fixed (3-bit) Q	90.41%	63.93%	29.28%
Fixed (2-bit) Q	9.30%	9.41%	11.11%
Fixed (1-bit) Q	11.11%	9.26%	11.11%
Proposed Adaptive Quantization			
Fixed $T_{margin} = 0.5$ (Average bit-width)	95.78% (**1.38**)	93.83% (3.73)	93.74% (3.74)
Adaptive $T_{margin} = 0.5$ (Average bit-width)	**96.06%** (1.44)	**95.07%** (**3.43**)	**94.44%** (**3.70**)

Table 2. Comparison with the existing studies (without pre-trained weights). The parameters size is shown (in bits) as multiplication of number of parameters and the average bit-width of each parameter.

Model	Accuracy	Parameters Size
ResNet50	96.43%	24M × 32
ResNet50	96.57%	24M × 32
ResNet101	97.22%	43M × 32
AlexNet	83.7%	61M × 32
DenseNet	92.5%	29M × 32
MobileNetV3	87.7%	6M × 32
EfficientNetV2	90.8%	24M × 32
ViT32	91.7%	86M × 32
SwinB	92.6%	88M × 32
Proposed AQ VGG19	**96.06%**	144M × **1.44**
Proposed AQ ResNet50	**94.44%**	24M × **3.70**
Proposed AQ ResNet18	**95.07%**	12M × **3.43**

Conclusion

This study introduced an **adaptive layer-wise quantization technique** for deep neural networks, guided by **layer importance metrics**. Evaluated on VGG19, ResNet18, and ResNet50 using the **EuroSAT dataset**, the proposed method:

- Achieved **accuracy nearly equal** to the full-precision models.
- Significantly **reduced the average bit-width**, resulting in compact models.
- Outperformed traditional **fixed-precision quantization** methods.
- Demonstrated strong potential for deployment on **resource-constrained edge devices**.

Future Work

Future extensions of this work will focus on:

- Exploring **additional network architectures and datasets** to evaluate generalizability.
- Investigating **dynamic quantization during inference** to support real-time applications.
- Combining with other compression techniques like **model pruning** for enhanced efficiency.
- Leveraging a **broader range of spectral bands** in satellite imagery to improve classification reliability and interpretability.

References

[1] X. Zhang, G. Chen, W. Wang, Q. Wang, and F. Dai, "Object-based land-cover supervised classification for very-high-resolution uav images using stacked denoising autoencoders," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 7, pp. 3373–3385, 2017.

[2] M. Khan, A. Hanan, M. Kenzhebay, M. Gazzea, and R. Arghandeh, "Transformer-based land use and land cover classification with explainability using satellite imagery," *Scientific Reports*, vol. 14, no. 1, p. 16744, 2024.

[3] 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*, pp. 770–778, 2016.

[4] C. Tang, K. Ouyang, Z. Wang, Y. Zhu, W. Ji, Y. Wang, and W. Zhu, "Mixed-precision neural network quantization via learned layer-wise importance," in *European Conference on Computer Vision*, pp. 259–275, Springer, 2022.

[5] T. Shinde, "Adaptive quantization and pruning of deep neural networks via layer importance estimation," in *Workshop on Machine Learning and Compression, NeurIPS 2024*, 2024.

Acknowledgment

This work was in part supported by the Walmart Center of Technical Excellence (IIT Madras) Project Grant Award.