

Convolutional neural network-based crop disease detection model using transfer learning approach

Segun Adebayo¹, Halleluyah Oluwatobi Aworinde², Akinwale O. Akinwunmi²,
Adebamiji Ayandiji³, Awoniran Olalekan Monsir²

¹Mechatronics Programme, College of Agricultural, Engineering and Science, Bowen University, Iwo, Nigeria

²Computer Science Programme, College of Computing and Communication Studies, Bowen University, Iwo, Nigeria

³Agriculture Programme, College of Agriculture, Engineering and Science, Bowen University, Iwo, Nigeria

Article Info

Article history:

Received Dec 6, 2021

Revised Sep 26, 2022

Accepted Oct 13, 2022

Keywords:

Convolutional neural network

Crop disease detection

Deep learning

Pattern recognition

Transfer learning

ABSTRACT

Crop diseases disrupt the crop's physiological constitution by affecting the crop's natural state. The physical recognition of the symptoms of the various diseases has largely been used to diagnose cassava infections. Every disease has a distinct set of symptoms that can be used to identify it. Early detection through physical identification, however, is quite difficult for a vast crop field. The use of electronic tools for illness identification then becomes necessary to promote early disease detection and control. Convolutional neural networks (CNN) were investigated in this study for the electronic identification and categorization of photographs of cassava leaves. For feature extraction and classification, the study used databases of cassava images and a deep convolutional neural network model. The methodology of this study retrained the models' current weights for visual geometry group (VGG-16), VGG-19, SqueezeNet, and MobileNet. Accuracy, loss, model complexity, and training time were all taken into consideration when evaluating how well the final layer of CNN models performed when trained on the new cassava image datasets.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Halleluyah Oluwatobi Aworinde

Computer Science Programme, College of Computing and Communication Studies, Bowen University

Iwo, Osun State, Nigeria

Email: aworinde.halleluyah@bowen.edu.ng

1. INTRODUCTION

Crop diseases affect the physiological makeup of the crop while impairing its natural state [1]. Plant infections that afflict the host plants and make them sick are to blame for these deficiencies. These diseases, which can harm any component of a plant above or below the earth, can be bacterial, viral, fungal, or parasitic nematodes [2]. These illnesses can alter the physical, chemical, and biological makeup of agricultural crops, which in turn alters how the affected plant component's function [3]. As a result, the changed physiology of the farm plants lowers the production of the crops [4]. Some of the elements that affect disease occurrences and their dissemination in farm crops include seasonal variations, environmental circumstances, the presence of specific pathogens, and crop variety features [5]-[7]. This makes it difficult to predict and identify potential disease attacks on agricultural crops. Cassava (*Manihot esculenta Crantz*) is an agricultural crop that is very susceptible to many different forms of illnesses [8], [9]. One of the most significant staple foods farmed in Africa is cassava, which is also an essential raw material for industries in Latin America and Asia [9]. Cassava is vegetatively propagated by stem cuttings, which has many benefits but also means that the crop's diseases can readily and quickly be spread from one generation of the crop to another, endangering the crop's ability to

be produced [9]. As a result, it is crucial to predict, identify, and effectively combat the different illnesses that affect agricultural crops.

Physically identifying the symptoms of the various diseases has proven to be the most effective way to diagnose cassava diseases [9]-[11]. Every disease has a distinct set of symptoms that can be used to identify it. Early detection through physical identification, however, is quite difficult for a vast crop field. The use of electronic tools for illness identification then becomes necessary to promote early disease detection and control. The convolutional neural network is a tried-and-true technique for the electronic identification and classification of images.

Convolutional neural networks (CNNs) are a class of artificial neural networks and a deep learning technique that are most frequently used for image analysis and categorization [12]. It has taken over as the technique of choice for computer vision tasks. CNN (or ConvNets) utilizes many building blocks known as fully linked convolution layers to automatically learn and adapt planar hierarchical properties of pictures using backpropagation. These layers also contain further improvements, such as the pooling layers [13]. The skills gained from using CNNs to solve one problem can be applied to other challenges. Transfer learning is what this is. Predictive modeling, which is comprised of various data patterns in another domain, is facilitated by transfer learning approaches, which aim to adapt the knowledge gained from data processing in fundamental domains [14]. Transfer learning has several benefits, including a tendency to improve neural network performance, time savings during network training, and a decrease in the amount of training data required [15]. Through the application of CNN transfer learning, the goal of this study is to enhance the early detection of cassava infections. Deep learning techniques, in particular CNNs, have made significant strides in the recognition and classification of images, and consequently, in the autonomous detection of plant diseases [16]. AlexNet, DenseNet, ResNet, visual geometry group (VGG), MobileNet, Inception, and Xception are examples of popular CNN frameworks that have been applied to image recognition, and more specifically to the diagnosis of diseases [17]. Deep convolutional neural networks were shown to be superior to Xception in circumstances when there was limited computational resources and the resolution of the images had to be lowered [18].

A mobile-based deep learning model for the diagnosis of cassava infections was built [19] as part of the application of CNN to the detection of agricultural diseases. The generated model and application did, however, record inconsistent performance between the real-world videos and photographs. Convolutional neural networks were also used to create a model for the detection and classification of cassava diseases from an unbalanced dataset [20]. But the highly unbalanced nature of the data caused the generated model to favor some classes over others. The data were balanced through oversampling; however, the model runs the danger of being overfit for the underrepresented class of data. Once more, a deep learning model for identifying plant diseases was created [21] to improve smart farming. Chan-Vese (CV) method was utilized for feature detection, and region proposal network (RPN) was used to get over the leaves' background as a barrier in the research. The model showed very good performance, but the feature extraction CV technique needs to execute repeating iterative computations, which takes time.

Additionally, crop disease detection has also been done via transfer learning. Support vector machines (SVM), k closest neighbors (KNN), and CNN architectures (inception) were utilized in the method to detect and classify cassava illnesses based on images [22]. All three machine learning methods used with Inception produced very high performance from the model, with the CNN architecture producing the highest results.

2. METHOD

The problem of crop disease detection using CNN can be represented as data pairs of type (x, y) , where the learning algorithm's input x is translated into a particular output y . The primary objective of the learning method is to train the function f given in equation 1 in a way that maps the space input V_x , a set of all possible input vectors of x , to the space target V_y , a set of all possible output vectors of y :

$$f : V_x \rightarrow V_y \quad (1)$$

however, the training process, which is stated as, attempts to approximate f with a function g over the data (x, y) as it does not have access to the entire set of V_x and V_y . It is given as (2):

$$g : x \rightarrow y \quad (2)$$

given a cost function L and a set of model parameters ψ , the goal of computing g is to minimize the in-sample error between the model's predictions, $g(x)$, and the actual values of the data, y :

$$\min_{\psi} L(\varepsilon_{in}) \quad (3)$$

the model final output is given as (4):

$$\hat{y} = g(x|\hat{\psi}) \quad (4)$$

where $\hat{\psi}$ is the learned parameter.

On cassava image datasets, transfer learning using a deep convolutional neural network (CNN) model was used as opposed to the more labor-intensive traditional strategy of training classifiers with manually constructed feature extraction. To classify photos, our method entails retraining the current weights of the AlexNet, SqueezeNet, MobileNet, VGG-16, and VGG-19 models. These models' final layer's training performance was assessed using the brand-new cassava image datasets. The learning rate, minimum batch size, epoch, and validation frequency are some of the model characteristics that were taken into consideration in this study.

2.1. Experimental setup

The experimental environment includes the following specifications: 12 GB of RAM, 512 GB of SSD storage, an Intel Core™ i5-7500U CPU, Windows 10 operating system, and the MATLAB R2020a application suite, the experiment environment was set up. A technical computing platform called MATLAB combines calculation, visualization, and programming in a user-friendly setting. Toolboxes on the platform are application-specific solutions. These toolboxes are collections of MATLAB functions (M-files) used to address various problem types. In this study, data input, preprocessing, training, optimization, and classification were all done using the neural network toolbox.

2.2. Dataset acquisition

Images for the experiment as shown in Figure 1 were taken from two sources; the first dataset came from the research farm at Bowen University. This dataset included 18,000 tagged photos of thirteen field excursions using a Nikon Digital SLR Camera D3200, representing several kinds of cassava leaf disease. The camera is a single lens reflex digital model with a 1.5x lens focal length effective angle of view and 24.7 million effective pixels. To maximize the data variation and ensure that the model developed is not limited in accuracy to a specific time of day, the photos were taken three times during the day: in the morning, afternoon, and evening. The second batch of data came from Kaggle [16], [17]. Five types of cassava leaf diseases are categorized in the dataset. 10,000 annotated photos in total were gathered during a routine survey in Uganda, the majority of which came from farmers [23].



Figure 1. Cassava dataset

2.3. Model architecture selection

The issue of choosing a model from a group of potential models is known as model selection. method refers to the process of choosing a statistical model from a list of potential models based on input data. The simplest model is most likely to be the best option when there are several candidate models with comparable explanatory or predictive power [24]. Identified two broad categories of techniques: probabilistic and resampling [25]. The former entails analytically rating a candidate model based on both its performance on the training dataset and the complexity of the model. The model's performance on data outside of the sample is estimated by the resampling method. Using the probabilistic method shown in Table 1, this paper selected a model. For training on the dataset, models with different levels of complexity from AlexNet were used. Each model's performance on the test dataset was compared to its level of complexity.

Table 1. Model selection parameters

Algorithm	Lower hyper parameter		Baseline	Higher hyper parameter	
	SqueezeNet	MobileNet	AlexNet	VGG-16	VGG-19
Complexity	1.24M	3.50M	61.0M	138M	144M
Layer	18	53	8	16	19
Size	4.60 MB	13 MB	227 MB	515 MB	535 MB

2.3.1. AlexNet

AlexNet as presented in Figure 2, has an eight-layer CNN architecture with three fully linked layers and five convolutional layers. An overlapping maximum pooling layer comes after each of the first two convolutional layers [26]. There is a direct connection between the third, fourth, and fifth convolution layers. Overlapping maximum pooling layer comes after the fifth convolutional layer, which is subsequently connected to fully connected layers. Each of the fully connected layers has 4,096 neurons, and a SoftMax classifier with 1,000 classes receives input from the second completely connected layer.

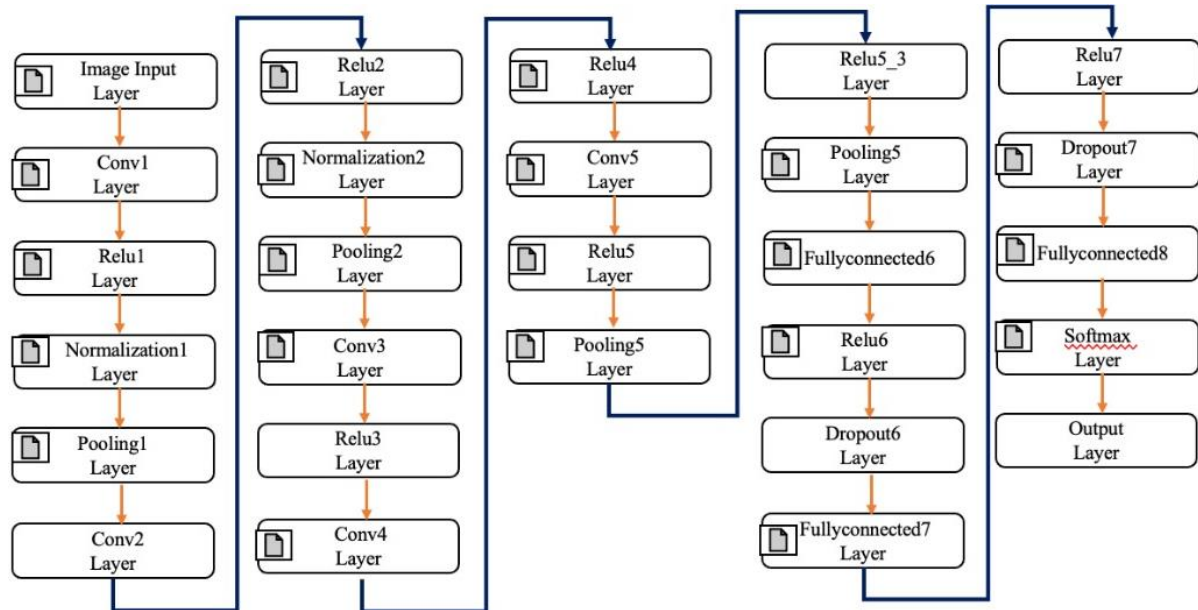


Figure 2. AlexNet architecture

2.3.2. SqueezeNet

A convolutional neural network of 18 layers deep is called SqueezeNet which is as shown in Figure 3. The network is more compact and has roughly 50 times less parameters than AlexNet, making it a smaller network [27]. The primary concepts behind SqueezeNet are the down sampling technique and the substitution of 1x1 filters for 3x3 filters when dealing with large feature maps.

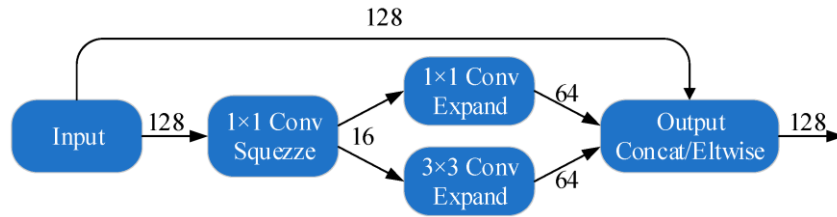


Figure 3. SqueezeNet architecture

2.3.3. MobileNet

An effective model for embedded and mobile vision applications, MobileNet is a simplified architecture that builds shallow deep convolutional neural networks using depth-wise separable convolutions [28]. Except for the first layer, which is a full convolutional layer, Mobilenet is formed using depth-wise separable convolutions. Batch normalization and ReLU non-linearity come after each layer. The final layer, which feeds to the softmax for classification, is a fully linked layer without any non-linearity. Stride convolution is employed for the first completely convolutional layer as well as depth-wise convolution for down sampling. If depth wise and pointwise convolution are counted as separate layers, mobilenet has a total of 28 layers. An architure of MobileNet is as shown in Figure 4.

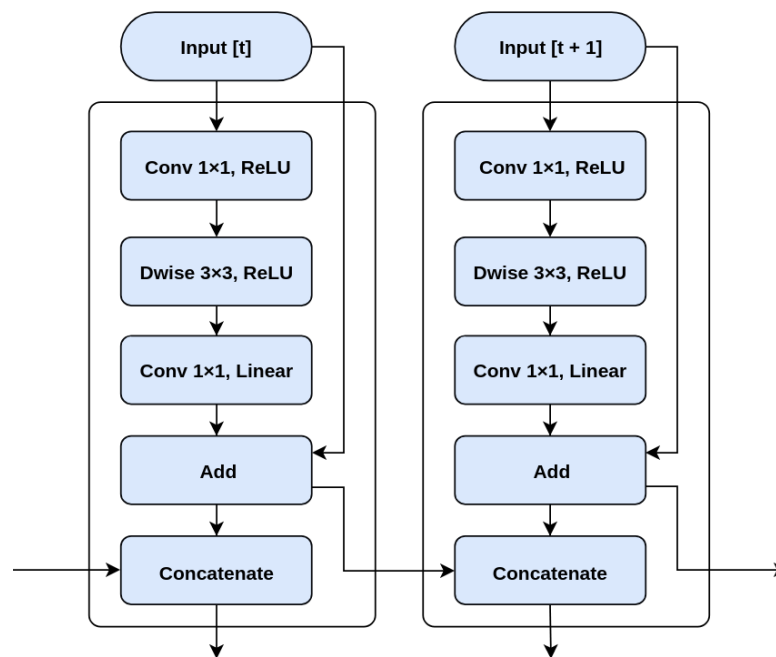


Figure 4. MobileNet architecture

2.3.4. VGG-16

Simonyan and Zisserman [29] introduced the convolutional neural network model, VGG-16, which was an improvement over the AlexNet model by employing smaller multiple kernel-sized filters rather than very large filters. As shown in Figure 5, it is a 16-layer deep network with a fixed size input layer of 224x224 red, green, blue (RGB) images. Rectified linear unit (ReLU) and maximum pooling after each convolutional layer. After the last convolutional layer, the network has three fully connected (FC) layers. While the third FC only has 1,000 channels, the previous two FCs each have 4,096 channels. The soft-max layer, which conducts multiple classifications, is the top layer i this network.

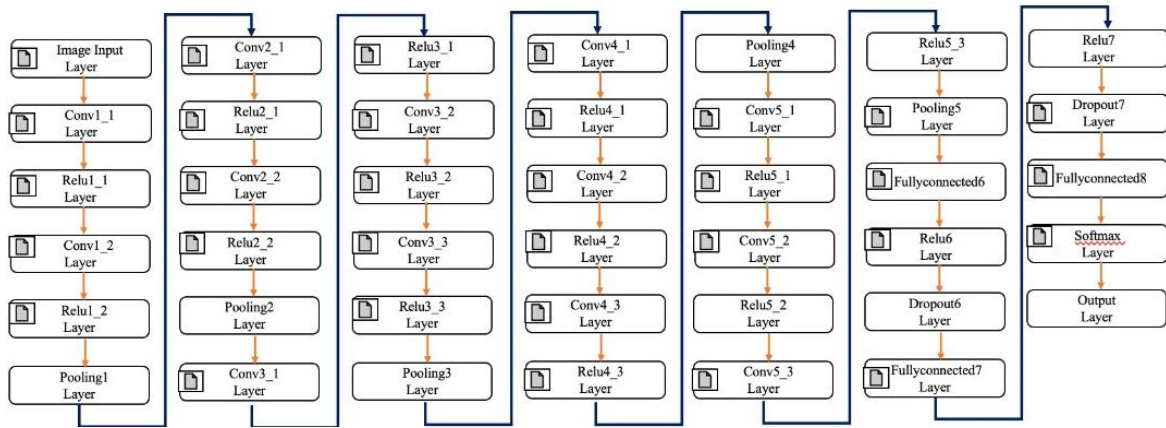


Figure 5. VGG-16 architecture

2.4. Optimization

The neural network is a framework that receives inputs and generates results. There is a mistake when the network output is compared to "the ground truth". The network uses this error to compute an error function while producing another output. This procedure, often referred to as optimization, is carried out repeatedly until the mistake is reduced to a minimum. The most popular optimization methods use gradient descent with a loss function as the objective function. The loss function calculates the discrepancy between the algorithm's current output and its desired output. The hyper-parameter optimization is given as follows to test the ideal pairing of a pre-trained deep learning model and optimizer for a multi-cassava illness classification task using transfer learning:

$$x^* = \underset{x \in X}{arg \min} f(x) \tag{5}$$

where $f(x)$ is the objective function to be minimized with x taking any value in the domain X . Bayesian optimization [23]-based hyper-parameter tuning reduces the amount of time needed to find the ideal collection of parameters. To achieve the best model performance, the following variables were changed: number of epochs, initial learning rate, mini-batch size, initial learning rate schedule patience and repetition with validation.

2.5. Performance metrics

The developed model was evaluated using the following statistical methods. Accuracy: this is given as (6).

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} \tag{6}$$

Loss function: given x as the space of all possible inputs ($x \in \mathbb{R}^d$), and y as the set of labels (possible outputs), a typical goal of classification algorithms is to find a function $f: x \rightarrow \mathbb{R}$ which best predicts a label y for a given input \vec{x} . However, an input \vec{x} may predict different values of y due to several factors such as incomplete information, noise in the measurement and so on. Thus, the goal of learning problem is to reduce the expected loss which is given as (7):

$$I[f] = \int_x^y V(f(\vec{x}), y) p(\vec{x}, y) d\vec{x} dy \tag{7}$$

where $V(f(\vec{x}), y)$ is the loss function and $p(\vec{x}, y)$ is the probability density function.

3. RESULTS AND DISCUSSION

3.1. Dataset distribution

The dataset's distribution is displayed in Figure 6. It demonstrates an unbalanced dataset that is skewed in favor of some datasets like those affected by the cassava mosaic disease (CMD), cassava bacteria blight (CBB), and healthy classes. The trained algorithm favors the biased diseases during detection because of this imbalance. Thus, methods like data preparation are used to solve the problem of imbalanced datasets.

We developed a method called synthetic minority over-sampling technique (SMOT) to address an unbalanced dataset [23]. In this technique, fresh synthetic data of comparable cases are created using a subset of data from a minority class and added to the original dataset. After using SMOT, a balanced dataset is shown in Figure 6 as well. The classification models are trained using a sample from the fresh dataset.

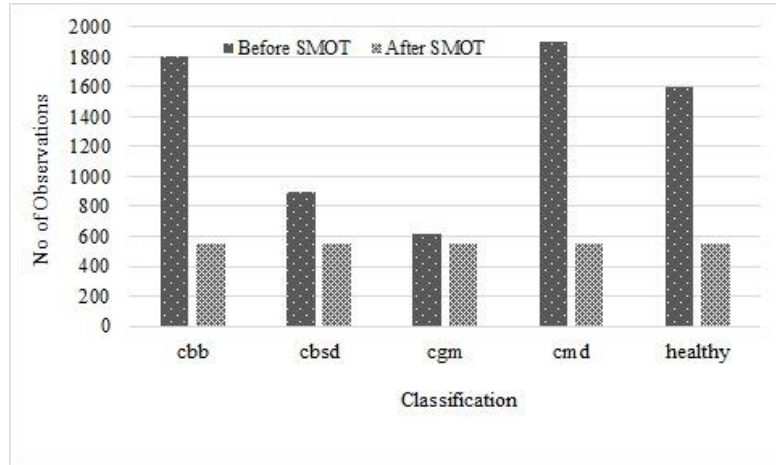


Figure 6. Dataset distribution

3.2. Model performance evaluation

Figure 7 shows compared to the complexity as shown in Figure 7(a) and speed as presented in Figure 7(b) of the algorithm, the performance of the trained algorithm as shown in Table 2 demonstrates that MobileNet performs relatively better than both VGG-16 and VGG-19 in terms of accuracy. The least validation loss among the chosen algorithms is achieved by MobileNet. These characteristics imply that it is the most ideal option for our system. Figure 8 shows how MobileNet performed. As can be seen, the algorithm can continue to improve performance by learning features from the inputs as the number of epochs is raised.

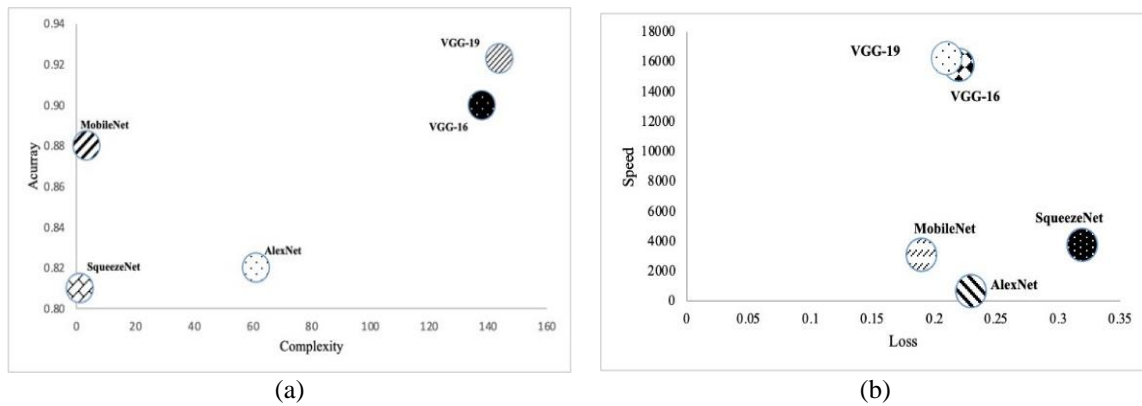


Figure 7. Model performance (a) complexity against accuracy and (b) speed against loss

Table 2. Model performance evaluation

Network architecture	Complexity	Accuracy (%)	Loss	Time per epoch (s)
AlexNet	61M	81.78	0.23	673
SqueezeNet	1.24M	81.18	0.32	3,750
MobileNet	3.5M	88.28	0.19	3,075
VGG-16	138M	90.08	0.22	15,750
VGG-19	144M	92.28	0.21	16,250

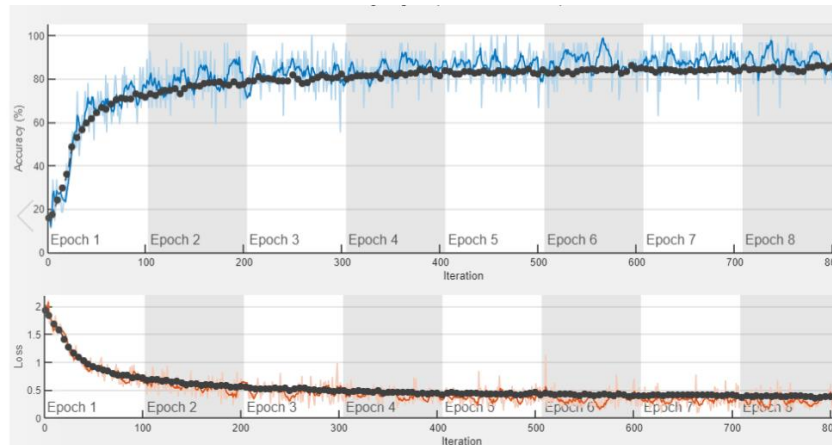


Figure 8. MobileNet training performance

3.3. Hyper parameter optimization

The hyper-parameters of the promising CNN model (MobileNet) were automatically tuned using Bayesian optimization technique. Bayesian algorithm aims to select the best combination of which the network is trained in all the specified range combinations. Table 3 shows the parameters considered, the range and optimum value.

Table 3. Optimum hyper-parameters results obtained using Bayesian algorithm

Parameters description	Range of parameters	Optimum value
Number of convolution and max pooling layers	[1, 2, 3, 4]	3
Number of FC layers	[1, 2, 3, 4]	3
Number of filters	[16, 24, 32, 48, 64, 96, 128]	96
Filter size	[3, 4, 5, 6, 7]	4, 4
Activation function	[ELU, SELU, ReLU, Leaky ReLU]	ReLU
Mini-batch Size	[4, 8, 16, 32, 64]	32
Momentum	[0.80, 0.85, 0.9, 0.95]	0.85
Learning rate	[0.0001, 0.0005, 0.001, 0.005]	0.0005
l2 Regularization	[0.0001, 0.0005, 0.001, 0.005]	0.0005

4. CONCLUSION

The study's findings demonstrate the effectiveness of the transfer learning technique for achieving high classification accuracy in picture identification and detection. With the use of a convolutional neural network, the study had been able to extract important information for algorithm training. CNN has aided in easing the strain associated with feature extraction and selection when using a conventional feature extraction method. The study has managed to strike a balance between complexity and accuracy, two crucial ideas in algorithm development. According to the results, MobileNet, which has a low complexity of 3.5 million neurons and an accuracy rate of 88.28%, is the best CNN algorithm for our application deployment. Future research will focus on creating an ensemble algorithm that takes advantage of the advantages of the appropriate method. This will guarantee that we can continue to evaluate how well the models work.

ACKNOWLEDGEMENTS

This work is supported by Bowen University, Iwo, Nigeria 2020 Research Grant BRE/2020/004.




REFERENCES

- [1] M. C. Shurtleff, M. J. Pelczar, A. Kelman, and R. M. Pelczar, "Plant disease," *Encyclopedia Britannica*, 2020. [Online]. Available: <https://www.britannica.com/science/plant-disease>. (Accessed: Mar. 04, 2021).
- [2] L. K. T. Al-Ani and E. L. Furtado, "The effect of incompatible plant pathogens on the host plant," in *Molecular Aspects of Plant Beneficial Microbes in Agriculture*, Elsevier, 2020, pp. 47-57.
- [3] M. Donatelli, R. D. Magarey, S. Bregaglio, L. Willocquet, J. P. M. Whish, and S. Savary, "Modelling the impacts of pests and diseases on agricultural systems," *Agricultural Systems*, vol. 155, pp. 213-224, 2017, doi: 10.1016/j.agsy.2017.01.019.




- [4] R. Cerda, J. Avelino, C. Gary, P. Tixier, E. Lechevallier, and C. Allinne, "Primary and secondary yield losses caused by pests and diseases: assessment and modeling in coffee," *PLoS One*, vol. 12, no. 1, p. e0169133, 2017, doi: 10.1371/journal.pone.0169133.
- [5] A. C. Velásquez, C. D. M. Castroverde, and S. Y. He, "Plant-pathogen warfare under changing climate conditions," *Current Biology*, vol. 28, no. 10, pp. R619-R634, 2018, doi: 10.1016/j.cub.2018.03.054.
- [6] Y. Elad and I. Pertot, "Climate change impacts on plant pathogens and plant diseases," *Journal of Crop Improvement*, vol. 28, no. 1, pp. 99-139, 2014, doi: 10.1080/15427528.2014.865412.
- [7] J. F. Sundström *et al.*, "Future threats to agricultural food production posed by environmental degradation, climate change, and animal and plant diseases - a risk analysis in three economic and climate settings," *Food Security*, vol. 6, no. 2, pp. 201-215, 2014, doi: 10.1007/s12571-014-0331-y.
- [8] G. Nakabonge, C. Samukoya, and Y. Baguma, "Local varieties of cassava: conservation, cultivation and use in Uganda," *Environment, Development and Sustainability*, vol. 20, no. 6, pp. 2427-2445, Dec. 2018, doi: 10.1007/s10668-017-9997-6.
- [9] J. P. Legg *et al.*, "Cassava virus diseases: Biology, epidemiology, and management," in *Advances in Virus Research*, vol. 91, no. 1, Academic Press Inc., 2015, pp. 85-142.
- [10] V. Verdier, S. Ojeda, and G. Mosquera, "Methods for detecting the cassava bacterial blight pathogen: a practical approach for managing the disease," *Euphytica*, vol. 120, no. 1, pp. 103-107, 2001, doi: 10.1023/A:1017516007945.
- [11] J. Legg and S. Winter, "Cassava Mosaic Viruses (Geminiviridae)," in *Encyclopedia of Virology*, Elsevier, 2021, pp. 301-312.
- [12] P. Gopika *et al.*, "Single-layer convolution neural network for cardiac disease classification using electrocardiogram signals," in *Deep Learning for Data Analytics*, Elsevier, 2020, pp. 21-35.
- [13] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging*, vol. 9, no. 4, pp. 611-629, 2018, doi: 10.1007/s13244-018-0639-9.
- [14] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, and G. Zhang, "Transfer learning using computational intelligence: a survey," *Knowledge-Based Systems*, vol. 80, pp. 14-23, 2015, doi: 10.1016/j.knsys.2015.01.010.
- [15] N. Best, J. Ott, and E. J. Linstead, "Exploring the efficacy of transfer learning in mining image-based software artifacts," *Journal of Big Data*, vol. 7, no. 1, pp. 1-10, 2020, doi: 10.1186/s40537-020-00335-4.
- [16] J. Boulent, S. Foucher, J. Théau, and P. L. St-Charles, "Convolutional neural networks for the automatic identification of plant diseases," *Frontiers in Plant Science*, vol. 10, p. 941, 2019, doi: 10.3389/fpls.2019.00941.
- [17] M. H. Saleem, J. Potgieter, and K. M. Arif, "Plant disease classification: a comparative evaluation of convolutional neural networks and deep learning optimizers," *Plants*, vol. 9, no. 10, pp. 1-17, 2020, doi: 10.3390/plants9101319.
- [18] H. F. Pardede *et al.*, "Plant diseases detection with low resolution data using nested skip connections," *Journal of Big Data*, vol. 7, no. 1, pp. 1-21, 2020, doi: 10.1186/s40537-020-00332-7.
- [19] A. Ramcharan *et al.*, "A mobile-based deep learning model for cassava disease diagnosis," *Frontiers in Plant Science*, vol. 10, p. 272, 2019, doi: 10.3389/fpls.2019.00272.
- [20] G. Sambasivam and G. D. Opiyo, "A predictive machine learning application in agriculture: cassava disease detection and classification with imbalanced dataset using convolutional neural networks," *Egyptian Informatics Journal*, vol. 22, no. 1, pp. 27-34, 2020, doi: 10.1016/j.eij.2020.02.007.
- [21] Y. Guo *et al.*, "Plant disease identification based on deep learning algorithm in smart farming," *Discrete Dynamics in Nature and Society*, vol. 2020, p. 2479172, 2020, doi: 10.1155/2020/2479172.
- [22] A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes, "Deep learning for image-based cassava disease detection," *Front. Plant Sci.*, vol. 8, pp. 1852, 2017, doi: 10.3389/fpls.2017.01852.
- [23] M. Pelikan, D. E. Goldberg, and E. Cantt U-Paz, "BOA: The Bayesian optimization algorithm," *Proceedings of the genetic and evolutionary computation conference GECCO-99*, pp. 525-532, 1999, doi: 10.5555/2933923.2933973.
- [24] C. E. Rasmussen and Z. Ghahramani, "Occam's Razor," in *Advances in Neural Information Processing Systems 13*, 13th ed., T. K. Leen and V. T. Dietterich, Eds. Cambridge, MA, USA: MIT Press, 2001, pp. 294-300.
- [25] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527-1554, 2006, doi: 10.1162/neco.2006.18.7.1527.
- [26] B. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84-90, 2012, doi: 10.1145/3065386.
- [27] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: alexnet-level accuracy with 50x fewer parameters and <0.5mb model size," 2017, Accessed: Mar. 04, 2021. [Online]. Available: <https://github.com/DeepScale/SqueezeNet>.
- [28] A. G. Howard *et al.*, "MobileNets: efficient convolutional neural networks for mobile vision applications," Apr. 2017, [Online]. Available: <http://arxiv.org/abs/1704.04861>.
- [29] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," Sep. 2014, [Online]. Available: <http://arxiv.org/abs/1409.1556>.

BIOGRAPHIES OF AUTHORS






Segun Adebayo    holds a Ph.D. in Electrical and Electronics Engineering. He is a passionate scholar and researcher in the Department of Mechatronics Engineering, Bowen University, Iwo. His research interest includes communication and signal processing, machine learning, robotics, and internet of things. He can be contacted through his email: segun.adebayo@bowen.edu.ng.






Halleluyah Oluwatobi Aworinde    is an alumnus of both Ladoko Akintola University of Technology, Ogbomoso and University of Ibadan where he bagged his B.Tech., M.Sc. and PhD respectively. His areas of interest include cognitive computing, mechanism design and intelligent systems. He currently holds a teaching position with Bowen University, Iwo, Nigeria. He is a member of some professional bodies which include International Association of Engineers (IAENG), Association for Computing Machinery (ACM) among others. He is equally a recipient of the 8th Heidelberg Laureate Forum, Germany. He can be reached through email: aworinde.halleluyah@bowen.edu.ng.






Akinwale O. Akinwunmi    is a Senior lecturer at the Computer Science Programme College of Computing and Communication Studies, Bowen University, Iwo. He has B.Sc. Computer Engineering, M.Sc. and PhD Computer Science all from Obafemi Awolowo University Ile-Ife, Nigeria. He is a member of the following professional bodies namely Association for Computing Machinery (ACM), International Association of Engineers (IAENG), Nigeria Computer Society (NCS) and Association for Computing Machinery (ACM) Special Interest Group on Economics and Computation. His research interest among others includes Networking and Communication, Hardware, Distributed Systems, Mobile Computing, Modelling and Simulation and Data Analytics. He is the group lead of Computing and Analytics Research Lab (a research cluster in Computer Science Programme Bowen University Iwo Nigeria). He can be reached through email: akinwale.akinwunmi@bowen.edu.ng.



Adebamiji Ayandiji    is a Senior Lecturer in the College of Agriculture, Engineering and Science, Bowen University, Iwo, Nigeria. He is an alumnus of Obafemi Awolowo University, Ile-Ife, University of Ibadan and University of Ilorin where he obtained his B.Sc., M.Sc. and Ph.D. respectively. His area of research is Agricultural Extension and Rural Development. He can be contacted through email: adebamiji.ayandiji@bowen.edu.ng.



Awoniran Olalekan Monsir    holds a doctoral degree in Computer Science from Obafemi Awolowo University, Ile-Ife. He is an ardent, computer programmer, information systems expert, and data scientist. His research interests are in information systems, knowledge engineering, and data science. He currently teaches Computer Science courses at the Department of Computer Science and Information Technology, Bowen University, Iwo, Nigeria. He can be contacted through email: olalekan.awoniran@bowen.edu.ng.