A Deep Learning Model Compression and Ensemble Approach for Weed Detection

Martinson Ofori
Dakota State University
martinson.ofori@trojans.dsu.edu

Omar El-Gayar
Dakota State University
omar.el-gayar@dsu.edu

Austin O'Brien
Dakota State University
austin.obrien@dsu.edu

Cherie Noteboom
Dakota State University
cherie.noteboom@dsu.edu

Abstract

Site-specific weed management is an important practice in precision agriculture. Current advances in artificial intelligence have resulted in the use of large deep convolutional neural networks for weed detection. In this paper, a transfer learning, model compression, and ensemble learning approach is introduced that is suitable for resource-limited hardware such as mobile and embedded devices. The resulting ensemble model achieves 91.2% classification accuracy which is comparable to the performance of state-of-the-art deep learning models (such as the vanilla VGG16, DenseNet, and ResNet) while being about 62.22% smaller in size than DenseNet (the smallest-sized full-sized model). The approach used in this study is beneficial for further development of deep convolutional neural networks on smaller resource-limited hardware typically used in agriculture, as well as other industries such as healthcare and telecommunication.

1. Introduction

Since its introduction in the 1980s, precision agriculture – defined as a practice that manages the spatial and temporal variability associated with agricultural soil, crops, and livestock for improved performance and sustainability with the aid of agricultural information technologies and smart farm technologies [1]–[4], and Green IS emphasizing the use of information systems to achieve environmental objectives [5] – has made significant progress towards improving the sustainability of agriculture [6]. In the last few decades, precision farming has made substantial advancements to cropping systems. Using methods such as site-specific weed management, the practice can reduce the environmental impact of weed management through precise weed treatments that follow a four-step cyclical process consisting of 1) weed monitoring or detection, 2) management planning for action on weeding, 3) execution of the weed control method and 4) evaluation of performance [7].

The recent resurgence of artificial intelligence (AI) in the form of Deep Learning (DL) has resulted in phenomenal results in various problem domains. DL techniques known as *deep* Convolutional Neural Networks (DCNN) [8] have been successful because they learn to distinguish complex inherent patterns within images, often difficult to observe otherwise. The success of the AlexNet in the ImageNet Large Scale Visual Recognition Challenge 2012 – achieving a top-5 test error rate of 15.3% as compared to 26.2% achieved by the second-best entry [9] –has resulted in a substantial increase in the body of research that employs DCNNs across several disciplines and industries. For site-specific weed management, past research [10]–[15] has successfully employed DCNNs to distinguish various crops in different growth stages using different DCNN models and methods. Consequently, the use of DCNNs could provide increased benefits to the practice of precision agriculture and site-specific weed management.

However, DCNNs are known for their high computational and energy demands due to their complexity. This could be a significant barrier to the commercial adoption of DCNNs for weed management; the type of weed control systems used by these practices are often resource-constrained [16]. Increasingly powerful hardware systems are being developed to aid DCNN implementation but contribute to the cost of their commercial acceptance and use. Hence, the cost associated with such hardware could be a barrier to their adoption [17], [18]. This could have profound implications for practice. Although the acceptance of technology in farming has been promising, precision agriculture, on the whole, suffers from a slow adoption rate [19]. Failure to adopt agricultural technology has been attributed to concerns about complexity and high investment costs [2]. Especially for most rural dwellers and small-scale farmers, the cost of buying and servicing both hardware and software can be a

URI: https://hdl.handle.net/10125/79470
978-0-9981331-5-7

significant challenge that leads to non-adoption [20], [21]. Sustainable technology adoption should not affect farm profitability and efficiency [22], [23]. As such, there is a persistent need to pursue definitive ways to maintain or lower costs associated with maintaining or replacing current systems with new technology.

Consequently, this study proposes an approach to reducing the complexity of DCNN models for increased efficiency in ground-based plant classification systems. To demonstrate the effectiveness, we have implemented this novel method using publicly available deep learning libraries and evaluated the proposed method using a plant seedling dataset. From a theoretical perspective, the research demonstrates the potential of leveraging transfer learning, model compression, and ensemble learning to reduce the complexity (and thus the resource demands) of the resultant model while still maintaining classification performance that is comparable to full-size models. By reducing model complexity, the proposed method can also have implications for practice as it decreases the demands for computational resources and supporting technology infrastructure, thus contributing to the improved likelihood of adoption in resources-constrained environments such as precision agriculture.

2. Background and Related Work

Despite the successes that demonstrate the potential of DL for site-specific weed management, the producers of precision farming equipment have left DCNN systems relatively underutilized. A criticism of DL and DCNN models is primarily their complexity resulting in the constant need for computing power which requires them to be run on high-end computers or graphical processing units – CPUs and GPUs [17]. An added disadvantage to this high computing power requirement is that it results in high power consumption to make predictions – which is ineffective for sustainable farming [17], [24]. Various literature reviews on the subject of agricultural information technology adoption for PA [23], [25], [26] have demonstrated that farmers are often concerned with their bottom-line, which makes the cost of technology a key issue when developing equipment. Lowenberg-DeBoer et al. [27], in their analysis of the economics of robots and automation, found that although switching from conventional mechanization to automated systems could have positive ripple effects on the whole farm, such a shift in on-farm mechanics will only gain traction if new systems can prove their cost-effectiveness. Similarly, Ofori and El-Gayar [28], in their survey of social media posts, found that reducing the cost and complexity of agricultural information technologies could result in the uptake of technology and the adoption of precision agriculture. Hence, for commercial farm equipment producers (and ultimately farmers) to accept and adopt DL systems for precision agriculture, research that introduces less complex models to reduce the demand for computing resources is required.

Model compression solves this problem by compacting models by about 35-50x the size of the base model [29]. Model compression involves network pruning, quantization, and Huffman coding. Model pruning, which goes back to the 1990s [30], refers to the biologically inspired algorithms that emphasize further changes to existing models to retain only the bare minimum information needed to achieve comparative accuracy to their base model [31]–[33]. Pruning aims to reduce DCNN models by eliminating the redundancy and number of operations required for prediction. Further, quantization and weight sharing compress the pruned network by reducing the number of bits required to represent each weight, and Huffman coding ensures additional data compression.

Compressing a DCNN model leads to a decrease in the number and complexity computations, as well as the number of memory accesses for inference (the processing time for making a prediction) [34]–[36]. The compressed model is more energy- and resource-efficient due to its smaller size and faster inference speed [29]. Successfully fitting a compressed model on an embedded or mobile device and performing inference at the edge (without a need to transmit data to an intermediary server) has some additional advantages. For example, in most embedded systems where compressed models have been implemented, training is performed offline; and only inference is run on the embedded device. In this case, the compressed model preserves user privacy and reduces transmission cost [34], [35]. Further, the use of offline training (training once and deploying to several devices) reduces the resource requirement of the model as compared to continuous training [37].

As demonstrated in Figure 1 from the work of Han et al. [29], all three compression techniques under the right conditions retain the prediction accuracy of the original model. Regardless, some studies have found that the pruning ratio affects the accuracy of the model [38], [39]. In effect, a slight reduction in accuracy is possible depending on the percentage of the model's trainable weights that are pruned [38]–[40].

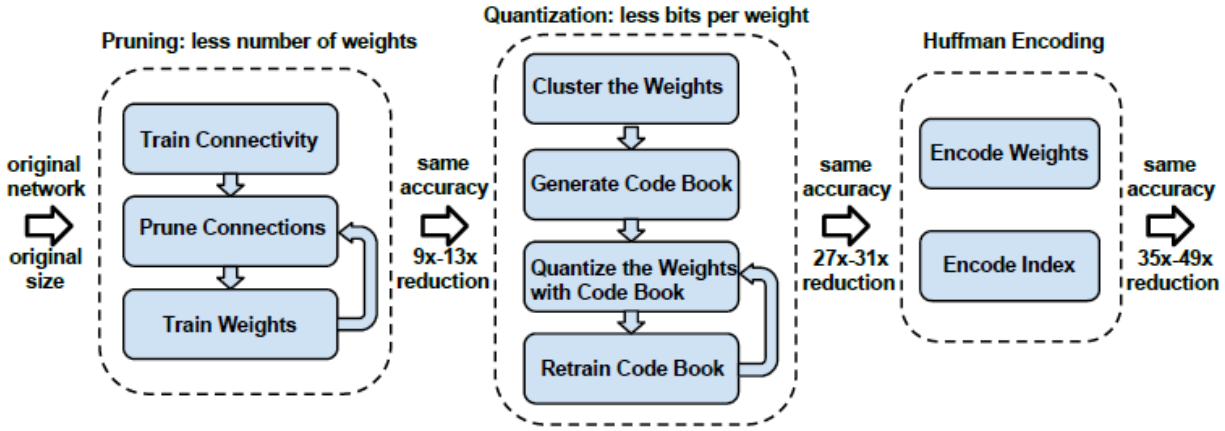

Figure 1. The three-stage model compression pipeline: pruning, quantization, and Huffman coding [29]

The current study makes the following contributions: theoretically, the study presents an approach that combines well-known DL techniques to reduce model complexity such that they require less expensive equipment to run without the performance degradation demonstrated in past studies; and practically, the proposed approach which solves some of the issues with DL at the edge could be employed in different contexts other than precision agriculture.

3. The Approach

The current study presents an approach for reducing DL model complexity for resource-constrained environments. The ensuing section discusses the proposed approach further and contains additional details on how these techniques were employed.

3.1. Model Architectures

As demonstrated in Figure 2, the models used in this research follow their architectural properties as well as performance in earlier research [41], [42]. A summary discussion on these models are presented below:

Spatial Exploitation Based. These kinds of networks take advantage of spatial filters to improve the performance of the network. The ***VGG,*** a popular DCNN network that replaced previous large filters with a smaller set of 3x3 filters and pushing depth to 16 and 19 layers, will be used [8]. The VGG won second place in the ImageNet Challenge 2014 classification track.

Depth and Multi-Path Based. The ***ResNet*** won the ImageNet 2015 challenge in image classification, detection, and localization, as well as the Winner of MS COCO 2015 detection, and segmentation uses both depth and multiple connections [43]. It is a very deep network that learns the residual representation functions instead of learning the signal representations directly.

Multi-Path Based. To reduce the problem of performance degradation, gradient vanishing, or explosion problems, these networks connect one layer to another by skipping some intermediate layers while still allowing the flow of information across the layers through multiple paths or shortcut connections. The ***DenseNet*** connects each layer to every other layer in a feed-forward fashion such that feature maps of all preceding layers are used as input to subsequent ones [44].

3.2. Transfer Learning

DCNN models often require several samples of training data to perform well on a classification task. In effect, deep models rely on a linearly related amount of data. Due to the dearth of high-quality labeled data containing several samples of plant seedlings, transfer learning is employed as the first stage of training. During transfer learning, a base model was trained on the dataset by freezing the first several layers of the base model (consisting of generic features), and then re-trained the remaining layers with randomly initialized weights using the target dataset (to acquire the target-specific features) [45]. In this case, about a third of each model was frozen. Intuitively, this works because DL models have *generic* features near the input while the domain-*specific* features lie much deeper in the model [45]. This step serves to establish a benchmark for the expected performance of state-of-the-art models on this dataset.

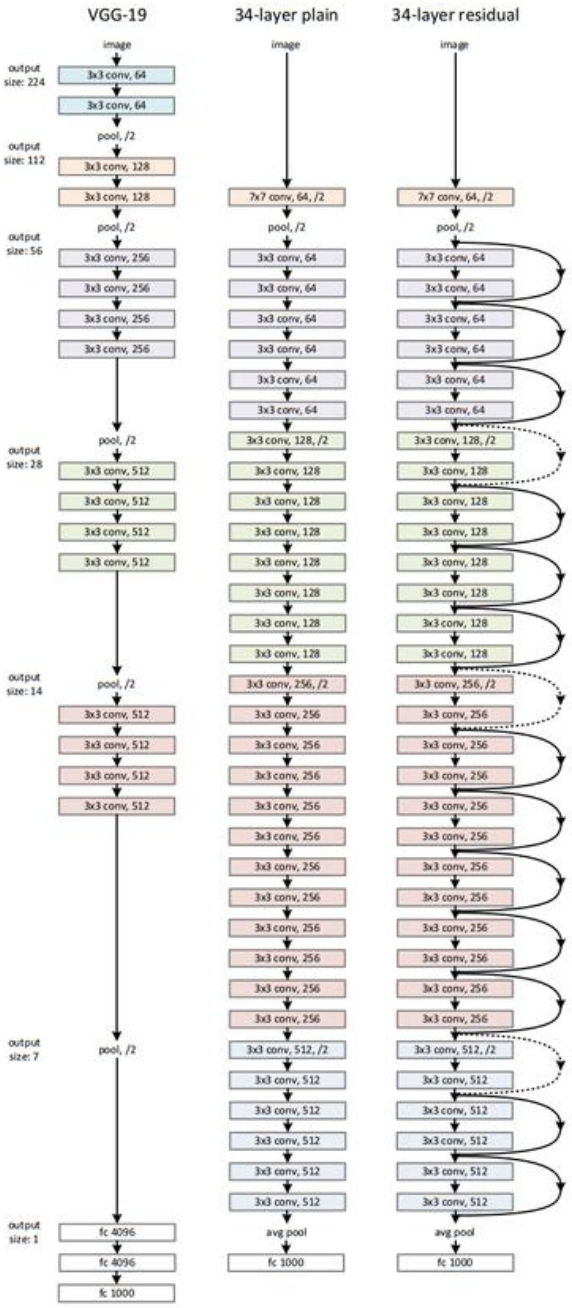

Figure 2. Comparing the classic VGG architecture (right) to a residual network (left) [43].

3.3. Model Compression

Even though DL techniques such as TL often decrease training time and/or increase classification accuracy, DCNN models are known to be overparametrized; hence, require significant computational resources. Due to the resource limitation of most precision agriculture systems, this study followed the work of Zhu and Ghupta [33] to prune the model iteratively. This involved adding a binary mask variable, the same size, and shape as the layer's weight tensor, to determine which weights participated in the model training. This process was used to mask out unnecessary weights. In this study, model training was started at 50% sparsity with a target of 80% sparsity by the end of training. After pruning, the model weights, represented as a sparse row, were easier to compress. Following this, post-training quantization and Huffman's encoding to reduce CPU and hardware accelerator latency, processing, power, and model size were performed [46]. This was done by reducing the number of bits needed to store each weight and compressing the resulting model in a lossless format.

3.4. Model Ensemble

Model compression is known to result in some loss of accuracy in the model. This study posits that model ensemble could be a useful technique to obtain improved results over the single compressed models for predictions. Although several types of model ensembles exist, such as simple voting or equally averaging the model predictions, the current study used a weighted average of the model predictions based on their performance. The optimal weights of the models were obtained through the direct optimization process known as differential evolution [47] which finds the set of weights that deliver the highest performance gains.

4. Methods

4.1. Dataset

Giselsson et al. [48] introduced the public image database for benchmarking plant seedling classification aimed at ground-based weeds or species spotting (https://vision.eng.au.dk/plant-seedlings-dataset/). The dataset is intended for researchers to perform object analysis, species recognition, or plant appearance analysis without the difficult and costly task of image acquisition, segmentation, and annotation. It consists of 5,539 images of approximately 960 unique plants belonging to 12 species at several growth stages. The plants were grown indoors in Styrofoam boxes and images were captured over 20 days. As overlapping plant leaves are minimal at the onset of plant growth, where most weed control such as broadcast spraying is undertaken, the images were captured in non-overlapping mode. Also, to avoid errors that may occur in pixel-based segmentation algorithms, plants were grown in soil that is covered in small stones. Figure 3 demonstrates images from the dataset.

4.2. Data Preparation

The following preprocessing techniques were applied:

Image resizing. All images were resized to 200x200 pixels to ensure the same aspect ratio.

Normalization of pixel values. This was done to ensure that all the pixels had similar data distribution. Pixel normalization aids the convergence of neural networks.

Data augmentation. Since plants do not grow in a single orientation and images could be captured from different angles, image augmentation was performed using horizontal and vertical flips, random rotations of up to 45 degrees, and zooms of up to 10 percent of the original image height and width.

4.3. Evaluation

The dataset was divided into two sets: 90% for training and 10% for tests. During training, a k-fold cross-validation approach was used where the training dataset *D*, was randomly divided into *k* number of mutually exclusive folds (subsets): S_1, S_2, $S_{3,\ldots}$, S_k. The model was trained *k* number of times where *k-1* subsets are used in training and each *k* was used as a validation set iteratively. In this study, *k=5* representing 5-fold cross-validation over 5 repetitions was performed. Model accuracy and size were then evaluated.

4.4. Technical Implementation

The experimentation carried out in this study was conducted using the Python programming language and libraries such as the TensorFlow and the Keras high-level API [49], [50].

The development environment was set up on the Google Colab Pro cloud, which assigns virtual machines equipped with either a Tesla T4 (5.5 Teraflops Single-Precision Performance and 8GB GPU Memory) or P100 GPU (4.7 Teraflops Double-Precision Performance and 16GB GPU Memory) for model training. Both GPUs employ an NVIDIA Pascal Architecture. Models were trained for 20 epochs with mini-batch sizes of 32 image instances. The initial learning rate was set at 0.0001 and decreased by a factor of 0.5 after every 3 epochs where the validation accuracy did not improve.

Figure 3. The plant seedling dataset.

5. Results

This section reports the results of the experiments conducted to compress DCNN models while maintaining accuracy on the plant seedling dataset through a novel combination of transfer learning; model compression; and weighted average model ensemble. Table 1 below summarizes the results.

The results presented in Table1 depict the performance of the DCNN models in 3 stages: a) transfer learning with an approximate third of the network frozen, b) the iteratively pruned network, and c) an ensemble of the pruned networks with weights applied through optimization.

5.1. Model Accuracy

In the first stage where the vanilla versions of the state-of-the-art models were fine-tuned by training with TL, the models delivered a consistent performance baseline that averaged 91.5%±0.1. The DenseNet

delivered the best performance with an average of 92.53%±0.1, followed by the VGG at 91.84%±0.1, and then the ResNet at 90.31%±0.4. This result was comparable to earlier research on this dataset albeit without the use of data augmentation [42].

Pruning the models over 20 epochs resulted in about a 6% average degradation of the prediction accuracies at 85.1%±0.08. In a similar fashion to the full-sized models, the best results were realized by the DenseNet at 86.02%±0.1, then the VGG with 84.71%±0.2, and last the ResNet at 84.52%±0.3.

When the models were then ensembled for prediction, using a simple average where each model contributed equal weights to the prediction result improved accuracy by a factor of 5% to 90.1%±0.2. This result was further improved by using an optimization process to find the best combination of model weights. Thus, increasing the prediction accuracy to 91.2%±0.2, a 6% increase as compared to the average result of the compressed models without ensemble. Overall, the ensemble method resulted in a better performance than the full-sized ResNet model and a slightly lower result compared to the other full-sized models: VGG (-0.6%) and DenseNet (1.3%).

5.2. Model Compression

The raw model sizes of the vanilla models trained on the plant seedlings were recorded at 371.07MB for the ResNet, 105.09MB for the VGG, and 45.48MB for the DenseNet.

By iteratively pruning the model weights during training, the models were reduced to an average of 20%±4.75 of the original sizes – ResNet decreased to 67.65MB, VGG16 to 16.76MB, and the DenseNet to 11.4MB. Further, post-training quantization and compression resulted in models that were on average 5%±1.62 of the original model sizes.

Thus, an ensemble of all three models will still reflect as a simple sum of the three models at 95.81MB when pruned and 28.97MB after compression, which is about 62.22% lower than the size of the DenseNet (the smallest-sized model).

Table 1. Results of experiments

Model	Accuracy	Size*	Compressed Accuracy	Size*: P[a]	Size*: P+Q+H[a]
VGG16	0.918±0.01	105.0908	0.847±0.02	16.7565	4.8554
DenseNet121	0.925±0.01	45.4808	0.860±0.01	11.4070	3.5403
ResNet152V2	0.903±0.04	371.0736	0.845±0.03	67.6545	20.5780
Simple Average of Compressed Models			0.901±0.02	95.8180^	28.9737^
Weighted Average of Compressed Models[b]			0.912±0.02		

* Size in megabytes
^ Calculated as sum of the model sizes
[a] P = Pruning; Q = Quantization, H = Huffman's encoding
[b] Weights applied by optimization process = VGG16: 0.386; DenseNet: 0.358; ResNet: 0.256

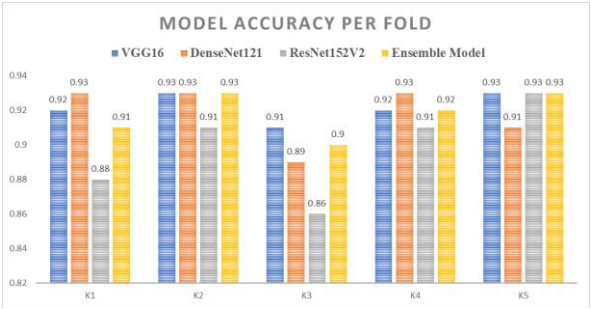

Figure 4. Model accuracies per each fold.

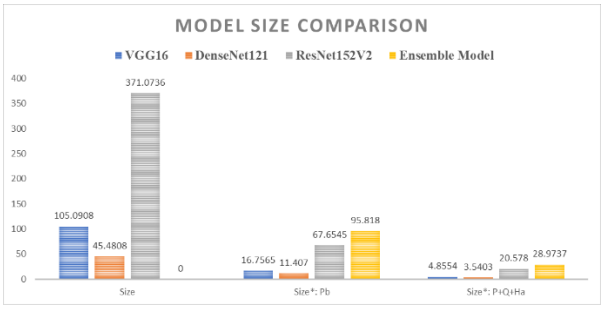

Figure 5. Model sizes comparison before and after pruning and compression

6. Discussion

The race for better chemical agents with higher biodegradability and lower environmental persistence continues unabated. This study complements prior research and ensures further digital transformation that leverages new technology capabilities to ensure sustainable development. For instance, computer vision equipment used for weed and pest management should be able to capture images and distinguish between food crops and weeds quickly and efficiently, especially at the onset of plant growth, where lax weed control could result in up to 100% yield loss. As such, and given the success of DCNNs, ensuring their applicability to farming scenarios will represent a huge milestone for Precision Agriculture and Green IS. However, the drawback of DL and other machine learning tasks is in their requirement for huge amounts of data for training which has a direct impact on both energy consumption and computing power of the infrastructure involved.

In this study, several existing DL techniques are combined in an approach that can ensure sustainability in the face of resource-constrained precision farming hardware. Each technique – transfer learning; model compression; and ensemble learning – delivers benefits that can enhance and underpin the generalizability of DL to precision agriculture systems. Transfer learning, where a model trained on one task can be ported to another task, offers opportunities for reducing overfitting and ensuring robust results in the face of limited training data [51]. Model compression offers additional benefits for reducing the size of the DL models, which means an equivalent reduction in both energy consumption and inference time [33]. Last, ensemble learning improves classification performance by combining several architecturally different models into a single prediction. As demonstrated in the current study, the proposed approach resulted in a considerable reduction in the model sizes while keeping prediction accuracies comparable to the full-sized state-of-the-art models.

In the first stage of the proposed approach, state-of-the-art models with pre-trained weights were trained on the plant seedling dataset. The result of this baseline (average 91.5% prediction accuracy) was comparable to the earlier result of prior studies that employed the same dataset [42], [51]–[53]. Further, the literature points to a relationship between model accuracy and model compression such that pruning models could result in decreased accuracy [38], [39]. This was seen in the current study as compressing the models to 80% sparsity resulted in about a 6% decrease in the accuracy of predictions. Fountsop et al. [38] demonstrated similar results in their study where the highest accuracy achieved by a VGG16 model (trained over 100 epochs on the plant seedling dataset at 90% pruning ratio and post-training quantization applied) was 89.84%. An ensemble approach using weighted model averages was introduced in the current study to resolve this drop in prediction accuracy. This approach which combined a hybrid ensemble DL technique with model compression to compensate for the performance degradation resulted in increased performance (average 91.2% prediction accuracy) comparable to state-of-the-art DCNN models at a fraction of the size.

In summary, as demonstrated by past research [33]–[35], [54], model compression reduces the complexity and resource demands of the DCNN models to allow for faster real-time inference. The approach presented in this study reduced the complexity of DCNN models and presented benchmarks to demonstrate 1) the reduction in model size by pruning out unused weights and 2) accuracy retention through ensemble learning. This approach will be beneficial to ground-based weed detection systems and contribute to minimizing the environmental footprint of agricultural technology while maximizing production efficiency. Although the context for the research is limited to use cases in precision agriculture and green information systems, this proposed method could be applied to similar computer vision tasks in resource-constrained environments commonly encountered in other industries such as healthcare and telecommunication.

7. Conclusion and Future Research

This study proposes a DCNN approach for plant seedling classification and weed detection using a set of techniques for reducing the hardware requirements of resource-constrained systems while keeping accuracy at par with full-sized state-of-the-art DCNNs. The approach employed three stages to devise a sparse network: transfer learning, model compression via pruning, quantization, and Huffman's encoding; and weighted average model ensemble to determine the appropriate combination of model weights that deliver the best accuracy.

Transfer learning over 20 epochs using three state-of-the-art models – VGG, DenseNet, and ResNet – demonstrated a performance baseline of 91.5%, with the smallest model size being the DenseNet at 45.48MB. Although model compression resulted in models that were up to 5% of the original sizes, this also resulted in a 6% loss in accuracy over the same training regime. Thus, model ensemble using an optimization technique to find the best weighted

average combination was introduced to counteract this effect. The ensemble approach achieved an average accuracy of 91.2%.

Theoretically, the approach proposed by the current study demonstrates that combining transfer learning and ensemble learning can resolve the performance degradation associated with model compression. Practically, this approach could be beneficial for further development of DCNNs for inference on the edge in agriculture, as well as other industries such as healthcare and telecommunication.

The limitations of the study that warrant further analysis include additional investigation with other datasets, training over longer time periods using different optimizers for the DCNNs, and exploring other ensemble approaches such as model stacking. The approach presented in the current study is meant to steer the conversation from the drawbacks of DL (such as the need for large amounts of data, longer training times, and expensive computers) to inference implemented directly on cheaper embedded and mobile devices. The lifetime cost/energy savings of employing this approach has not been measured and could warrant additional investigation.

8. References

[1] A. T. Balafoutis *et al.*, "Smart Farming Technologies – Description, Taxonomy and Economic Impact," in *Precision Agriculture: Technology and Economic Perspectives*, S. M. Pedersen and K. M. Lind, Eds. Cham: Springer International Publishing, 2017, pp. 21–77. doi: 10.1007/978-3-319-68715-5_2.

[2] M. Kernecker, A. Knierim, A. Wurbs, T. Kraus, and F. Borges, "Experience versus expectation: farmers' perceptions of smart farming technologies for cropping systems across Europe," *Precis. Agric.*, vol. 21, pp. 34–50, 2020, doi: 10.1007/s11119-019-09651-z.

[3] Y. Wang, L. Jin, and H. Mao, "Farmer Cooperatives' Intention to Adopt Agricultural Information Technology—Mediating Effects of Attitude," *Inf. Syst. Front.*, vol. 21, 2019, doi: 10.1007/s10796-019-09909-x.

[4] S. Wolfert, D. Goense, and C. A. G. Sorensen, "A Future Internet Collaboration Platform for Safe and Healthy Food from Farm to Fork," in *2014 Annual SRII Global Conference*, San Jose, CA, USA, Apr. 2014, pp. 266–273. doi: 10.1109/SRII.2014.47.

[5] J. Dedrick, "Green IS: Concepts and Issues for Information Systems Research," *Commun. Assoc. Inf. Syst.*, vol. 27, no. 1, Aug. 2010, doi: 10.17705/1CAIS.02711.

[6] P. C. Robert, "Precision agriculture: a challenge for crop nutrition management," in *Progress in Plant Nutrition: Plenary Lectures of the XIV International Plant Nutrition Colloquium: Food security and sustainability of agro-ecosystems through basic and applied research*, W. J. Horst, A. Bürkert, N. Claassen, H. Flessa, W. B. Frommer, H. Goldbach, W. Merbach, H.-W. Olfs, V. Römheld, B. Sattelmacher, U. Schmidhalter, M. K. Schenk, and N. v. Wirén, Eds. Dordrecht: Springer Netherlands, 2002, pp. 143–149. doi: 10.1007/978-94-017-2789-1_11.

[7] F. López-Granados, "Weed detection for site-specific weed management: Mapping and real-time approaches," *Weed Res.*, vol. 51, pp. 1–11, 2011, doi: 10.1111/j.1365-3180.2010.00829.x.

[8] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *ArXiv14091556 Cs*, Sep. 2014, Accessed: Sep. 11, 2019. [Online]. Available: http://arxiv.org/abs/1409.1556

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[10] M. Dyrmann, H. Karstoft, and H. S. Midtiby, "Plant species classification using deep convolutional neural network," *Biosyst. Eng.*, vol. 151, pp. 72–80, Nov. 2016, doi: 10.1016/j.biosystemseng.2016.08.024.

[11] A. Milioto, P. Lottes, and C. Stachniss, "Real-Time Blob-Wise Sugar Beets Vs Weeds Classification for Monitoring Fields Using Convolutional Neural Networks," *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.*, vol. IV-2/W3, pp. 41–48, Aug. 2017, doi: 10.5194/isprs-annals-IV-2-W3-41-2017.

[12] X. E. Pantazi, A. A. Tamouridou, T. K. Alexandridis, A. L. Lagopodi, J. Kashefi, and D. Moshou, "Evaluation of hierarchical self-organising maps for weed mapping using UAS multispectral imagery," *Comput. Electron. Agric.*, vol. 139, pp. 224–230, Jun. 2017, doi: 10.1016/j.compag.2017.05.026.

[13] R. A. Sørensen, J. Rasmussen, J. Nielsen, and R. N. Jørgensen, "Thistle detection using convolutional neural networks," *2017 Efita Wcca Congr. - Eur. Conf. Dedic. Future Use Ict*, pp. 161–162, 2017.

[14] W. Xinshao and C. Cheng, "Weed seeds classification based on PCANet deep learning baseline," in *2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, Hong Kong, Dec. 2015, pp. 408–415. doi: 10.1109/APSIPA.2015.7415304.

[15] B. A. M. Ashqar, B. S. Abu-Nasser, and S. S. Abu-Naser, "Plant Seedlings Classification Using Deep Learning," vol. 3, no. 1, p. 8, 2019.

[16] B. Steward, J. Gai, and L. Tang, "The use of agricultural robots in weed management and control," in *Burleigh Dodds Series in Agricultural Science*, J. Billingsley, Ed. Burleigh Dodds Science Publishing, 2019, pp. 161–186. doi: 10.19103/AS.2019.0056.13.

[17] L. Santos, F. N. Santos, P. M. Oliveira, and P. Shinde, "Deep Learning Applications in Agriculture: A Short Review," in *Robot 2019: Fourth Iberian Robotics Conference*, vol. 1092, M. F. Silva, J. Luís Lima, L. P.

Reis, A. Sanfeliu, and D. Tardioli, Eds. Cham: Springer International Publishing, 2020, pp. 139–151. doi: 10.1007/978-3-030-35990-4_12.

[18] A. Wang, W. Zhang, and X. Wei, "A review on weed detection using ground-based machine vision and image processing techniques," *Comput. Electron. Agric.*, vol. 158, pp. 226–240, Mar. 2019, doi: 10.1016/j.compag.2019.02.005.

[19] J. Lowenberg-DeBoer and B. Erickson, "Setting the Record Straight on Precision Agriculture Adoption," *Agron. J.*, vol. 111, no. 4, p. 1552, 2019, doi: 10.2134/agronj2018.12.0779.

[20] E. Misaki, M. Apiola, S. Gaiani, and M. Tedre, "Challenges facing sub-Saharan small-scale farmers in accessing farming information through mobile phones: A systematic literature review," *Electron. J. Inf. Syst. Dev. Ctries.*, vol. 84, no. 4, p. e12034, Jul. 2018, doi: 10.1002/isd2.12034.

[21] A. Saidu, A. M. Clarkson, S. H. Adamu, M. Mohammed, and I. Jibo, "Application of ICT in Agriculture: Opportunities and Challenges in Developing Countries," *Int. J. Comput. Sci. Math. Theory*, vol. 3, no. 1, p. 11, 2017.

[22] Y. S. Tey and M. Brindal, "Factors influencing the adoption of precision agricultural technologies: a review for policy implications," *Precis. Agric.*, vol. 13, no. 6, pp. 713–730, Dec. 2012, doi: 10.1007/s11119-012-9273-6.

[23] S. Wolfert, L. Ge, C. Verdouw, and M.-J. Bogaardt, "Big Data in Smart Farming – A review," *Agric. Syst.*, vol. 153, pp. 69–80, May 2017, doi: 10.1016/j.agsy.2017.01.023.

[24] J. Wang, Y. Ma, L. Zhang, R. X. Gao, and D. Wu, "Deep learning for smart manufacturing: Methods and applications," *J. Manuf. Syst.*, vol. 48, pp. 144–156, Jul. 2018, doi: 10.1016/j.jmsy.2018.01.003.

[25] J. Lowenberg-DeBoer and S. M. Swinton, "Economics of Site-Specific Management in Agronomic Crops," in *ASA, CSSA, and SSSA Books*, F. J. Pierce and E. J. Sadler, Eds. Madison, WI, USA: American Society of Agronomy, Crop Science Society of America, Soil Science Society of America, 1997, pp. 369–396. doi: 10.2134/1997.stateofsitespecific.c16.

[26] S. I. Moazzam, U. S. Khan, M. I. Tiwana, J. Iqbal, W. S. Qureshi, and S. I. Shah, "A Review of Application of Deep Learning for Weeds and Crops Classification in Agriculture," in *2019 International Conference on Robotics and Automation in Industry (ICRAI)*, Oct. 2019, pp. 1–6. doi: 10.1109/ICRAI47710.2019.8967350.

[27] J. Lowenberg-DeBoer, I. Y. Huang, V. Grigoriadis, and S. Blackmore, "Economics of robots and automation in field crop production," *Precis. Agric.*, May 2019, doi: 10.1007/s11119-019-09667-5.

[28] M. Ofori and O. El-Gayar, "Drivers and challenges of precision agriculture: a social media perspective," *Precis. Agric.*, Oct. 2020, doi: 10.1007/s11119-020-09760-0.

[29] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," p. 14, 2016.

[30] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.

[31] Y.-D. Kim, E. Park, S. Yoo, T. Choi, L. Yang, and D. Shin, "Compression of deep convolutional neural networks for fast and low power mobile applications," p. 16, 2016.

[32] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning Convolutional Neural Networks for Resource Efficient Inference," Nov. 2016, Accessed: Oct. 02, 2020. [Online]. Available: https://arxiv.org/abs/1611.06440v2

[33] M. Zhu and S. Gupta, "To prune, or not to prune: exploring the efficacy of pruning for model compression," Oct. 2017, Accessed: Oct. 02, 2020. [Online]. Available: https://arxiv.org/abs/1710.01878v2

[34] S. Voghoei, N. Hashemi Tonekaboni, J. G. Wallace, and H. R. Arabnia, "Deep Learning at the Edge," in *2018 International Conference on Computational Science and Computational Intelligence (CSCI)*, Las Vegas, NV, USA, Dec. 2018, pp. 895–901. doi: 10.1109/CSCI46756.2018.00177.

[35] V. S. Lalapura, J. Amudha, and H. S. Satheesh, "Recurrent Neural Networks for Edge Intelligence: A Survey," *ACM Comput. Surv.*, vol. 54, no. 4, pp. 1–38, Jul. 2021, doi: 10.1145/3448974.

[36] T. Choudhary, V. Mishra, A. Goswami, and J. Sarangapani, "A comprehensive survey on model compression and acceleration," *Artif. Intell. Rev.*, vol. 53, no. 7, pp. 5113–5155, Oct. 2020, doi: 10.1007/s10462-020-09816-7.

[37] Y. Chen, B. Zheng, Z. Zhang, Q. Wang, C. Shen, and Q. Zhang, "Deep Learning on Mobile and Embedded Devices: State-of-the-art, Challenges, and Future Directions," *ACM Comput. Surv.*, vol. 53, no. 4, pp. 1–37, Sep. 2020, doi: 10.1145/3398209.

[38] A. N. Fountsop, J. L. Ebongue Kedieng Fendji, and M. Atemkeng, "Deep Learning Models Compression for Agricultural Plants," *Appl. Sci.*, vol. 10, no. 19, p. 6866, Sep. 2020, doi: 10.3390/app10196866.

[39] S. Rajaraman, J. Siegelman, P. O. Alderson, L. S. Folio, L. R. Folio, and S. K. Antani, "Iteratively Pruned Deep Learning Ensembles for COVID-19 Detection in Chest X-Rays," *IEEE Access*, vol. 8, pp. 115041–115050, 2020, doi: 10.1109/ACCESS.2020.3003810.

[40] Tensorflow.org, "Pruning in Keras example | TensorFlow Model Optimization," 2021. https://www.tensorflow.org/model_optimization/guide/pruning/pruning_with_keras (accessed Jun. 01, 2021).

[41] A. Khan, A. Sohail, U. Zahoora, and A. Saeed, "A Survey of the Recent Architectures of Deep Convolutional Neural Networks," *Artif. Intell. Rev.*, 2019, doi: 10.1007/s10462-020-09825-6.

[42] M. Ofori and O. El-Gayar, "Towards Deep Learning for Weed Detection: Deep Convolutional Neural Network Architectures for Plant Seedling Classification," *AMCIS 2020 Proc.*, Aug. 2020, [Online]. Available: https://aisel.aisnet.org/amcis2020/sig_green/sig_green/3

[43] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” *ArXiv151203385 Cs*, Dec. 2015, Accessed: Feb. 14, 2020. [Online]. Available: http://arxiv.org/abs/1512.03385
[44] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, “Densely Connected Convolutional Networks,” *ArXiv160806993 Cs*, Jan. 2018, Accessed: Feb. 14, 2020. [Online]. Available: http://arxiv.org/abs/1608.06993
[45] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, “How transferable are features in deep neural networks?,” *ArXiv14111792 Cs*, Nov. 2014, Accessed: Oct. 25, 2019. [Online]. Available: http://arxiv.org/abs/1411.1792
[46] Tensorflow.org, “Post-training quantization | TensorFlow Model Optimization,” 2021. https://www.tensorflow.org/model_optimization/guide/quantization/post_training (accessed May 31, 2021).
[47] Scipy.org, “Differential Evolution — SciPy v1.6.3 Reference Guide,” 2021. https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.differential_evolution.html (accessed May 31, 2021).
[48] T. M. Giselsson, R. N. Jørgensen, P. K. Jensen, M. Dyrmann, and H. S. Midtiby, “A Public Image Database for Benchmark of Plant Seedling Classification Algorithms,” *ArXiv171105458 Cs*, Nov. 2017, Accessed: Sep. 05, 2019. [Online]. Available: http://arxiv.org/abs/1711.05458
[49] M. Abadi *et al.*, “TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems,” *ArXiv160304467 Cs*, Mar. 2016, Accessed: Oct. 25, 2019. [Online]. Available: http://arxiv.org/abs/1603.04467
[50] F. Chollet and others, *Keras*. 2015. [Online]. Available: https://keras.io
[51] M. Ofori and O. El-Gayar, “An Approach for Weed Detection Using CNNs And Transfer Learning,” in *Hawaii International Conference on System Sciences*, Jan. 2021, p. 10.
[52] N. R. Rahman, Md. A. M. Hasan, and J. Shin, “Performance Comparison of Different Convolutional Neural Network Architectures for Plant Seedling Classification,” in *2020 2nd International Conference on Advanced Information and Communication Technology (ICAICT)*, Nov. 2020, pp. 146–150. doi: 10.1109/ICAICT51780.2020.9333468.
[53] C. R. Alimboyong and A. A. Hernandez, “An Improved Deep Neural Network for Classification of Plant Seedling Images,” in *2019 IEEE 15th International Colloquium on Signal Processing Its Applications (CSPA)*, Mar. 2019, pp. 217–222. doi: 10.1109/CSPA.2019.8696009.
[54] L. Deng, G. Li, S. Han, L. Shi, and Y. Xie, “Model Compression and Hardware Acceleration for Neural Networks: A Comprehensive Survey,” *Proc. IEEE*, vol. 108, no. 4, pp. 485–532, Apr. 2020, doi: 10.1109/JPROC.2020.2976475.