engineering proceedings

MDPI

Proceedings Paper

A Lightweight Deep Learning Model for Identifying Weeds in Corn and Soybean Using Quantization †

Alex Aaron [1,*], Muhammad Hassan [2], Mohamed Hamada [3,*] and Habiba Kakudi [4]

[1] Department of Computer Science, Bayero University, Kano, Nigeria; alex.aaron.cs@gmail.com
[2] Department of Software Engineering, Bayero University, Kano, Nigeria; mhassan.se@buk.edu.ng
[3] Software Engineering Lab, University of Aizu, Japan; hamada@u-aizu.ac.jp
[4] Department of Computer Science, Bayero University, Kano, Nigeria; hakakudi.cs@buk.edu.ng
* Correspondence: alex.aaron.cs@gmail.com

† Presented at the The 4th International Electronic Conference on Applied Sciences, 27 Oct–10 Nov 2023; Available online: https://asec2023.sciforum.net/.

Abstract: Deep learning models are applied in precision agriculture for site-specific weed management by identifying weeds in farmlands. Unfortunately, because deep learning models are usually large, they are rarely adopted in resource-constrained devices (like edge devices) used in precision agriculture. In this study, we propose a lightweight deep learning model for detecting weeds in corn and soybean plants. We used transfer learning to train an InceptionnetV3 model for the task. The dataset used consists of a total of 13,177 samples of corn, soybean, and weeds. The InceptionV3 model, whose size is 183.34MB, achieved a classification accuracy of 97%. We then applied the quantization technique to reduce the size of the model. The quantized model was reduced to a size of 23.38MB, achieving an accuracy of 87%. The results show that quantization can reduce the size of a deep learning model while maintaining a reasonable amount of its performance [1].

Keywords: deep learning; quantization; precision agriculture

Citation: Aaron, A.; Hassan, M.; Hamada, M.; Kakudi, H. A Lightweight Deep Learning Model for Identifying Weeds in Corn and Soybean Using Quantization. *Eng. Proc.* **2023**, *52*, x. ttps://doi.org/10.3390/xxxxx

Academic Editor(s):

Received: date
Revised: date
Accepted: date
Published: date

1. Introduction

Weeds are unwanted plants that grow along with cash and food crops [2]. They compete for water, nutrients, space, and sunlight, making it difficult for these crops to thrive. This problem causes a significant reduction in crop yield. Weeds cause about a 30% reduction in crop yield worldwide. For example, every year, weeds cause about $25 billion loss for North America's corn and soybean crops. Therefore, there is a need to control the growth of weeds in farmlands. Weed control usually involves the spread of herbicides on farmlands. Herbicides can pollute land and deposit hazardous chemical substances on crops if applied carelessly. When using herbicides, farmers should target weeds. For site-specific weed management, automated techniques have been employed in precision agriculture, where weeds are precisely detected and treated with herbicides. These computerized techniques include machine learning and computer vision. Support Vector Machines and other machine learning approaches have been used to train computers to recognize weeds [3].

For instance, programmers can teach robotic devices with vision technology to identify weeds in corn and soybean plants. Deep learning is a subset of machine learning in which convolutional neural networks train models for tasks like object classification and detection [4]. However, these models are often characterized by large sizes, making them challenging to deploy in low-resource devices for precision agriculture. Model compression techniques (such as weight quantization, pruning, and knowledge) must be applied to these models to reduce their size. This makes it possible to deploy them in low-resource devices. Quantization involves converting model weights from high-precision floating-point form to low-precision floating-point or integer representation, such as 16-bit or 8-bit. One can change the high-precision floating-point representation of a model's weights to a lower-precision form, reducing the model's size and inference time (latency) without compromising too much accuracy. Weight quantization is a method used to reduce the model’s size. On the other hand, activation quantization is applied to enhance the model's latency. Additionally, quantization will enhance a model's performance by lowering memory bandwidth needs and raising cache utilization [5].

i. Weight Quantization: By applying weight quantization, the model size is decreased, and the training and inference processes are sped up. Lowering the number of bits required to represent the weight matrices, suggested a weight quantization technique to compress the deep neural network. They make an effort to minimize the quantity of weights that must be kept in memory. By doing this, similar weights are taken out of the equation, and the remaining weight is used to create many connections [6]. A quantization strategy that uses integer arithmetic for the inference was proposed by. Compared to floating-point operations, integer arithmetic is more efficient and requires fewer bits for representation.

ii. Activation Quantization: The network training process can be accelerated even more by using quantized activations to replace inner products with binary operations. By avoiding full precision activations, we may also lower the amount of memory required [7]. The activations were quantized to 8 bits. After training the network, they quantized the activations using a sigmoid function that restricts activations to the range [0, 1]. To quantify activation and weights, the authors proposed wide reduced-precision networks (WRPN). They discovered that activations take considerably more memory space than weights do. To counteract the accuracy loss brought on by quantization, they used an approach that involved increasing the number of filters in each layer [7].

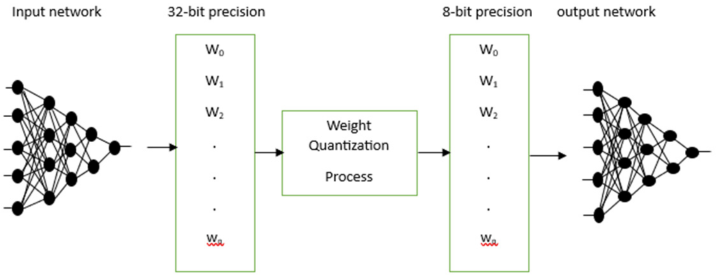


Figure 1. The concept of weight quantization.

In this study, we report the performances of two deep learning models trained for classifying weeds in corn and soybean: the large base model (trained using the InceptionV3 pre-trained model) and the lightweight model (trained using quantization-aware training).

The remainder of the paper is divided into three parts. The next section will discuss the “Methods” which introduces the dataset used, explains the deep learning techniques used, and explains the different evaluation metrics for assessing results. The results and findings are then reported and discussed in the “Results and Discussion” section. The conclusion is reported in the “Conclusion” section followed by the last section on “Acknowledgement”.

2. Methods

2.1. Dataset

In this work, two publicly available datasets were used: the “Soybean weed dataset” and the “Corn weed dataset”. Acquired soybean and weed images using a “Sony EXMOR” RGB camera mounted on an Unmanned Aerial Vehicle. The images of size 4000 x 3000 were then segmented [8]. The Corn weed dataset was taken from a corn field in China. The size of the images is 800 x 600 [9]. The combined dataset used consists of 7404 images of soybean, 4573 images of weed, and 1200 images of corn. In total, there are 13177 images. Here is a sample of the images in the dataset:

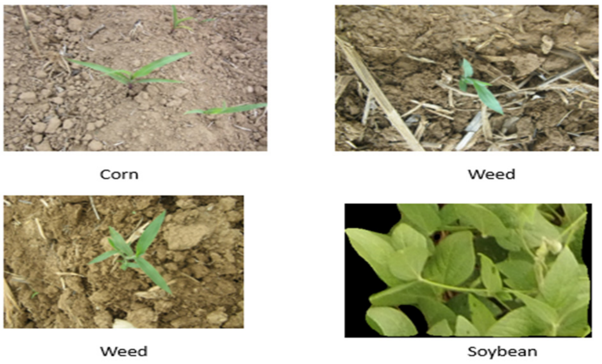
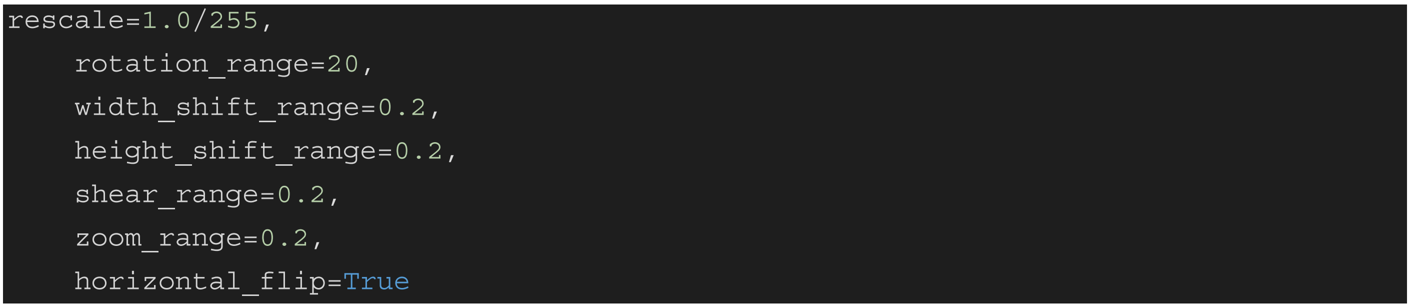


Figure 2. Sample data.

2.2. *Data Augmentation*

Due to dataset imbalance, we carried out data augmentation on the data before splitting it into training, validation, and test data. In the dataset, the corn class with a total of 1200 samples is less represented. The augmentation parameters and values used are:



```
rescale=1.0/255,
    rotation_range=20,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True
```

2.3. *Image classification*

In this study, we used the Keras deep learning framework with the TensorFlow backend to train the image classification model. We used transfer learning to train an InceptionV3 model. InceptionV3 uses a deeper network with fewer training parameters (23,851,784). The model consists of symmetric and asymmetric building blocks with convolutions, average pooling, max pooling, concatenations, dropouts, and fully connected layers. After training the InceptionV3 model, we then applied the technique of quantization to reduce the size of the model as depicted in the diagram below.

2.4. *Experimental setup*

This section describes the environment in which the experiment was carried out. The models were trained using Keras framework with the Tensorflow backend. We used the Google Collaboratory Environment with T4 GPU to do the training.

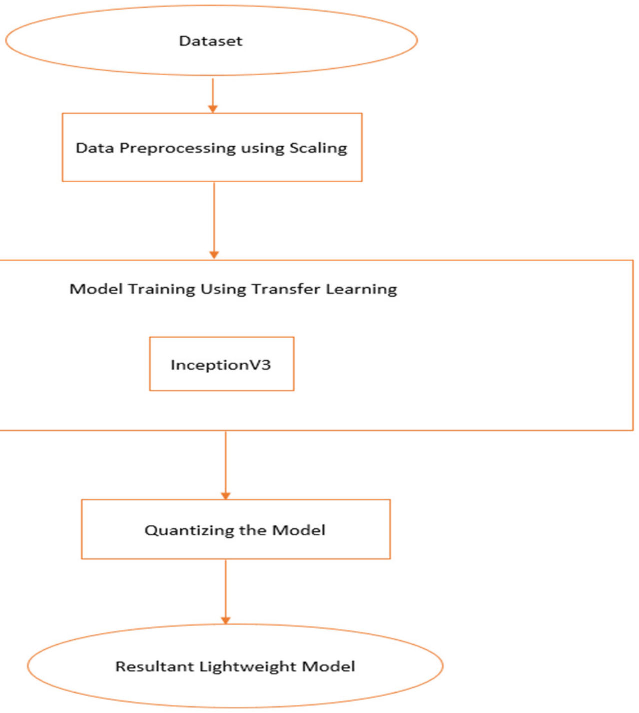

Dataset
Data Preprocessing using Scaling
Model Training Using Transfer Learning
InceptionV3
Quantizing the Model
Resultant Lightweight Model

Figure 3. Methodology development flow.

Results and Discussion

After experimenting in this study, results show that the InceptionV3 model achieved an accuracy of 97% with a size of 183.34MB while the quantized model attained an accuracy of 87% with a size of 23.38MB. The performances of the two models were also measured using other evaluation metrics and values were recorded accordingly.

The evaluation metrics were calculated thus:

Table 1. Table of values for metrics used to evaluate the models.

	Accuracy (%)	Precision (%)	F1-Score (%)	Recall (%)	AUC (%)	Model Size (MB)
Base Model	97	98	98	97	99	183
Quantized Model	87	91	90	87	98	23

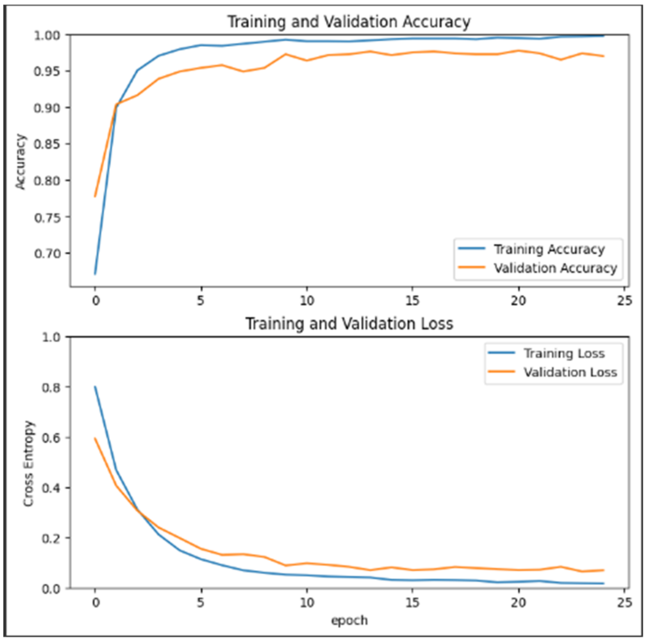


Figure 4. Accuracy and Loss for Base Model.

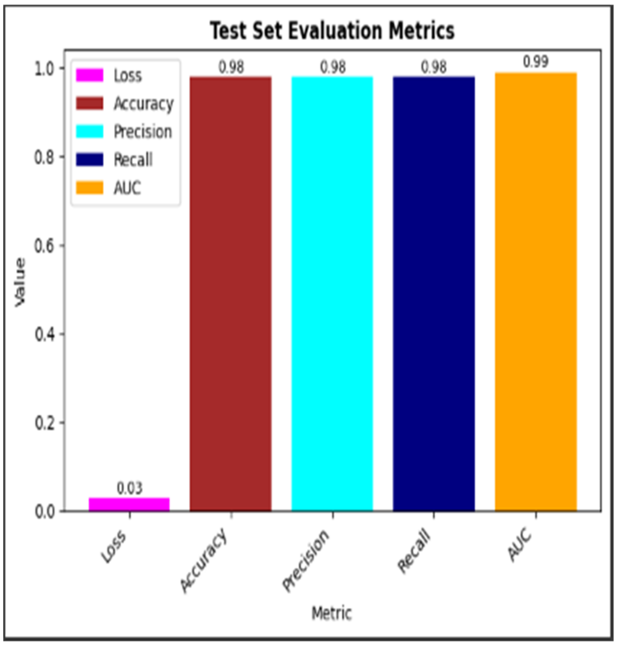


Figure 5. Evaluation metrics for Base Model.

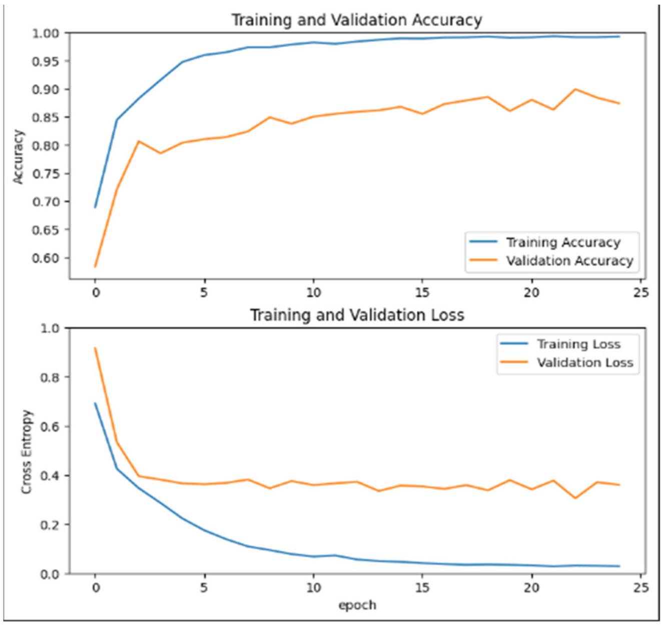


Figure 6. Accuracy and Loss for Quantized Model.

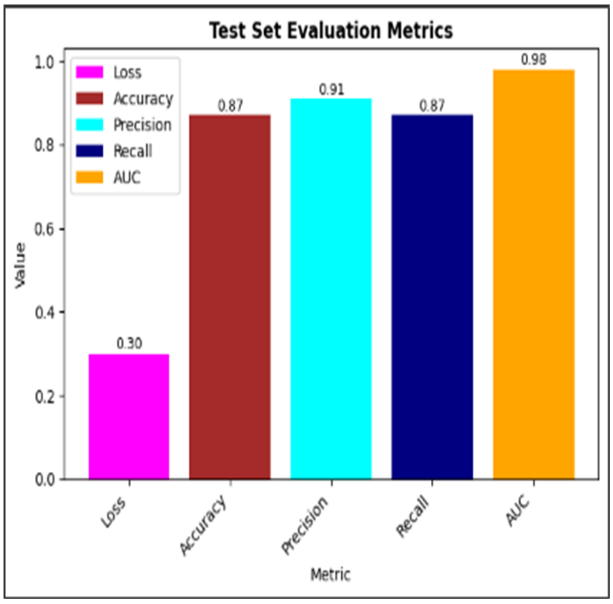


Figure 7. Evaluation metrics for Quantized Model**.**

From the results of the experiment, it can be seen that the base model performs better than the lightweight model in terms of accuracy, precision, f1-score, recall, and AUC, but the lightweight model is better in terms of size, and so can be deployed on low-resource devices.

4. Conclusions

From the result of the experiment, we show that quantization technique can be used to compress the size of a large deep learning model without loosing a significant amount of its performance. Also, the compressed model can be installed on low-resource devices for detecting weeds in corn and soybean.

References

1. K. Tan and D. Wang, "Towards Model Compression for Deep Learning Based Speech Enhancement," *IEEE/ACM Trans Audio Speech Lang Process*, vol. 29, 2021, doi: 10.1109/TASLP.2021.3082282.

2. N. Lee and C. Thierfelder, "Weed control under conservation agriculture in dryland smallholder farming systems of southern Africa. A review," *Agronomy for Sustainable Development*, vol. 37, no. 5. 2017. doi: 10.1007/s13593-017-0453-7.
3. SYAMSUL BAHRI, "No IMPROVING STUDENTS' VOCABULARY ACHIEVEMENT THROUGH THE WORD MEMORIZATION METHOD USING HANDBOOK IN MAN 2 POSO," *Front Neurosci*, vol. 14, no. 1, 2021.
4. N. Dewi and F. Ismawan, "IMPLEMENTASI DEEP LEARNING MENGGUNAKAN CNN UNTUK SISTEM PENGENALAN WAJAH," *Faktor Exacta*, vol. 14, no. 1, 2021, doi: 10.30998/faktorexacta.v14i1.8989.
5. S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding," in *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*, 2016.
6. S. Anwar, K. Hwang, and W. Sung, "Fixed point optimization of deep convolutional neural networks for object recognition," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2015. doi: 10.1109/ICASSP.2015.7178146.
7. B. Jacob *et al.*, "Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018. doi: 10.1109/CVPR.2018.00286.
8. S. M. Mahmudul Hasan, F. Sohel, D. Diepeveen, H. Laga, and M. G. K. Jones, "Weed recognition using deep learning techniques on class-imbalanced imagery," *Crop Pasture Sci*, 2022, doi: 10.1071/CP21626.
9. J. Virmani, V. Kumar, N. Kalra, and N. Khandelwal, "PCA-SVM based CAD system for focal liver lesions using B-mode ultrasound images," *Def Sci J*, vol. 63, no. 5, 2013, doi: 10.14429/dsj.63.3951.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.