MaldoNetB0 - Enhanced EfficientNetB0 for Disease Detection in Apple (Malus domestica) Using Optuna with Bayesian Optimization (TPE)

Shreyas D M[1], Anika R Nayak[2], Shree Bhuvan S R[3], Udeep S Lokesh[4]

Tintu Vijayan

School of Computer Science and Engineering, Presidency University,

Bangalore, Karnataka-560064, India

tintu.20223cse0013@presidencyuniversity.com

***Abstract* — The developed study investigates how detection of plant leaf diseases can be improved, including deep learning coupled with EfficientNetB0 for feature extraction and Optuna for hyperparameter optimization. EfficientNetB0 is effective in feature extraction and becomes the backbone for classifying images of plant leaves into different diseases. This work used Optuna to tune important parameters like learning rate and dropout rate and obtained an optimal tradeoff between accuracy and computational efficiency. To handle such high computational demands, early stopping and learning rate scheduling were implemented. Early stopping was done to avoid overfitting, where the training stopped when the validation performance had leveled off. Learning rate scheduling allowed the model to smoothly converge. Furthermore, Optuna's trials improved this model's performance even more-achieving validation accuracy well over 94% with reduced loss values of 0.1382. By using our optimized model MaldoNetB0, we have seen the training accuracy increase from 68.69% to 97.37% within 15 epochs, with rather constant validation accuracy around 98-99%. Additionally, the AUC came back as 0.9998, together with balanced precision and recall, indicating that this model is very effective for any kind of disease detection. This approach has great potential for early disease diagnosis and plant health; many such approaches can be employed in agricultural settings for scalable solutions to improve crop yield and management**.

Keywords: MaldoNetB0 · Plant Leaf Diseases · EfficientNetB0 · Hyperparameter Tuning · Hyperparameter Optimization · Optuna · Early Stopping · Feature Extraction

1. Introduction

One of the major areas needing further advancements in agri-tech involves plant disease detection for maximum yield and minimum loss to the economy. Traditional methods of plant disease identification are labor-dependent, time-consuming, and require expert knowledge. This approach is generally so inconsistent that it could be utterly impracticable over large-scale farming. Image processing techniques have the potential capability for complete automation, with the use of machine learning models, and enhance the accuracy of detecting plant diseases.

The paper reviews the methodologies concerning image processing and the MaldoNetB0 model in detecting diseases on plant leaves, especially those affecting crops such as apples. Consequently, major steps in this process involve image acquisition, preprocessing, segmentation, feature extraction, and classification. In this work, algorithms will be adopted to enhance the efficiency of disease detection through hyperparameter optimization, CNNs, and data augmentation.

Also, some previous related work has attempted to employ metaheuristic optimization techniques with the view of enhancing the efficacy of certain tasks in the chain of image processing. The optimization of several steps involved in the image processing chain makes use of hyperparameter optimizers for feature selection and tuning the parameters of the classifiers, including Optuna. These optimization methods are put forward to enhance accuracy and computational efficiency in disease detection systems.

The results obtained in this work have shown that the integration of image processing with deep learning and optimization of hyperparameters stands potentially to achieve a paradigm shift in plant disease detection. It shows a scalable, effective solution to farmers for maintaining healthy crops with optimization in the use of pesticides. This paper is part of ongoing efforts toward developing intelligent agricultural systems that could eventually alter the face of crop management and productivity.

2. Related Work

Samajpati BJ and Degadwala SD, (2016) has proposed a hybrid approach for apple fruit diseases detection and classification using random forest classifier, this research paper presents a method for detecting and classification of apple fruit diseases by using image processing techniques. The three common general diseases were considered in this research work: apple scab, apple rot, and apple blotch diseases. The proposed methodology incorporates feature extraction for color and texture from test images and then feature-level fusion. A random forest classifier will come into action for disease classification, and in cases of detection of a disease, infected area segmentation by using K-means clustering is performed. The results clearly depict that the feature combination using GCH, LBP, CLBP, and Gabor features has increased the accuracy of disease classification significantly. This study has evidence that fusion of features always performs better than individual ones. The results of this paper highlight that the incorporation of color and texture features with a machine learning technique appears effective in the improvement of accuracy and efficiency in detecting diseases affecting apple fruits. Texture feature extraction combined with color and K-means clustering segmentation followed by a random forest classifier did well in this application. [1]

The results obtained in this work have shown that, with the integration of image processing with deep learning and optimization of hyperparameters, a paradigm shift can be achieved in plant disease detection. It serves as a scalable and efficient solution to farmers to maintain healthy crops by optimizing the use of pesticides. This paper forms part of the efforts towards the development of intelligent agricultural systems that might eventually change the face of crop management and productivity. The review summarizes ways in which image processing techniques can be applied to plant disease detection and classification, keeping in mind the important role agriculture plays in the economy of Tanzania. Most staple crops, such as maize and cassava, are attacked by diseases that afterwards have formed the basis for food insecurity. The current methodology is reviewed whereby, at present, the methodologies can be considered to fall broadly into two categories, namely Convolutional Neural Network and conventional approaches. Basically, CNNs are known for handling feature extraction by themselves from normal images and achieve high accuracy in disease classification. The sad reality is that most traditional works done previously relied utterly on manual feature extraction, which is a time-consuming task. This paper has married the processing of images to smart expert systems to assist farmers in reducing losses by ensuring that detection takes place in time, hence enhancing food security. [12]

Deepak AH, Gupta A, Choudhary M, Meghana S (2019) has proposed a disease detection in tomato plants and remote Monitoring of agricultural parameters, this paper "Disease Detection in Tomato Plants and Remote Monitoring of Agricultural Parameters" has proposed a methodology which utilizes deep learning and IoT technologies in monitoring the health status of tomato plants. It utilizes CNNs for the detection of diseases such as Leaf Curl and Septoria Leaf Spot affecting the leaves. It was trained on images of leaves for the prediction of diseased or non-diseased plants. Using the deep learning model, it attained an accuracy of 95% for training and 92% in validation. The paper also covers, besides disease detection, remote monitoring of agricultural parameters such as soil moisture, temperature, and humidity by using sensors. These sensors will use an application to provide real-time data to the farmers, thus enabling automated irrigation and remote control over farm conditions. The FC-28 soil moisture sensor and the DHT11 temperature and humidity sensor will be employed in this, with data transmission through the Blynk IoT platform. The system can be applied to farmers to monitor the remote field areas and manage them without a large need for laborers, improving the efficiency of farming. The idea of the paper is to integrate CNN for image classification with IoT for remote monitoring; this will provide an effective way toward improving agricultural practices-for instance, in control of diseases. [6]

Pandey AK, and Soni A, (2017), proposed a CNN-based machine learning approach for detecting and classifying diseases in the tomato plant. The three most prevalent diseases taken for consideration include the following: Leaf Curl, Septoria Spot, and Bacterial Blight. Further, this approach was utilized for the acquisition of images in both healthy and diseased states of tomato leaves; later on, preprocessing of images is done for the removal of noise and enhancement of the features. This dataset was then used to train a CNN model to extract features automatically that are relevant for disease classification, thus avoiding feature engineering. The results showed that the CNN model recorded high classification accuracy for its efficiency in capturing complex patterns in the images, hence the efficacy of deep learning applications in agriculture. Recent research emphasizes that CNNs have an advantage over conventional machine-learning models by handling feature extraction automatically, and thus it is suitable to install large-scale agricultural field real-time monitoring of disease conditions. [7]

Dhingra G, Kumar V, Joshi HD (2018) Study of digital image processing techniques for leaf disease detection and classification, this paper reviews some digital image processing methodologies applied to plant leaf diseases recognition and classification. The authors analyze the various steps taken in image-processing methods and using machine learning between 1997 and 2016, to identify ways of improving accuracy, reliability, and speed in leaf disease inspections, compared to the classical methods, i.e., those that are based on human vision, which is usually inconsistent and time-consuming. In this study, two large groups are made for the classification of techniques: disease detection and classification. Emphasis has also been drawn to feature extraction methodologies of color and texture using co-occurrence matrices and hue-saturation transformations that have been deemed so vital in the segmentation of diseased areas on leaves. Classifiers discussed herein that have been involved in the correct classification of these diseases across multiple plant species include those based on Support Vector Machine, Random Forest, and Neural Networks. Additionally, the authors identify a few challenges regarding natural light conditions in which these methods are put to work and describe the future prospects of improvement in image-based plant disease detection. This survey opens vistas toward the potentials of integrating advanced image-processing techniques with classification for developing automated, precise tools aimed at agricultural disease management. [9]

3. Proposed System

The Fig.1 depicts the architecture of the MaldoNetB0 model, used for plant leaf detection and optimized with Optuna. The input image of size 224x224 undergoes several convolutional layers and shrinks in spatial dimensions while increasing depth in features. Such a spatial feature map, from shallow to deeper, captures highly variable details ranging from fine to abstract features at different levels from P3 to P7. These are then flattened and fed into the classification layer to identify the species of the plant. Optimized hyperparameters using Optuna are learning rate, which helps in increasing the model's performance in identifying leaves.

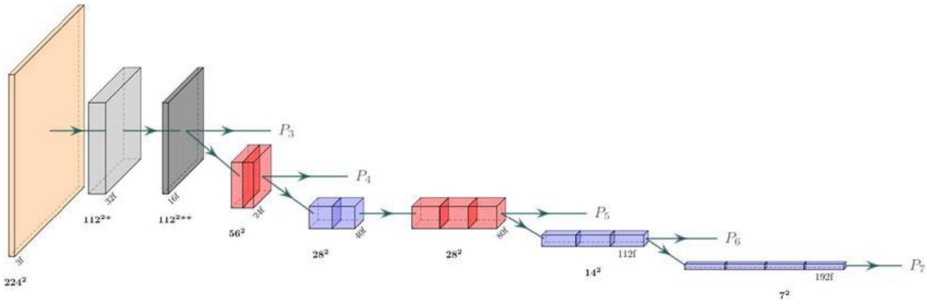


Fig 3.1 The architecture of the EfficientNetB0 backbone in our transfer learning model consists of blue and red blocks representing mobile inverted bottleneck convolution (MBConv) blocks (see Fig 3.1), with kernel sizes of 3x3 and 5x5, respectively. The light gray block indicates a stride convolution followed by batch normalization and a Swish activation. The dark gray block represents an MBConv block with an expansion factor of 1 and a kernel size of 3. The resolution at each level is shown in bold, while the number of filters (output channels) is indicated with an 'f' suffix.

4. Methodology

4.1 Dataset Description

The dataset used in this project is sourced from the publicly available Plant Village database, comprising over three thousand images across 14 different crop species. Sample images of apple leaves are shown from Fig. 4.1.1 to Fig. 4.1.4. For our analysis, we utilized 3160 images of apple leaves, each assigned one of the following class labels:

Fig4.1.1 - Apple scab leaf

Fig4.1.2 - Black rot leaf

Fig4.1.3 - Apple rust leaf

Fig4.1.4 - Healthy leaf

4.2 Hyperparameter Optimization

Bayesian optimization using the TPE algorithm through the Optuna framework was done for the improvement of an EfficientNetB0-based model performance in disease detection. The optimization minimized the validation loss with respect to fine-tuning dropout rate between 0.3 and 0.7, and learning rate logarithmically $(1 \times 10^{-5}) to (1 \times 10^{-3})$. These hyperparameters are iteratively adjusted by the TPE algorithm, which learns from each trial in order to guide subsequent trials toward more promising values. This allowed convergence to an optimized dropout and learning rate combination that minimized validation loss, thereby improving the model's generalization on unseen data in an efficient manner. The resultant tuned model was therefore clear evidence with strong robustness in terms of predictive accuracy, further intimating the efficiency of Bayesian optimization in deep learning aspects of disease prediction.

4.3 Training Configuration

Optuna optimizes dropout_rate and learning_rate by trying a different value of them in every trial. The architecture consists of the model with data augmentation, followed by a pre-trained base model; afterwards, global average pooling, batch normalization, and dense layers with dropout. It also employs the Adam optimizer, categorical cross-entropy loss, and monitoring of accuracy. Model Training EarlyStopping to prevent over-fitting, and ReduceLROnPlateau for dynamic changes of the learning rate. Limited to five epochs per trial for faster evaluations.

Optuna Optimization: Starts a study to minimize validation loss with n_trials=10.
Logs results for every trial, including learning rate changes and dropout rates.

4.4 Model Architecture Equations

The EfficientNetB0 model is utilized as a feature extractor with additional custom layers for classification.

4.4.1. Input Layer (Image):

$$(\boldsymbol{X} \in R^{224 \times 224 \times 3})$$

where X is an RGB image resized to 224x224 pixels.

4.4.2 Feature Extraction Layer (EfficientNetB0):

EfficientNetB0's layers are pre-trained on ImageNet, and they process X to extract useful features F:

$$\boldsymbol{F} = \text{EfficientNetB0}(\boldsymbol{X})$$

4.4.3 Global Average Pooling:

After feature extraction, we apply Global Average Pooling to reduce the spatial dimensions:

$$\boldsymbol{F_{avg}} = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} f_{ij}$$

4.4.4 Fully Connected Layers:

Each dense layer applies a transformation with ReLU activation:

$$\boldsymbol{y} = \mathrm{ReLU}(\boldsymbol{W} \cdot \boldsymbol{x} + \boldsymbol{b})$$

4.4.4.1 Followed by dropout for regularization:

$$\boldsymbol{h_{dropout}} = \mathrm{Dropout}(\boldsymbol{h}, \mathrm{rate} = p)$$

Finally, the output layer with softmax activation for multi-class classification

4.4.4.2

$$\widehat{\boldsymbol{y}} = \mathrm{Softmax}(\boldsymbol{W_2 h_{dropout}} + \boldsymbol{b_2})$$

4.5 Objective Function for Hyperparameter Tuning:

This model proceeds with an optimization process, driven by the objective function, while Bayesian optimization uses the Tree-Structured Parzen Estimator. The aim is to minimize validation loss with the adjustment of important hyperparameters: the dropout rate p and learning rate α. The objective may be expressed:

$$\min_{p,\alpha} \boldsymbol{L_{val}} = (\boldsymbol{x_{val}}, \boldsymbol{y_{val}}; p, \alpha)$$

Where:

p: Dropout rate (probability) - regulates the amount of regularization applied in this model by randomly setting a fraction of input units to zero during training, which helps prevent overfitting.

α (learning rate) defines the step size at each update of model parameters and significantly affects the convergence speed and accuracy of the model.

This is an optimization framework, where the final model configuration needs to strike a balance between effective learning and prevention of overfitting, very important in plant disease detection applications, which typically have high accuracy on predictions on new data.

4.6 Cross-Entropy Loss (Objective):

The categorical cross-entropy loss, often referred to as the cost or objective function on classification problems, essentially quantifies the difference between true labels and predicted probabilities in multiclass classification problems. Mathematically, this is represented as:

$$\mathcal{L} = -\sum_{c=1}^{C} y_c \log(\hat{y}_c)$$

where:

C=4 indicates the total number of classes (e.g., "Apple Scab," "Black Rot," "Cedar Apple Rust," and "Healthy" in apple leaf disease detection).

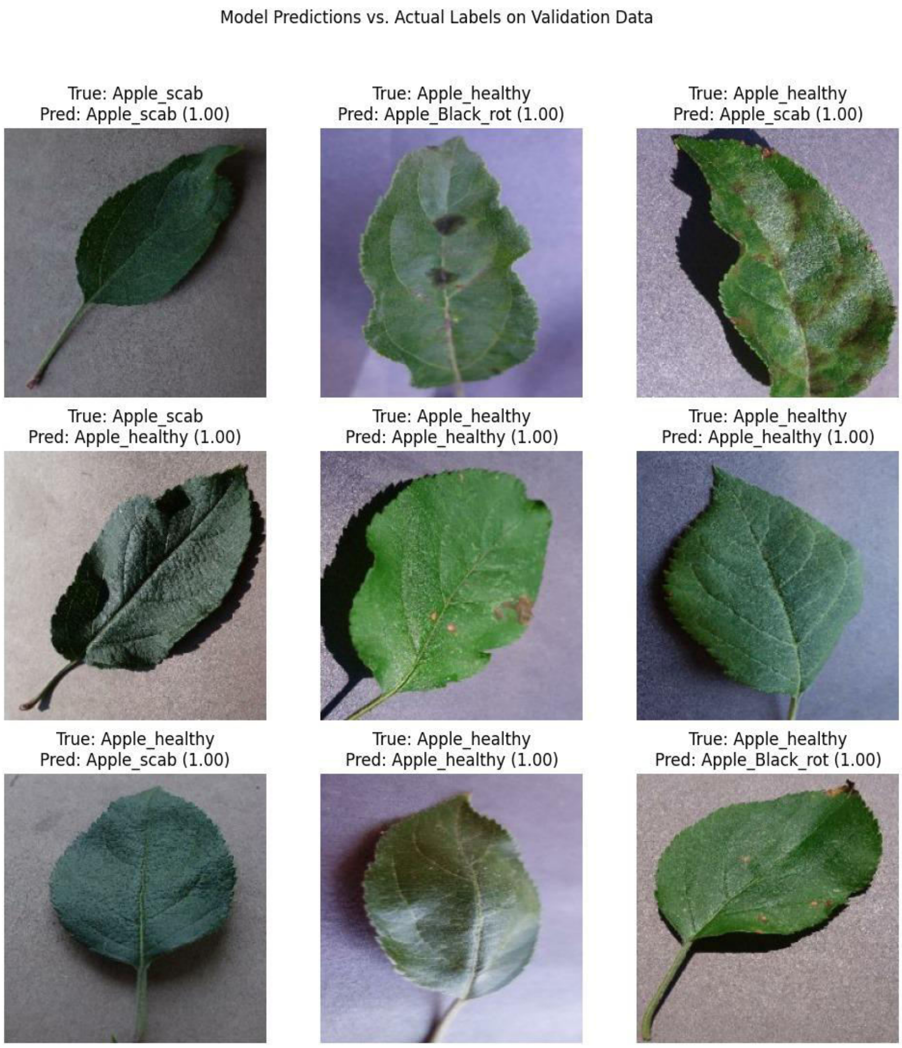
4.7 Learning Rate Adjustment (ReduceLROnPlateau):

The ReduceLROnPlateau callback adjusts the learning rate whenever the validation loss stops improving for a specified patience period. This prevents overshooting of minimum loss and helps the model converge further with a better solution by dynamically reducing the learning rate. This could be summed up in a learning rate update rule:

$$\alpha_{\text{new}} = 0.2 \times \alpha_{\text{current}}$$

5. Results and Analysis:

Fig 5.1



The model performs well on the validation set, matching a large quantity with correct sets of predictions. However, its misclassifications are quite notable. For example, it correctly identifies "Apple_scab" in one of the instances and then makes an incorrect classification of "Apple_healthy," where the correct classification should have been "Apple_scab." This particular confusion matrix between "Apple_scab" and "Apple_healthy" just shows that the model does not effectively tell the difference between the classes.

The solution to this problem would be the need for more exploratory work regarding the distinctive features of the classes. If some traits distinguish "Apple_scab" from "Apple_healthy", then such traits should get more attention in training, based on the model learning of it. Moreover, adding more examples in the training dataset pertaining to both the classes may help in learning the fine discriminants between them.

Although the general performance of the model appears promising, a more detailed discussion involving a number of performance metrics such as accuracy, precision, recall, and F1-score will make clearer the effectiveness of the model. These can further help in indicating not only the percentage of correct classifications but also the balance between false positives and false negatives, hence showing where the model is doing well and where improvements need to be done.

In all, the general performance of the model is good; yet, looking further at these errors, as well as doing data augmentation and an analysis based on the performance metrics, might greatly improve the accuracy and reliability of this model.

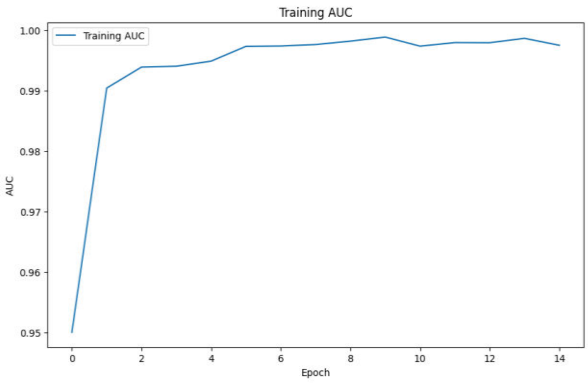


Fig 5.2 – Two main metrics were AUC, or Area Under the ROC Curve, and precision, and these were measured over several epochs in training. Firstly, the metric of AUC in the first plot increases rapidly and by the fifth epoch is mostly stabilized near the perfect mark of 1.0. This trend shows the strong ability of the model to discriminate between classes, hence learning well and converging with minimal false positives and false negatives.

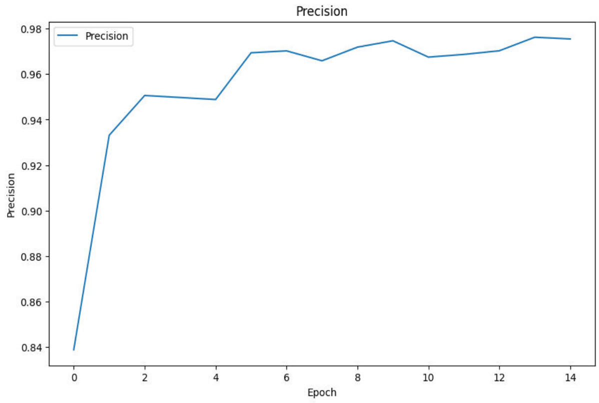


Fig 5.3 - In the second plot, the precision metric also shows quite an uphill, reaching a value of approximately 0.98 and remaining stable after several epochs. Such a high precision score indicates that this model has a very low number of false positives and is able to correctly identify positive cases without misclassifying many of them. The findings taken together, therefore, depict that the model is well-suited for applications in which accurate class distinction is highly essential, making a good discriminatory power without any compromise in precision.

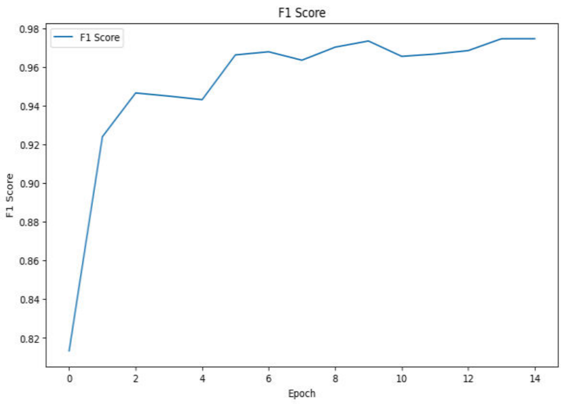


Fig 5.4 - The following plot illustrates the progress of the F1 score for 15 epochs. The F1 score rises rapidly during the first few epochs, then goes up very high after epoch 5 to above 0.96 and reaches an appropriate settling value of about 0.97 for the rest of the training process. This means the model has a very strong balance between its precision and recall, thus giving high accuracy in assigning true positives with reduced false positives and false negatives. The steady trend of performance after the initial increase of the F1 score metric is representative of the model's stable performance across the epochs with a very minimal influence concerning overfitting or underfitting.

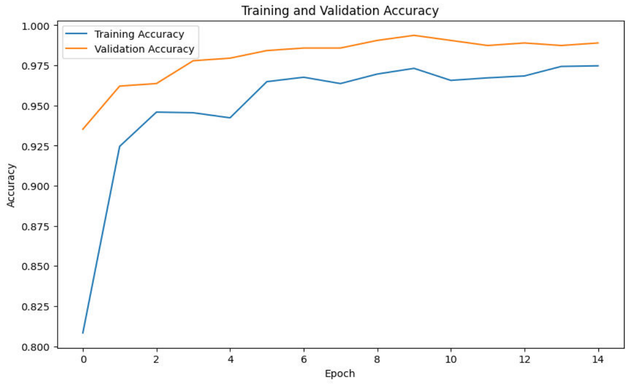


Fig 5.5 - The above plot represents training and validation accuracy of the model in 15 epochs. Training accuracy increases almost monotonically with the epochs, which serves as a good indicator that the model is capable enough to learn from the provided training data. On the other hand, validation accuracy also improves quickly in the early epoch and then after stabilizes at approximately 0.98, which suggests good generalization of the model on unseen data.

Further verification with other performance metrics, validation results, and possible overfitting since there is consistent high AUC and precision could give more information to provide evidence on generalization ability over unseen data.

6. Conclusion and future scope:

MaldoNetB0, the proposed model with EfficientNetB0 as its backbone, optimized using a Bayesian approach via Optuna, is able to classify diseases on apple leaves with high accuracy and computational efficiency. This trained model shows the achievement of high performance with an accuracy of more than 97.3% and an AUC close to 99.7%, therefore verifying that this model might fulfill excellent results in plant disease detection and turns out to scalable and reliable in early identification in agricultural contexts.

It essentially shows the feasibility of hyperparameter optimization coupled with transfer learning using deep learning models for increased agricultural productivity and reduction in losses due to plant diseases.

Future Works:

In fact, this model can still be further developed for more plant species and more kinds of diseases in the future for better applicability within the general setting of agriculture. IoT sensors can integrate with on-site real-time image processing to be done for disease monitoring. This can also be explored by other research works in the future that will involve advanced optimization techniques or architectures such as EfficientNetV2 or vision transformers. More advantages in big datasets regarding better accuracy and efficiency are expected to increase.

7. References

[1] Samajpati BJ, Degadwala SD (2016) Hybrid approach for apple fruit diseases detection and classification using random forest classifier. In: 2016 International conference on communication and signal processing (ICCSP). IEEE, pp 1015–1019.

[2] Sandika B, Avil S, Sanat S, Srinivasu P (2016) Random forest-based classification of diseases in grapes from images captured in uncontrolled environments. In: 2016 IEEE 13th international conference on signal processing (ICSP). IEEE, pp 1775–1780.

[3] Zawbaa, HM, Hazman M, Abbass M, Hassanien AE (2014) Automatic fruit classification using random forest algorithm. In: 2014 14th International conference on hybrid intelligent systems. IEEE, pp 164–168.

[4] Iqbal MA, Talukder KH (2020) Detection of potato disease using image segmentation and machine learning. In: 2020 International conference on wireless communications signal processing and networking (WiSPNET). IEEE, pp 43–47.

[5] Chauhan MD (2021) Detection of maize disease using random forest classification algorithm. Turkish J Comput Math Educ (TURCOMAT) 12(9):715–720.

[6] Deepak AH, Gupta A, Choudhary M, Meghana S (2019) Disease detection in tomato plants and remote Monitoring of agricultural parameters. In: 2019 11th International conference on advanced computing (ICoAC). IEEE, pp 28–33.

[7] Hidayatuloh A, Nursalman M, Nugraha E (2018) Identification of tomato plant diseases by leaf image using squeeze net model. In: 2018 International conference on information technology systems and innovation (ICITSI). IEEE, pp 199–204.

[8] Sarangdhar AA, Pawar VR (2017) Machine learning regression technique for cotton leaf disease detection and controlling using IoT. In: 2017 International conference of electronics, communication and aerospace technology (ICECA), pp 449–454. https://doi.org/10.1109/ICECA.2017. 8212855

[9] Dhingra G, Kumar V, Joshi HD (2018) Study of digital image processing techniques for leaf disease detection and classification. Multimedia Tools Appl 77(15):19951–2000

[10] Das D, Singh M, Mohanty SS, Chakravarty S (2020) Leaf disease detection using support vector machine. In: 2020 International conference on communication and signal processing (ICCSP). IEEE, pp 1036–1040

[11] Albattah W, Nawaz M, Javed A, Masood M, Albahli S (2022) A novel deep learning method for detection and classification of plant diseases. Complex Intell Syst 8(1):507–524

[12] Hungilo GG, Emmanuel G, Emanuel AW (2019) Image processing techniques for detecting and classification of plant disease: a review. In: Proceedings of the 2019 international conference on intelligent medicine and image processing, pp 48–52

[13] Khirade SD, Patil AB (2015) Plant disease detection using image processing. In: 2015 International conference on computing communication control and automation. IEEE, pp 768–771

[14] Ramesh S, Vydeki D (2020) Recognition and classification of paddy leaf diseases using optimized deep neural network with Jaya algorithm. Information processing in agriculture 7(2):249–260.