SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

# Tomato Leaf Classification Using Computer Vision and Deep Learning: Comparing Different EfficientNets

Kaan Eroltu[1*]

[1]United World College of the Adriatic, 34011 Duino TS, Italy. E-mail: ad22kero@uwcad.it

**Article Info**

Volume 4, Issue 1, January 2024
Received : 17 September 2023
Accepted : 10 December 2023
Published : 05 January 2024
doi: 10.51483/IJAIML.4.1.2024.61-79

## Abstract

Agriculture is an essential field that includes crop production, plant and fruit growing, and livestock. Plant disease is a significant challenge in agriculture, which can drastically impact crop production and lead to reduced productivity and potentially severe shortages. Hence, it is essential to detect plant diseases as fast as possible in order to start separating diseased ones from healthy ones. However, this process is arduous and challenging to accomplish manually. This paper shows a possible automation technique using EfficientNet CNN models. Gray-level, binarized, and color-level datasets were separately given in this study. The images in the dataset were resized. In order to increase the training data's variety, the input images underwent a horizontal flip. The data was rotated, helping the model in handling minor orientation variances. Randomized zoom was implemented to improve the model's ability to recognize leaf images from varying distances and sizes. The highest training accuracy achieved is 99.81% with the EfficientNetB5 model at 50 epochs and a batch size of 128. The EfficientNetB0 model achieves the lowest training accuracy with 97.17% accuracy at 20 epochs and a batch size of 16, however. The highest testing accuracy achieved is 97.83% by the EfficientNetB7 model with 50 epochs and a batch size of 64; on the other hand, the lowest testing accuracy is 77.84% by the EfficientNetB4 model with 20 epochs and a batch size of 32.

***Keywords:*** *Leaf disease, EfficientNet, CNNs, Computer vision, Tomato leaf disease, Plant diseases*



## 1. Introduction

Oxygen and edible plants are integral to people's daily life. It is known that as human civilization has risen, agriculture has become more advanced (Hughes and Salathe, 2015). In an environment where the population is growing, there is always a need for sufficient amounts of agricultural products. With the onset of hunger, various diseases begin to emerge.

* *Corresponding author: Kaan Eroltu, United World College of the Adriatic, 34011 Duino TS, Italy. E-mail: ad22kero@uwcad.it*

Tomato, *Solanum lycopersicum*, is a widely used farm good which that be found in many breakfasts. *Solanum lycopersicum* is from *Solanaceae* family, originally from the Andean region of South America (OECD, 2017). The *Solanum* genus comprises around 1,500 species (OECD, 2017). *Solanum lycopersicum* originated from two wild progenitor species, *Solanum pimpinellifolium* and *Solanum cerasiforme* (OECD, 2017). In 2020, 186.821 million metric tonnes of tomatoes were produced, and in 2021, it climbed up to 189.133 million (François-Xavier, 2021). Major tomato producers include China, with 67.538 million tons, India, with 21.181 million tons, Turkey, with 13.095 million tons, and the USA, with 10.475 million tons (François-Xavier, 2021).

Diseases and pests negatively affect agricultural goods in terms of quality and quantity. Plant diseases in tomatoes can happen due to pathogens, including fungal, bacterial, phytoplasma, virus, viral, nematode, and viroid pathogens. In the below table (Table 1), you may see some of the pathogens in tomato plants.

**Table 1: Tomato Pathogens and their Scientific Names**

| Pathogens | Pathogen Names |
|---|---|
| Fungal Infections | *Alternaria solani, Phytophthora infestans, Colletotrichum coccodes, Fusarium oxysporum f.sp. lycopersici, Verticillium dahliae or V. albo-atrum* |
| Bacterial Infections | *Pseudomonas syringae pv. tomato, Xanthomonas vesicatoria, Pseudomonas corrugata* |
| Viral Infections | Tomato Spotted Wilt Virus., Tobacco Mosaic Virus., Tomato yellow leaf curl virus. |
| Nematode Infections | *Meloidogyne spp.* |
| Viroid | Potato spindle tuber viroid, Tomato apical stunt viroid |

**Source:** *Kanda et al. (2022), Panno et al. (2021), Das (2020), Gilardi et al. (2021), Rivarez et al. (2021), Rodrigues andFurlong (2022)*

Also, environmental stress and pests, such as Aphids, Tomato hornworm (*Manduca quinquemaculata*), Whitefly (*Trialeurodes vaporariorum* and *Bemisia tabaci*) can induce diseases in tomatoes. These diseases may yield quality loses, hence accurate and early detection is essential. In the past, the detection processes were performed manually. The process of identifying plant diseases and pests through visual inspection by experts

**Table 2: Machine Learning and CNN Differences**

| Traditional Machine Learning Methods | Deep Learning Methods |
|---|---|
| They need manual design features and classifiers | CNNs are capable of autonomously learning features from huge volumes of data. |
| Image segmentation techniques include threshold segmentation, along with Roberts, Prewitt, Sobel, Laplace, and Kirsh edge detection methods, and region segmentation. When it comes to feature extraction methods, tools such as SIFT, HOG, LBP, along with shape, color, and texture feature extraction are commonly used, For classification, methods like SVM, BP, and Bayesian approaches are typically used. | Only CNN is needed. |
| The imaging environment requirements are relatively stringent, necessitating a high contrast between lesion and non- lesion areas, and minimal noise. | The combination of adequate learning data and high-performance computing units is crucial for achieving optimal results in machine learning tasks. |
| Having sufficient learning data and using powerful computing units are essential elements in obtaining the best outcomes in machine learning endeavors. | Deep learning methods possess the capability to adapt to specific changes in both real and intricate natural environments. |

is laborious and often prone to a significant amount of errors (Liu and Wang, 2021). For farming applications, the development of efficient, quick, and highly reliable computer-assisted disease detection systems has emerged.

Machine learning models were developed prior to the advent of deep learning models. For instance, Dubey and Jalal's study detected three different apple fruits using local binary patterns and k-means clustering methods (Dubey and Jalal, 2012). Also, a Support Vector Machine (SVM) was used to classify (Dubey and Jalal, 2012). Singh and Misra (2017) classified five different diseases and four different plant species using SVM. Conventional machine-learning techniques are only effective when feature extraction is done properly (Altunta and Kocamaz, 2021). Furthermore, segmentation is important for feature extraction (Altunta and Kocamaz, 2021). There are major differences between traditional machine learning algorithms and deep learning techniques. The Table 2 above, modified from Liu et al.'s study, shows the difference (Liu and Wang, 2021).

In computational agriculture, leaf detection is very significant (Singh and Misra, 2017). After the recent advancements, multiple neuron networks start to become popular: Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Feedforward Neural Networks (FNN), and Artificial Neural Networks (ANN) are only some of them. These neuron networks help to detect diseases easier as plant diseases produce a variety of visible characteristics/symptoms, which neural networks can learn.

The objective of this research paper is to identify the most accurate model for tomato leaf disease classification among all the EfficientNet models. Color-level, gray-level, and binarized image datasets were used along with eight CNNs. Also, many variables were tested in this research paper, including EPOCHs and batch sizes. This is the only paper in the literature that compares all the EfficientNets for tomato leaf disease detection.

## 2. Literature Review

Considerable advancements have been achieved in detecting diseases in various plants, including bananas, cucumbers, apples, tomatoes, rice, and peppers (Kanda *et al.*, 2022; Zhou *et al.*, 2019).

Mohanty *et al.* (2016) analyzed 54.306 plant leaf images from a range of 38 class labels. The team modified the image dimensions to 256 x 256 pixels, which was initially 128 x 128, and used AlexNet and GoogLeNet for the ImageNet dataset. They used three different datasets, and the datasets were in three categories: color, gray-scale, and leaf segmented. The team applied transfer learning and used a color dataset in GoogLeNet; they split the dataset into 80/20 for the training and test set, achieving a 99.34% accuracy.

Durmus *et al.* (2017) used tomato leaf images from the PlantVillage dataset, consisting of ten different classes, including healthy tomato leaves. The study implemented CNNs, both AlexNet and SqueezeNet. It was found that despite AlexNet marginally outperforming SqueezeNet in terms of classification accuracy, it required approximately ten times the model size and three times the inference time. Also, the paper suggested that SqueezeNet can be used for mobile deep-learning classification. AlexNet achieved an accuracy of 95.65%, whereas SqueezeNet attained an accuracy of 94.30%.

Liu *et al.* (2017) inspired by GoogLeNet and ResNet-20 in order to classify 13,689 apple leaf diseases. The apple images were gathered from China. The suggested model showed an accuracy of 97.62%, and compared to the AlexNet model, the suggested model significantly reduced the parameter count and converges faster.

Tan *et al.* (2016) used a method involving the use of CNNs to identify apple pathological images was introduced. An accuracy of up to 96.08% is achieved for this approach, along with a notably fast convergence rate.

Karthik *et al.* (2020) used a PlantVillage dataset with 120,000 images. The dataset was divided into tomato early blight, late blight, and leaf mold. They proposed an attention-based deep residual network in order to detect tomato leaf infection. In the 5-fold cross-validation, the proposed model achieved an accuracy of 98% on the validation sets.

Waheed *et al.* (2020) used an efficient maize leaf recognition model based on an optimized DenseNet with fewer parameters. The proposed model achieved an accuracy of 98.06%. It used fewer parameters than many CNN models, such as EfficientNet, VGG19Net, and NASNet.

Huang *et al.* (2019) used the AI Challenger dataset with eight plant species: tomato, strawberry, citrus, pepper, potato, corn, apple, grape, peach, and nineteen plant diseases. The suggested model was comprised of two sub-models. The first was a leaf segmentation model, which used a U-Net to distinguish leaves from the background. The second sub-model, the two-headed network, was a plant disease classification model. This network classified plant diseases using feature extraction from some CNNs. The proposed model achieved 98.07% in plant classification and 87.45% accuracy in disease recognition.

Sethy *et al.* (2020) used a dataset with 5932 on-field images of four types of rice leaf diseases, including bacterial blight, blast, brown spot, and tungro. 11 CNN models were examined, and a combination of a ResNet50 model with an SVM classifier, achieved an accuracy of 98.38%.

Adedoja *et al.* (2019) used a CNN architecture on NASNet, which was trained and tested with the PlantVillage dataset. These images showed different infection locations on the plants. The model demonstrated an accuracy rate of 93.82%.

Zhao *et al.* (2022) used CNN and a spatial attention mechanism to get a 95.20% success rate for the three potato diseases in their web-based real-time plant disease detection system.

DeChant *et al.* (2017) used a computational pipeline of CNNs, feeding a dataset of 1,028 images showing infected leaves and 768 images of non-infected leaves. The method achieved an accuracy of 96.7%.

Ma *et al.* (2018) used PlantVillage with different augmentation techniques and a deep convolutional neural network to get an accuracy of 93.4%.

Khan *et al.* (2022) proposed an automated framework for the classification of cucumber leaf diseases. The team used deep learning and optimal feature selection techniques in order to achieve an accuracy of 98.4%.

Sladojevic *et al.* (2016) used last-generation CNNs in order to detect thirteen different plant diseases. The proposed model is for identifying diseases in various crops. They utilized a dataset containing 33,000 images and achieved an accuracy of 96.37%.

## 3. Dataset

In this study a total of 11,000 images of tomato leaves were used, which are among the open-access PlanVillage dataset (Hughes and Salathe, 2015). Some of the sample images can be seen in Figures 1 to 3.

**Table 3: Details of Classes in the Dataset**

| Class Name | Scientific Name | Training | Test | Total |
|---|---|---|---|---|
| Healthy | - | 900 | 200 | 1100 |
| Tomato Mosaic Virus | *Tomato Mosaic Virus* | 900 | 200 | 1100 |
| Tomato Yellow Curl Leaf Virus | *Begomovirus* | 900 | 200 | 1100 |
| Target spot | *Corynespora cassiicola* | 900 | 200 | 1100 |
| Spider Mites | *Tetranychus urticae* | 900 | 200 | 1100 |
| Septoria leaf spot | *Septoria lycopersici* | 900 | 200 | 1100 |
| Leaf mold | *Fulvia fulva* | 900 | 200 | 1100 |
| Late blight | *Phytophthora infestans* | 900 | 200 | 1100 |
| Early blight | *Alternaria solani* | 900 | 200 | 1100 |
| Bacterial spot | *Phytophthora infestans* | 900 | 200 | 1100 |

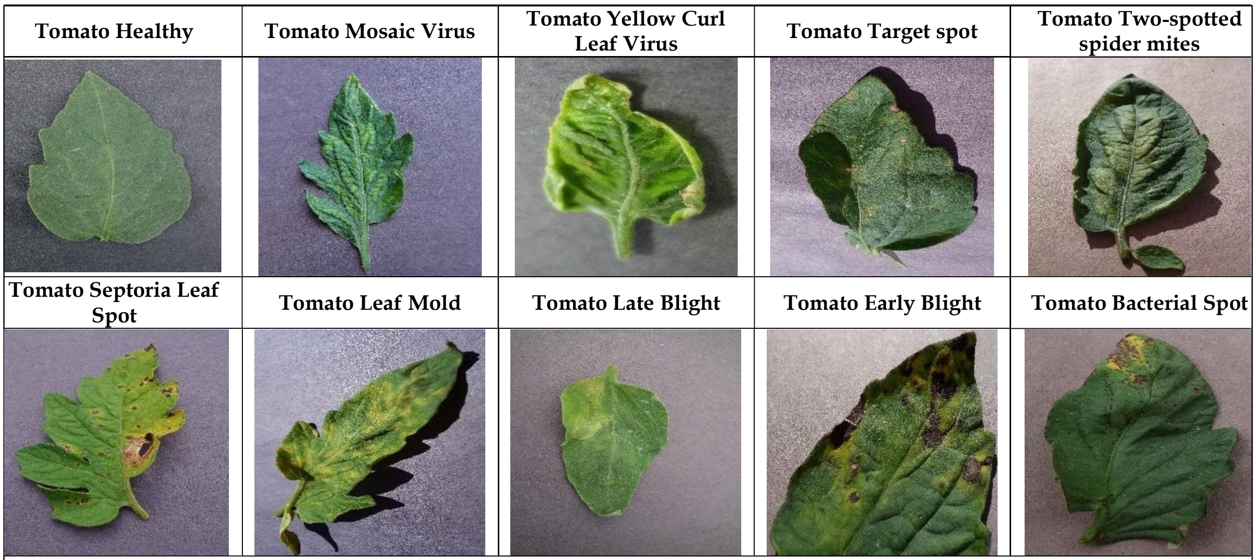


**Figure 1: Color-Level Sample Tomato Leaves in the Dataset**

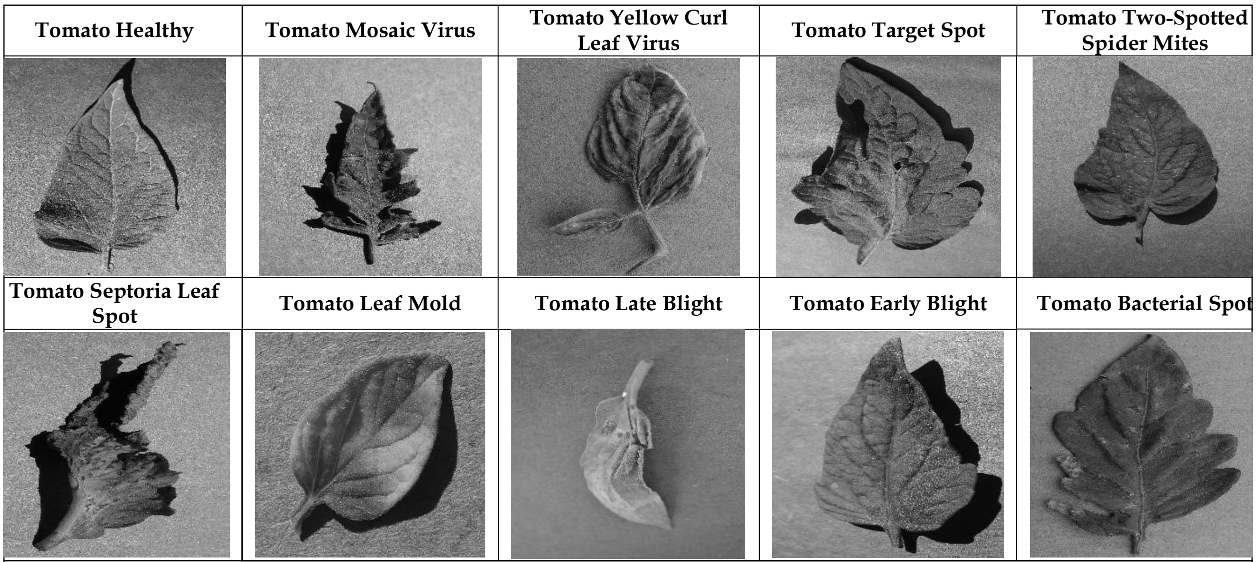


**Figure 2: Grey-Scale Sample Tomato Leaves**

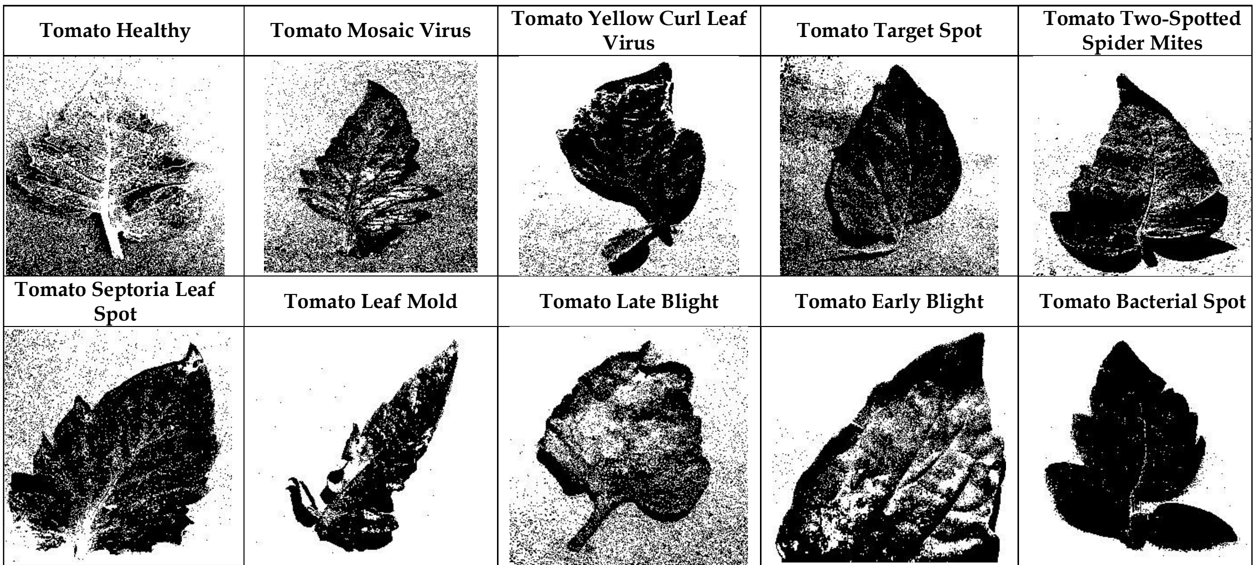


**Figure 3: Binarized Sample Tomato Leaves**

## 4. EfficientNet Architectures

EfficientNet is a pre-trained CNN, which has been used in this paper. EfficientNet is a CNN architecture and a scaling technique that scales depth, width, and resolution in a uniform manner using a compound coefficient for convolutional architectures (Figure 4). With an impressive 84.3% top-1 accuracy on ImageNet, EfficientNet-B7 outperforms the best existing CNN by being 8.4 times more compact and 6.1 times faster during inference (Tan and Le, 2019; Eroltu, 2023). EfficientNet-B1 is 7.6 times more compact and 5.7 times quicker than ResNet-152 (Tan and Le, 2019; Eroltu, 2023). Tan *et al*. (2019) proposed a straightforward but efficient method of compound scaling (Tan and Le, 2019).

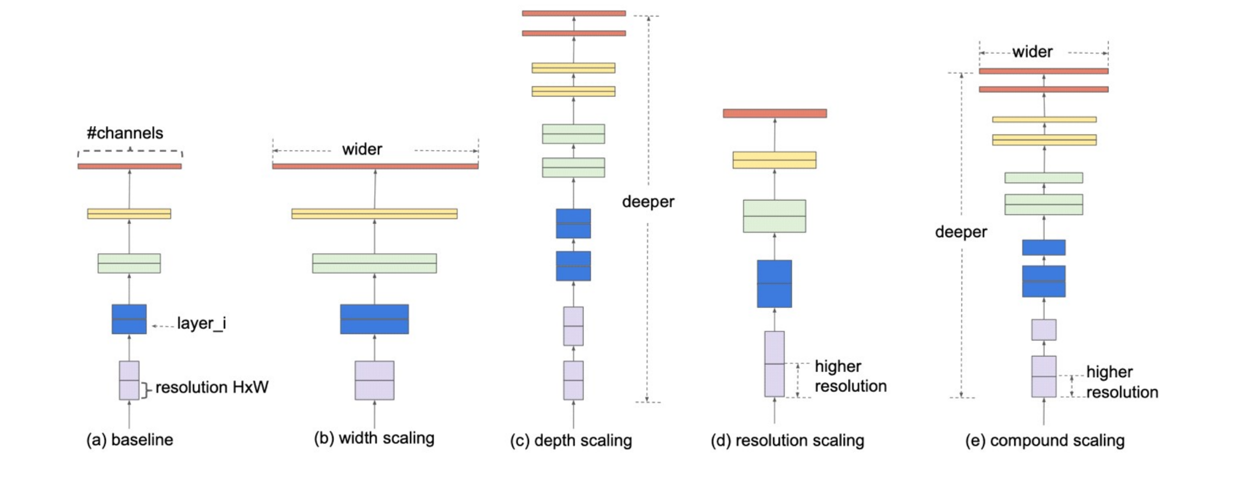


**Figure 4: EfficientNet Model Scaling (Tan and Le, 2019)**

**Note:** (a) serves as an example of a baseline network. (b) to (d) represent conventional scaling techniques that solely increase one dimension, be it network width, depth, or resolution. (e) introduces a compound scaling approach that consistently scales all three dimensions using a set ratio (Tan and Le, 2019). The backbone of this experiment, as earlier stated, is EfficientNet.

## 5. Methods

The hardware used for this experiment has a 2,4 GHz Quad-Core Intel Core i5, 8 GB RAM 2133 MHz LPDDR3, and Tesla P100-PCIE GPU. Experiments have been performed using Python code. The grayscale, color-level, and binarized images in the dataset were independently fed into the pre-trained EfficientNets (EfficientNet0 to EfficientNet7). For each of the three datasets, performance results were obtained using softmax, used as an activation function. Softmax is a function that transforms a K-dimensional vector z with any real numbers into a K-dimensional vector with real values between 0 and 1 that sum to 1. The softmax function transforms a small or negative input into a low probability, while a large input is converted into a high probability. The results always remain between 0 and 1. The mathematical expression for softmax is:

$$softmax(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}}$$

All eight different EfficientNet models were tested with different Epochs (20, 30, 40, 50) and Batch sizes (16, 32, 64, 128) . A total of 384 test results were obtained. 128 test results, on the other hand, were reported because the accuracies of binarized and gray-scale datasets were lower than color-level datasets.

Three different datasets were separately given in this study, as it was mentioned. Images were resized as the images in the dataset were 256x256 pixels. A horizontal flip to the input image along its vertical axis was applied in order to enhance the diversity of the training data. The data was rotated between -18 and +18 degrees as it helped the model to become more robust to slight differences in orientation. Randomized zoom is vital in enhancing the model's capability to interpret leaf images at diverse distances and scales. Moreover, the backgrounds of all the images were cleared in order to increase the accuracy of the model.

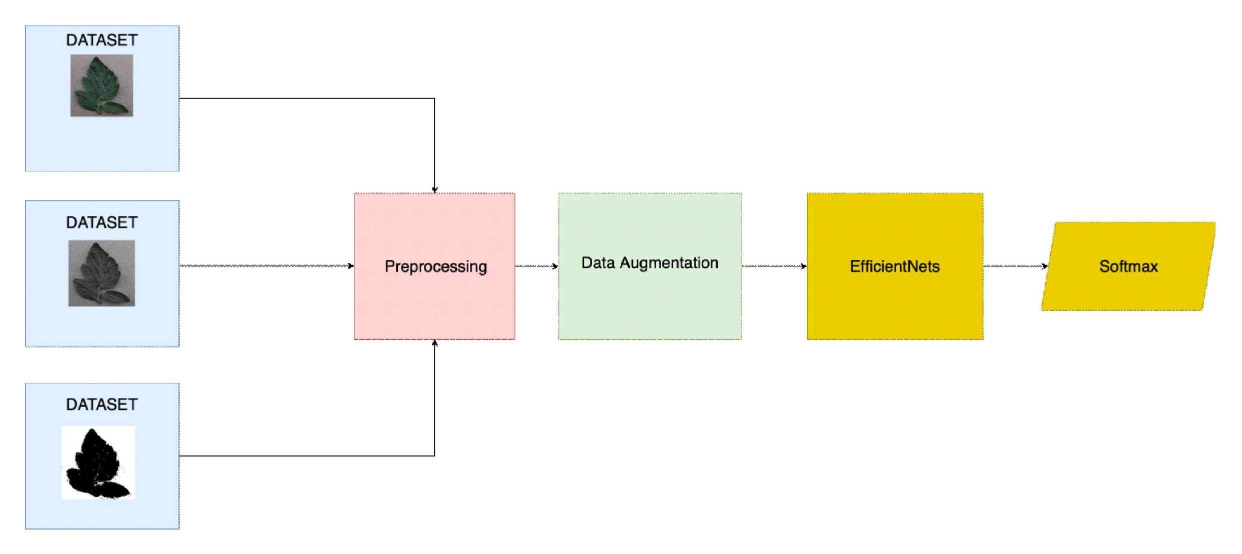


**Figure 5: Proposed Model for this Study**

## 6. Model Performance Metrics

Performance metrics have been used to deduce the reliability of the study. The performance matrices that are used in this study:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

$$Precision = \frac{TN}{TN + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F - score = \frac{2TP}{2TP + FP + FN}$$

The accuracy metric is used to represent the data's overall classification performance. The precision is used in order to express the success of finding negative samples, while the recall metric measures the success of finding positive samples. The f-score metric calculates the harmonic mean of precision and sensitivity, providing a balanced measure of their combined performance.

## 7. Results and Discussion

As mentioned in the experiment section before, the gray-scale and binarized datasets were not reported in this study due to their lower accuracies compared to the color-level dataset.

In Table 4, performance metrics are obtained from pre-trained EfficientNet models. The highest training accuracy has been achieved using EfficientNetB3, with an epoch count of 50 and a batch size of 128, resulting in an accuracy of 99.66%. The second highest accuracy has been recorded in EfficientNetB4, with an epoch count of 30 and a batch size of 128, resulting in an accuracy of 99.64%. The pre-trained EfficientNet6 ranked third in accuracy, with 40 epochs and a batch size of 64, achieving an accuracy of 99.62%. The highest scores in terms of accuracy, precision, and *F*-score have also been obtained with the EfficientNetB3. The lowest accuracy, precision, and f-score have been noted in EfficientNetB0, as we would expect. The experiment for EfficientNetB7 with a batch size of 128 couldn't be carried out because the computer's CPU was inadequate.

In Table 5, performance metrics obtained from pre-traaboined EfficientNet models and all of the models were compared in a table. The highest test accuracy has been achieved using EfficientNetB7, with an epoch

**Table 4: Training of EfficientNet Models and Performance Evaluations**

| Pre-trained CNN Model | EPOCH | Batch Size | Accuracy (%) | Precision (%) | Recall (%) | *F*-score (%) |
|---|---|---|---|---|---|---|
| EfficientNetB0 | 20 | 16 | 97.17 | 97.37 | 96.90 | 97.13 |
| EfficientNetB0 | 30 | 16 | 97.90 | 98.04 | 97.87 | 97.95 |
| EfficientNetB0 | 40 | 16 | 98.25 | 98.34 | 98.16 | 98.25 |
| EfficientNetB0 | 50 | 16 | 98.61 | 98.67 | 98.57 | 98.62 |
| EfficientNetB0 | 20 | 32 | 98.84 | 98.94 | 98.74 | 98.84 |
| EfficientNetB0 | 30 | 32 | 98.41 | 98.47 | 98.34 | 98.40 |
| EfficientNetB0 | 40 | 32 | 99.07 | 99.12 | 99.05 | 99.08 |
| EfficientNetB0 | 50 | 32 | 98.98 | 99.04 | 98.95 | 99.00 |
| EfficientNetB0 | 20 | 64 | 98.81 | 98.91 | 98.74 | 98.82 |
| EfficientNetB0 | 30 | 64 | 98.97 | 99.01 | 98.92 | 98.96 |
| EfficientNetB0 | 40 | 64 | 99.22 | 99.25 | 99.17 | 99.21 |
| EfficientNetB0 | 50 | 64 | 98.86 | 98.90 | 98.86 | 98.88 |
| EfficientNetB0 | 20 | 128 | 99.40 | 99.43 | 99.36 | 99.40 |
| EfficientNetB0 | 30 | 128 | 99.57 | 99.60 | 99.51 | 99.55 |
| EfficientNetB0 | 40 | 128 | 99.54 | 99.58 | 99.53 | 99.55 |
| EfficientNetB0 | 50 | 128 | 99.52 | 99.53 | 99.51 | 99.52 |
| EfficientNetB1 | 20 | 16 | 98.17 | 98.33 | 98.07 | 98.20 |
| EfficientNetB1 | 30 | 16 | 98.01 | 98.23 | 97.91 | 98.07 |
| EfficientNetB1 | 40 | 16 | 98.48 | 98.68 | 98.44 | 98.56 |
| EfficientNetB1 | 50 | 16 | 98.45 | 98.58 | 98.27 | 98.42 |
| EfficientNetB1 | 20 | 32 | 97.51 | 97.77 | 97.36 | 97.56 |
| EfficientNetB1 | 30 | 32 | 98.00 | 98.11 | 97.93 | 98.02 |
| EfficientNetB1 | 40 | 32 | 98.38 | 98.57 | 98.29 | 98.43 |
| EfficientNetB1 | 50 | 32 | 99.27 | 99.31 | 99.25 | 99.28 |
| EfficientNetB1 | 20 | 64 | 98.97 | 99.02 | 98.93 | 98.97 |
| EfficientNetB1 | 30 | 64 | 98.69 | 98.72 | 98.59 | 98.65 |
| EfficientNetB1 | 40 | 64 | 99.50 | 99.50 | 99.36 | 99.43 |
| EfficientNetB1 | 50 | 64 | 99.21 | 99.22 | 99.16 | 99.19 |
| EfficientNetB1 | 20 | 128 | 99.38 | 99.38 | 99.36 | 99.37 |
| EfficientNetB1 | 30 | 128 | 99.28 | 99.34 | 99.28 | 99.31 |
| EfficientNetB1 | 40 | 128 | 99.38 | 99.39 | 99.34 | 99.36 |
| EfficientNetB1 | 50 | 128 | 99.21 | 99.24 | 99.19 | 99.21 |
| EfficientNetB2 | 20 | 16 | 97.67 | 97.90 | 97.43 | 97.66 |

Table 4 (Cont.)

| | | | | | | |
|---|---|---|---|---|---|---|
| EfficientNetB2 | 30 | 16 | 97.99 | 98.19 | 97.92 | 98.05 |
| EfficientNetB2 | 40 | 16 | 98.97 | 99.06 | 98.91 | 98.98 |
| EfficientNetB2 | 50 | 16 | 98.36 | 98.56 | 98.20 | 98.38 |
| EfficientNetB2 | 20 | 32 | 97.60 | 97.64 | 97.52 | 97.58 |
| EfficientNetB2 | 30 | 32 | 98.54 | 98.65 | 98.48 | 98.56 |
| EfficientNetB2 | 40 | 32 | 99.00 | 99.01 | 98.87 | 98.94 |
| EfficientNetB2 | 50 | 32 | 99.13 | 99.19 | 99.12 | 99.15 |
| EfficientNetB2 | 20 | 64 | 98.30 | 98.52 | 98.24 | 98.38 |
| EfficientNetB2 | 30 | 64 | 98.48 | 98.54 | 98.47 | 98.50 |
| EfficientNetB2 | 40 | 64 | 99.42 | 99.42 | 99.42 | 99.42 |
| EfficientNetB2 | 50 | 64 | 99.56 | 99.58 | 99.56 | 99.57 |
| EfficientNetB2 | 20 | 128 | 99.23 | 99.26 | 99.21 | 99.23 |
| EfficientNetB2 | 30 | 128 | 99.34 | 99.36 | 99.29 | 99.32 |
| EfficientNetB2 | 40 | 128 | 99.58 | 99.58 | 99.57 | 99.57 |
| EfficientNetB2 | 50 | 128 | 99.22 | 99.23 | 99.14 | 99.18 |
| EfficientNetB3 | 20 | 16 | 97.30 | 97.63 | 97.01 | 97.32 |
| EfficientNetB3 | 30 | 16 | 97.94 | 98.10 | 97.77 | 97.93 |
| EfficientNetB3 | 40 | 16 | 99.03 | 99.07 | 98.97 | 99.02 |
| EfficientNetB3 | 50 | 16 | 98.51 | 98.63 | 98.49 | 98.56 |
| EfficientNetB3 | 20 | 32 | 98.09 | 98.24 | 98.01 | 98.12 |
| EfficientNetB3 | 30 | 32 | 98.50 | 98.63 | 98.40 | 98.51 |
| EfficientNetB3 | 40 | 32 | 98.72 | 98.86 | 98.67 | 98.76 |
| EfficientNetB3 | 50 | 32 | 99.06 | 99.07 | 99.05 | 99.06 |
| EfficientNetB3 | 20 | 64 | 98.63 | 98.82 | 98.53 | 98.67 |
| EfficientNetB3 | 30 | 64 | 98.99 | 99.09 | 98.97 | 99.03 |
| EfficientNetB3 | 40 | 64 | 99.61 | 99.61 | 99.61 | 99.61 |
| EfficientNetB3 | 50 | 64 | 99.42 | 99.45 | 99.38 | 99.41 |
| EfficientNetB3 | 20 | 128 | 99.09 | 99.16 | 99.00 | 99.08 |
| EfficientNetB3 | 30 | 128 | 99.61 | 99.65 | 99.61 | 99.63 |
| EfficientNetB3 | 40 | 128 | 99.24 | 99.34 | 99.20 | 99.27 |
| EfficientNetB3 | 50 | 128 | 99.66 | 99.67 | 99.63 | 99.65 |
| EfficientNetB4 | 20 | 16 | 98.48 | 98.59 | 98.24 | 98.41 |
| EfficientNetB4 | 30 | 16 | 98.09 | 98.20 | 98.01 | 98.10 |
| EfficientNetB4 | 40 | 16 | 98.72 | 98.82 | 98.58 | 98.70 |
| EfficientNetB4 | 50 | 16 | 98.87 | 99.04 | 98.84 | 98.94 |
| EfficientNetB4 | 20 | 32 | 98.59 | 98.73 | 98.52 | 98.62 |
| EfficientNetB4 | 30 | 32 | 98.37 | 98.48 | 98.28 | 98.38 |
| EfficientNetB4 | 40 | 32 | 98.77 | 98.86 | 98.75 | 98.80 |
| EfficientNetB4 | 50 | 32 | 99.61 | 99.68 | 99.56 | 99.62 |
| EfficientNetB4 | 20 | 64 | 98.95 | 99.01 | 98.87 | 98.94 |
| EfficientNetB4 | 30 | 64 | 99.39 | 99.43 | 99.38 | 99.40 |

| Table 4 (Cont.) | | | | | | |
|---|---|---|---|---|---|---|
| EfficientNetB4 | 40 | 64 | 99.41 | 99.43 | 99.36 | 99.40 |
| EfficientNetB4 | 50 | 64 | 98.96 | 99.14 | 98.89 | 99.01 |
| EfficientNetB4 | 20 | 128 | 98.89 | 99.06 | 98.98 | 99.02 |
| EfficientNetB4 | 30 | 128 | 99.64 | 99.64 | 99.64 | 99.64 |
| EfficientNetB4 | 40 | 128 | 99.18 | 99.21 | 99.12 | 99.16 |
| EfficientNetB4 | 50 | 128 | 99.39 | 99.40 | 99.34 | 99.37 |
| EfficientNetB5 | 20 | 16 | 97.41 | 97.67 | 97.23 | 97.45 |
| EfficientNetB5 | 30 | 16 | 98.37 | 98.48 | 98.24 | 98.36 |
| EfficientNetB5 | 40 | 16 | 99.11 | 99.16 | 99.06 | 99.11 |
| EfficientNetB5 | 50 | 16 | 99.14 | 99.22 | 99.03 | 99.12 |
| EfficientNetB5 | 20 | 32 | 98.15 | 98.32 | 98.05 | 98.18 |
| EfficientNetB5 | 30 | 32 | 98.48 | 98.56 | 98.45 | 98.50 |
| EfficientNetB5 | 40 | 32 | 99.03 | 99.09 | 99.01 | 99.05 |
| EfficientNetB5 | 50 | 32 | 99.32 | 99.37 | 99.30 | 99.33 |
| EfficientNetB5 | 20 | 64 | 98.91 | 98.97 | 98.80 | 98.88 |
| EfficientNetB5 | 30 | 64 | 98.42 | 98.56 | 98.29 | 98.42 |
| EfficientNetB5 | 40 | 64 | 99.35 | 99.39 | 99.28 | 99.33 |
| EfficientNetB5 | 50 | 64 | 99.05 | 99.07 | 99.03 | 99.05 |
| EfficientNetB5 | 20 | 128 | 99.05 | 99.16 | 99.01 | 99.08 |
| EfficientNetB5 | 30 | 128 | 99.35 | 99.41 | 99.32 | 99.36 |
| EfficientNetB5 | 40 | 128 | 99.53 | 99.53 | 99.53 | 99.53 |
| EfficientNetB5 | 50 | 128 | 99.81 | 99.89 | 99.81 | 99.85 |
| EfficientNetB6 | 20 | 16 | 97.96 | 98.21 | 97.76 | 97.98 |
| EfficientNetB6 | 30 | 16 | 98.56 | 98.68 | 98.38 | 98.53 |
| EfficientNetB6 | 40 | 16 | 98.85 | 98.89 | 98.80 | 98.84 |
| EfficientNetB6 | 50 | 16 | 99.26 | 99.34 | 99.12 | 99.23 |
| EfficientNetB6 | 20 | 32 | 98.25 | 98.57 | 98.19 | 98.38 |
| EfficientNetB6 | 30 | 32 | 99.09 | 99.24 | 99.00 | 99.12 |
| EfficientNetB6 | 40 | 32 | 99.02 | 99.08 | 98.99 | 99.03 |
| EfficientNetB6 | 50 | 32 | 98.80 | 98.89 | 98.78 | 98.83 |
| EfficientNetB6 | 20 | 64 | 98.96 | 99.04 | 98.88 | 98.96 |
| EfficientNetB6 | 30 | 64 | 98.90 | 98.97 | 98.82 | 98.90 |
| EfficientNetB6 | 40 | 64 | 99.62 | 99.65 | 99.59 | 99.62 |
| EfficientNetB6 | 50 | 64 | 99.06 | 99.14 | 99.01 | 99.07 |
| EfficientNetB6 | 20 | 128 | 99.34 | 99.35 | 99.30 | 99.32 |
| EfficientNetB6 | 30 | 128 | 99.48 | 99.49 | 99.47 | 99.48 |
| EfficientNetB6 | 40 | 128 | 99.50 | 99.54 | 99.48 | 99.51 |
| EfficientNetB6 | 50 | 128 | 99.60 | 99.60 | 99.60 | 99.60 |
| EfficientNetB7 | 20 | 16 | 97.50 | 97.78 | 97.13 | 97.45 |
| EfficientNetB7 | 30 | 16 | 98.33 | 98.44 | 98.25 | 98.34 |
| EfficientNetB7 | 40 | 16 | 98.71 | 98.84 | 98.64 | 98.74 |

| Table 4 (Cont.) | | | | | | |
|---|---|---|---|---|---|---|
| EfficientNetB7 | 50 | 16 | 98.38 | 99.47 | 99.38 | 98.92 |
| EfficientNetB7 | 20 | 32 | 98.11 | 98.32 | 97.96 | 98.14 |
| EfficientNetB7 | 30 | 32 | 98.54 | 98.72 | 98.48 | 98.60 |
| EfficientNetB7 | 40 | 32 | 99.18 | 99.29 | 99.14 | 99.21 |
| EfficientNetB7 | 50 | 32 | 99.45 | 99.45 | 99.45 | 99.45 |
| EfficientNetB7 | 20 | 64 | 98.08 | 98.15 | 97.88 | 98.01 |
| EfficientNetB7 | 30 | 64 | 98.78 | 98.82 | 98.76 | 98.79 |
| EfficientNetB7 | 40 | 64 | 99.01 | 99.09 | 98.93 | 99.01 |
| EfficientNetB7 | 50 | 64 | 99.42 | 99.50 | 99.42 | 99.46 |
| EfficientNetB7 | 20 | 128 | - | - | - | - |
| EfficientNetB7 | 30 | 128 | - | - | - | - |
| EfficientNetB7 | 40 | 128 | - | - | - | - |
| EfficientNetB7 | 50 | 128 | - | - | - | - |

**Table 5: Test Table**

| Pre-trained CNN Model | EPOCH | Batch Size | Accuracy (%) | Precision (%) | Recall (%) | *F*-score (%) |
|---|---|---|---|---|---|---|
| EfficientNetB0 | 20 | 16 | 92.95 | 93.71 | 92.55 | 93.13 |
| EfficientNetB0 | 30 | 16 | 85.54 | 86.06 | 85.34 | 85.70 |
| EfficientNetB0 | 40 | 16 | 88.90 | 89.37 | 88.54 | 88.95 |
| EfficientNetB0 | 50 | 16 | 92.83 | 92.97 | 92.67 | 92.82 |
| EfficientNetB0 | 20 | 32 | 89.50 | 89.85 | 89.02 | 89.43 |
| EfficientNetB0 | 30 | 32 | 83.80 | 84.08 | 83.40 | 83.74 |
| EfficientNetB0 | 40 | 32 | 84.58 | 85.14 | 84.01 | 84.57 |
| EfficientNetB0 | 50 | 32 | 92.03 | 92.23 | 91.75 | 91.99 |
| EfficientNetB0 | 20 | 64 | 89.02 | 89.35 | 88.78 | 89.06 |
| EfficientNetB0 | 30 | 64 | 84.24 | 85.20 | 83.52 | 84.35 |
| EfficientNetB0 | 40 | 64 | 85.88 | 86.11 | 85.68 | 85.90 |
| EfficientNetB0 | 50 | 64 | 94.59 | 94.69 | 94.39 | 94.54 |
| EfficientNetB0 | 20 | 128 | 89.14 | 89.88 | 88.69 | 89.28 |
| EfficientNetB0 | 30 | 128 | 93.13 | 93.54 | 92.85 | 93.20 |
| EfficientNetB0 | 40 | 128 | 95.81 | 96.15 | 95.48 | 95.81 |
| EfficientNetB0 | 50 | 128 | 95.11 | 95.37 | 94.90 | 95.13 |

| Table 5 (Cont.) | | | | | | |
|---|---|---|---|---|---|---|
| EfficientNetB1 | 20 | 16 | 94.27 | 94.72 | 94.07 | 94.40 |
| EfficientNetB1 | 30 | 16 | 95.55 | 96.01 | 95.47 | 95.74 |
| EfficientNetB1 | 40 | 16 | 89.42 | 89.78 | 89.06 | 89.42 |
| EfficientNetB1 | 50 | 16 | 89.02 | 89.62 | 88.58 | 89.10 |
| EfficientNetB1 | 20 | 32 | 92.99 | 93.45 | 92.59 | 93.02 |
| EfficientNetB1 | 30 | 32 | 89.34 | 90.30 | 88.78 | 89.53 |
| EfficientNetB1 | 40 | 32 | 91.59 | 92.18 | 91.11 | 91.64 |
| EfficientNetB1 | 50 | 32 | 87.46 | 88.27 | 86.86 | 87.56 |
| EfficientNetB1 | 20 | 64 | 90.99 | 91.95 | 90.67 | 91.30 |
| EfficientNetB1 | 30 | 64 | 95.23 | 95.64 | 94.95 | 95.30 |
| EfficientNetB1 | 40 | 64 | 88.30 | 88.77 | 88.06 | 88.53 |
| EfficientNetB1 | 50 | 64 | 91.51 | 91.90 | 91.35 | 91.62 |
| EfficientNetB1 | 20 | 128 | 89.97 | 90.38 | 89.64 | 90.00 |
| EfficientNetB1 | 30 | 128 | 93.79 | 94.27 | 93.34 | 93.80 |
| EfficientNetB1 | 40 | 128 | 94.65 | 94.96 | 94.49 | 94.72 |
| EfficientNetB1 | 50 | 128 | 89.88 | 90.49 | 89.64 | 90.06 |
| EfficientNetB2 | 20 | 16 | 92.47 | 92.96 | 92.07 | 92.51 |
| EfficientNetB2 | 30 | 16 | 91.51 | 91.93 | 91.27 | 91.60 |
| EfficientNetB2 | 40 | 16 | 84.42 | 85.25 | 83.81 | 84.52 |
| EfficientNetB2 | 50 | 16 | 91.43 | 91.83 | 90.91 | 91.37 |
| EfficientNetB2 | 20 | 32 | 92.31 | 92.94 | 91.83 | 92.38 |
| EfficientNetB2 | 30 | 32 | 83.69 | 84.14 | 83.13 | 83.63 |
| EfficientNetB2 | 40 | 32 | 88.34 | 89.17 | 87.74 | 88.45 |
| EfficientNetB2 | 50 | 32 | 94.87 | 95.39 | 94.47 | 94.93 |
| EfficientNetB2 | 20 | 64 | 86.50 | 87.60 | 85.77 | 86.68 |
| EfficientNetB2 | 30 | 64 | 92.72 | 93.10 | 92.44 | 92.77 |
| EfficientNetB2 | 40 | 64 | 95.13 | 95.24 | 94.89 | 95.06 |
| EfficientNetB2 | 50 | 64 | 90.96 | 91.23 | 90.73 | 90.98 |
| EfficientNetB2 | 20 | 128 | 91.20 | 91.65 | 91.12 | 91.38 |
| EfficientNetB2 | 30 | 128 | 93.71 | 93.90 | 93.67 | 93.78 |
| EfficientNetB2 | 40 | 128 | 95.15 | 95.37 | 94.90 | 95.13 |
| EfficientNetB2 | 50 | 128 | 92.80 | 93.41 | 92.64 | 93.02 |
| EfficientNetB3 | 20 | 16 | 90.83 | 91.20 | 90.46 | 90.83 |
| EfficientNetB3 | 30 | 16 | 86.34 | 86.34 | 86.81 | 86.57 |
| EfficientNetB3 | 40 | 16 | 87.30 | 87.93 | 86.94 | 87.43 |

| Table 5 (Cont.) | | | | | | |
|---|---|---|---|---|---|---|
| EfficientNetB3 | 50 | 16 | 87.70 | 88.60 | 87.46 | 88.03 |
| EfficientNetB3 | 20 | 32 | 81.85 | 82.98 | 81.09 | 84.68 |
| EfficientNetB3 | 30 | 32 | 86.22 | 86.67 | 85.74 | 86.20 |
| EfficientNetB3 | 40 | 32 | 93.35 | 93.75 | 93.15 | 93.45 |
| EfficientNetB3 | 50 | 32 | 90.10 | 90.40 | 89.78 | 90.09 |
| EfficientNetB3 | 20 | 64 | 90.50 | 90.86 | 89.66 | 90.26 |
| EfficientNetB3 | 30 | 64 | 91.11 | 92.25 | 90.58 | 91.40 |
| EfficientNetB3 | 40 | 64 | 87.74 | 88.25 | 87.54 | 87.90 |
| EfficientNetB3 | 50 | 64 | 88.86 | 89.95 | 88.26 | 89.10 |
| EfficientNetB3 | 20 | 128 | 90.25 | 90.60 | 90.01 | 90.30 |
| EfficientNetB3 | 30 | 128 | 95.07 | 95.29 | 94.86 | 95.07 |
| EfficientNetB3 | 40 | 128 | 94.05 | 94.77 | 93.77 | 94.27 |
| EfficientNetB3 | 50 | 128 | 92.60 | 92.97 | 92.48 | 92.72 |
| EfficientNetB4 | 20 | 16 | 90.91 | 91.59 | 90.79 | 91.19 |
| EfficientNetB4 | 30 | 16 | 87.46 | 88.01 | 86.74 | 87.37 |
| EfficientNetB4 | 40 | 16 | 88.86 | 89.28 | 88.46 | 88.87 |
| EfficientNetB4 | 50 | 16 | 91.43 | 92.16 | 91.35 | 91.75 |
| EfficientNetB4 | 20 | 32 | 77.84 | 79.07 | 77.32 | 78.19 |
| EfficientNetB4 | 30 | 32 | 91.27 | 91.62 | 91.15 | 91.38 |
| EfficientNetB4 | 40 | 32 | 88.14 | 88.54 | 87.94 | 88.24 |
| EfficientNetB4 | 50 | 32 | 92.67 | 92.84 | 92.43 | 92.63 |
| EfficientNetB4 | 20 | 64 | 89.18 | 89.91 | 88.54 | 89.22 |
| EfficientNetB4 | 30 | 64 | 89.42 | 90.17 | 88.94 | 89.55 |
| EfficientNetB4 | 40 | 64 | 90.83 | 91.18 | 90.71 | 90.94 |
| EfficientNetB4 | 50 | 64 | 89.07 | 89.89 | 88.27 | 89.07 |
| EfficientNetB4 | 20 | 128 | 91.16 | 91.57 | 91.12 | 91.34 |
| EfficientNetB4 | 30 | 128 | 95.27 | 95.77 | 94.94 | 95.35 |
| EfficientNetB4 | 40 | 128 | 89.39 | 90.09 | 88.94 | 89.51 |
| EfficientNetB4 | 50 | 128 | 95.97 | 96.13 | 95.89 | 96.01 |
| EfficientNetB5 | 20 | 16 | 87.50 | 88.08 | 87.02 | 87.55 |
| EfficientNetB5 | 30 | 16 | 92.95 | 93.27 | 92.71 | 92.99 |
| EfficientNetB5 | 40 | 16 | 95.19 | 95.44 | 94.83 | 95.13 |
| EfficientNetB5 | 50 | 16 | 96.99 | 97.25 | 96.59 | 96.92 |
| EfficientNetB5 | 20 | 32 | 83.29 | 83.92 | 82.77 | 83.34 |
| EfficientNetB5 | 30 | 32 | 93.87 | 94.26 | 93.47 | 93.86 |

| Table 5 (Cont.) | | | | | | |
|---|---|---|---|---|---|---|
| EfficientNetB5 | 40 | 32 | 90.87 | 91.24 | 90.50 | 90.87 |
| EfficientNetB5 | 50 | 32 | 90.71 | 90.83 | 90.46 | 90.64 |
| EfficientNetB5 | 20 | 64 | 96.51 | 96.86 | 96.27 | 96.56 |
| EfficientNetB5 | 30 | 64 | 94.54 | 95.76 | 93.73 | 94.73 |
| EfficientNetB5 | 40 | 64 | 93.51 | 93.64 | 93.27 | 93.45 |
| EfficientNetB5 | 50 | 64 | 96.55 | 96.70 | 96.31 | 96.50 |
| EfficientNetB5 | 20 | 128 | 92.48 | 92.62 | 92.35 | 92.48 |
| EfficientNetB5 | 30 | 128 | 86.10 | 86.54 | 85.90 | 86.22 |
| EfficientNetB5 | 40 | 128 | 96.63 | 96.86 | 96.46 | 96.66 |
| EfficientNetB5 | 50 | 128 | 97.12 | 96.86 | 96.92 | 96.89 |
| EfficientNetB6 | 20 | 16 | 89.82 | 90.46 | 88.94 | 89.70 |
| EfficientNetB6 | 30 | 16 | 92.99 | 93.35 | 92.79 | 93.07 |
| EfficientNetB6 | 40 | 16 | 95.03 | 95.53 | 94.95 | 95.24 |
| EfficientNetB6 | 50 | 16 | 97.28 | 97.43 | 97.04 | 97.23 |
| EfficientNetB6 | 20 | 32 | 84.66 | 85.15 | 84.29 | 84.72 |
| EfficientNetB6 | 30 | 32 | 92.63 | 93.00 | 92.55 | 92.77 |
| EfficientNetB6 | 40 | 32 | 94.51 | 94.94 | 93.99 | 94.46 |
| EfficientNetB6 | 50 | 32 | 93.79 | 94.26 | 93.51 | 93.88 |
| EfficientNetB6 | 20 | 64 | 87.50 | 88.22 | 87.02 | 87.62 |
| EfficientNetB6 | 30 | 64 | 93.51 | 93.91 | 93.23 | 93.57 |
| EfficientNetB6 | 40 | 64 | 94.35 | 94.53 | 94.19 | 94.36 |
| EfficientNetB6 | 50 | 64 | 93.31 | 93.49 | 93.23 | 93.36 |
| EfficientNetB6 | 20 | 128 | 85.86 | 86.49 | 85.57 | 86.03 |
| EfficientNetB6 | 30 | 128 | 90.17 | 90.37 | 89.93 | 90.15 |
| EfficientNetB6 | 40 | 128 | 93.79 | 93.89 | 93.59 | 93.74 |
| EfficientNetB6 | 50 | 128 | 94.09 | 94.27 | 93.92 | 94.10 |
| EfficientNetB7 | 20 | 16 | 93.39 | 93.93 | 93.07 | 93.50 |
| EfficientNetB7 | 30 | 16 | 95.03 | 95.22 | 95.03 | 95.12 |
| EfficientNetB7 | 40 | 16 | 88.34 | 89.05 | 87.94 | 88.50 |
| EfficientNetB7 | 50 | 16 | 87.06 | 88.00 | 86.34 | 87.16 |
| EfficientNetB7 | 20 | 32 | 89.14 | 89.66 | 88.62 | 89.14 |
| EfficientNetB7 | 30 | 32 | 91.19 | 91.87 | 91.03 | 91.45 |
| EfficientNetB7 | 40 | 32 | 94.07 | 94.41 | 93.99 | 94.20 |
| EfficientNetB7 | 50 | 32 | 92.51 | 93.13 | 92.35 | 92.74 |
| EfficientNetB7 | 20 | 64 | 94.83 | 95.27 | 94.51 | 94.89 |
| EfficientNetB7 | 30 | 64 | 93.50 | 93.97 | 93.26 | 93.61 |
| EfficientNetB7 | 40 | 64 | 92.11 | 92.47 | 91.55 | 92.00 |

count of 50 and a batch size of 64, resulting in an accuracy of 97.83%. The second highest accuracy has been recorded in EfficientNetB6, with an epoch count of 50 and a batch size 16, resulting in an accuracy of 97.28%. The pre-trained EfficientNet6 ranked third in accuracy, with 50 epochs and a batch size of 128, achieving an accuracy of 97.12%. These three models were very successful compared to EfficientNetB3.

The accuracy performances have been demonstrated in the Figure 6. The best Efficient models have been taken from the test table.

The Figure 6 shows the accuracy of the most successful EfficientNet models. It may be interesting to see that EfficientNetB3 performed the poorest out of all of the EfficientNets as EfficientNetB3 is more recent. There are plausible reasons why this happened. First of all, hyperparameters, such as learning rate, weight decay, and data augmentation strategies, might have affected the performance of neural networks. Furthermore, regularization and dropout strength influenced the models, preventing overfitting. As known, data augmentation plays a critical role in deep learning models/architectures. Another factor can be the choice of optimizer and the receptive field of the model, learning models/architectures. Another factor can be the choice of optimizer and the receptive field of the model.

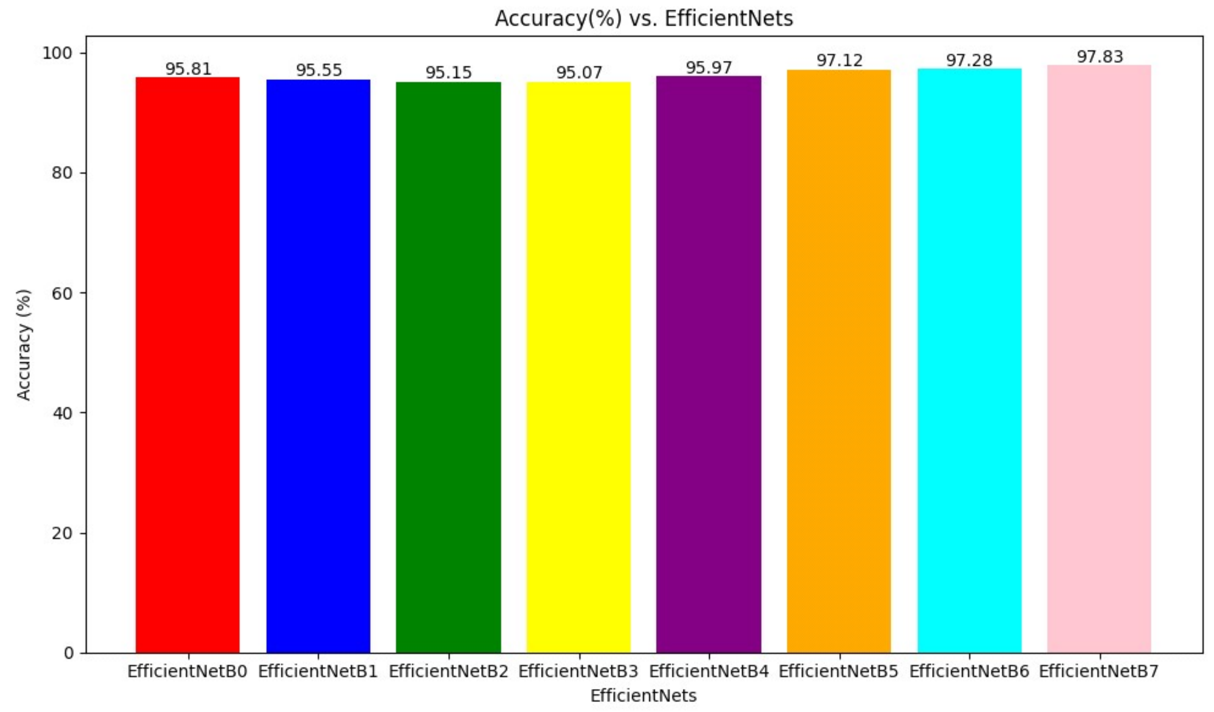


**Figure 6: Accuracy of all EfficientNets**

Table 6 compares this study with others in the literature. The proposed method yields higher accuracy than most of the existing techniques. The pre-trained model attained an impressive accuracy of 97.83%, surpassing the results of many other studies on this subject. This study proposes an acceptable model.

**Table 6: Comparison of Similar Studies in the Literature**

| Study | Model | Accuracy |
|---|---|---|
| This study | EfficientNetB7 | 97.83% |
| Wspanialy *et al.* | ResNet-50 | 97.00% |
| Abbas *et al.* | U-Net | 97.11% |
| Zaki *et al.* | Fine-tuned MobileNet | 90.30% |
| Mohanty *et al.* | Spatial attention with CNN | 95.20% |
| Rangarajan *et al.* | VGG16 | 96.19% |

| Table 6 (Cont.) | | |
|---|---|---|
| Al-gaashani *et al.* | MobileNetV2 and NASNetMobile | 97.00% |
| Hossain *et al.* | Statistical features test and SVM | 90.00% |
| Thanjai Vadivel *et al.* | Fast Enhanced Learning Method | 90.00% |
| Vidyashreeet *et al.* | K-mean clustering, GLCM, and SVM | 90.00% |
| Restrepo-Arias *et al.* | MobileNet | 96.31% |
| Turkoglu *et al.* | ResNet101 | 96.41% |
| Kawasaki *et al.* | CNN | 94.90% |
| Ioffe *et al.* | CNN | 95.48% |
| Nachtigall *et al.* | AlexNet | 97.30% |
| Attallah *et al.* | KNN | 99.92% |
| Ahmed *et al.* | CNN | 99.30% |

Source: *Rangarajan et al. (2018), Restrepo et al. (2022), Wspanialy and Moussa (2020), Abbas et al. (2021), Zaki et al. (2021), Al-gaashani et al. (2021), Kawasaki et al. (2015), Ioffe and Szegedy (2015), Nachtigall et al. (2016), Türkoglu and Hanbay (2019), Zhang et al. (2021), Kanabur et al. (2020), Attallah (2023), Ahmed et al. (2022)*

## 8. Conclusion

Detecting plant diseases early can lessen the e-ffect on the harvest, increasing the productivity of the products. This process can be automated using computer vision and deep learning methods; an example was suggested in this paper. Moreover, this model may be scaled to detect other plant diseases when adapted to other crops.

As it may be known, over the past few years, the efficacy of CNNs in image classification has seen significant advancements. Conventional machine learning methods for disease classification often concentrate on a limited set of classes, typically within a singular crop or for only one type of disease. However, in this study, EfficientNets, which is a CNN model, has been used. This study is unique in the field as it compares all EfficientNets comprehensively, providing both training and testing accuracy results. The utilized dataset consists of 11,000 images, which are reported in Table 6. The dataset consists of color-level, gray-level, and binarized images of the same leaf data that have been used in this study; some of the samples can be seen in Tables 3-5. It was demonstrated in this study that EfficientNetB7 was the best EfficientNets for the color-level dataset.

It is crucial to highlight that the approach presented in this paper is aimed at improving previous disease diagnosis methods/techniques rather than substituting them. Other methods may yield more reliable results compared to diagnoses based on visual cues. Also, early-stage diagnosis solely relying on visual examination often presents challenges.

This study had several constraints. When assessing images taken in varying conditions from the dataset used for training the model, there was a considerable decrease in accuracy. Additionally, the classification was currently restricted to individual leaves oriented upwards against a uniform backdrop.

For further advancements, transfer learning methods hold the potential to utilize various CNN models for extracting features. Lastly, LeNet, AlexNet, GoogleNet, MobileNetV1, DarkNet, and ResNet can serve as potential CNN models in addition to EfficentNet. Instead of using softmax in the last layer, SVM and decision-tree-based can be employed.

## References

Abbas, A., Jain, S., Gour, M. and Vankudothu, S. (2021). Tomato Plant Disease Detection Using Transfer Learning With C-GAN Synthetic Images. *Computers and Electronics in Agriculture*, 187(106279), 106279. https://doi.org/10.1016/j.compag.2021.106279

Adedoja, A., Owolawi, P. A. and Mapayi, T. (2019). Deep learning Based on NASNet for Plant Disease

Recognition using Leave Images. *2019 International Conference on Advances in Big Data, Computing and Data Communication Systems (IcABCD).*

Ahmed, S., Hasan, M.B., Ahmed, T., Sony, M.R.K. and Kabir, M.H. (2022). Less is More: Lighter and Faster Deep Neural Architecture for Tomato Leaf Disease Classification. *IEEE Access: Practical Innovations, Open Solutions*, 10, 68868-68884. https://doi.org/10.1109/access.2022.3187203

Al-gaashani, M.S.A.M., Shang, F., Muthanna, M.S.A., Khayyat, M. and Abd El-Latif, A.A. (2022). Tomato Leaf Disease Classification by Exploiting Transfer Learning and Feature Concatenation. *IET Image Processing*, 16(3), 913-925. https://doi.org/10.1049/ipr2.12397

Altunta, Y. and Kocamaz, F. (2021). Deep Feature Extraction for Detection of Tomato Plant Diseases and Pests Based on Leaf Images. *Celal Bayar Üniversitesi Fen Bilimleri Dergisi*, 17(2), 145-157. https://doi.org/10.18466/cbayarfbe.812375

Attallah, O. (2023). Tomato Leaf Disease Classification Via Compact Convolutional Neural Networks with Transfer Learning and Feature Selection. *Horticulturae*, 9(2), 149. https://doi.org/10.3390/horticulturae9020149

Das, S. (2020). Over View of Septoria Diseases on Different Crops and its Management. *International Journal of Agriculture Environment and Biotechnology*, 13(3). https://doi.org/10.30954/0974-1712.03.2020.13

DeChant, C., Wiesner-Hanks, T., Chen, S., Stewart, E.L., Yosinski, J., Gore, M.A., Nelson, R.J. and Lipson, H. (2017). Automated Identification of Northern Leaf Blight-Infected Maize Plants from Field Imagery Using Deep Learning. *Phytopathology*, 107(11), 1426-1432.

Dönmez, E. (2020). Classification of Haploid And Diploid Maize Seeds Based on Pre-trained Convolutional Neural Networks. *Celal Bayar Üniversitesi Fen Bilimleri Dergisi*. https://doi.org/10.18466/cbayarfbe.742889.

Donmez, E. (2020). Discrimination of Haploid and Diploid Maize Seeds Based on Deep Features. *28th Signal Processing and Communications Applications Conference (SIU).*

Dubey, S.R. and Jalal, A.S. (2012). Detection and Classification of Apple Fruit Diseases Using Complete Local Binary Patterns. *2012 Third International Conference on Computer and Communication Technology*.

Durmus, H., Gunes, E.O. and Kirci, M. (2017). Disease Detection on the Leaves of the Tomato Plants by Using Deep Learning. *2017 6th International Conference on Agro-Geoinformatics*.

Eroltu, K. (2023). Comparing different Convolutional Neural Networks for the classification of Alzheimer's Disease. *Journal of High School Science*, 7(3). https://jhss.scholasticahq.com/article/84787-comparing-different-convolutional-neural-networks-for-the-classification-of-alzheimer-s-disease

François-Xavier Branthôme. (2021). Worldwide (Total Fresh) Tomato Production in 2021 - Tomato News. (n.d.). Tomatonews.com. Retrieved August 10, 2023, from https://www.tomatonews.com/en/worldwide-total-fresh-tomato-production-in-2021_2_1911.html

Gilardi, G., Matic, S., Guarnaccia, V., Garibaldi, A. and Gullino, M. L. (2021). First Report of Fusarium clavum Causing Leaf Spot and Fruit Rot on Tomato in Italy. *Plant Disease*, 105(8), 2250. https://doi.org/10.1094/pdis-05-20-1096-pdn

Huang, H.S., Liu, W., Qi, F. and Yang, K. (2019). Development and Validation of a Deep Learning Algorithm for the Recognition of Plant Disease. *2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS).*

Hughes, D.P. and Salathe, M. (2015). *An open Access Repository of Images on Plant Health to Enable the Development of Mobile Disease Diagnostics*. https://doi.org/10.48550/ARXIV.1511.08060

Ioffe, S. and Szegedy, C. (2015). Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *arXiv [cs.LG]*. http://arxiv.org/abs/1502.03167

Kanabur, V., Harakannanavar, S.S., Purnikmath, V.I., Hullole, P. and Torse, D. (2020). Detection of Leaf Disease Using Hybrid Feature Extraction Techniques and CNN Classifier. In *Computational Vision and Bio-Inspired Computing*, 1213-1220. Springer International Publishing.

Kanda, P.S., Xia, K., Kyslytysna, A. and Owoola, E.O. (2022). Tomato Leaf Disease Recognition on Leaf Images Based on Fine-tuned Residual Neural Networks. *Plants*, 11(21), 2935. https://doi.org/10.3390/plants11212935

Karthik, Hariharan, Anand, S., Mathikshara, P., Johnson, A. and Menaka. (2020). Attention Embedded Residual CNN for Disease Detection in Tomato Leaves. *Applied Soft Computing*, 86(105933), 105933. https://doi.org/10.1016/j.asoc.2019.105933

Kawasaki, Y., Uga, H., Kagiwada, S. and Iyatomi, H. (2015). Basic Study of Automated Diagnosis of Viral Plant Diseases Using Convolutional Neural Networks. In *Advances in Visual Computing*, 638-645. Springer International Publishing.

Khan, M.A., Alqahtani, A., Khan, A., Alsubai, S., Binbusayyis, A., Ch, M.M.I., Yong, H.-S. and Cha, J. (2022). Cucumber Leaf Diseases Recognition Using Multi Level Deep Entropy-ELM Feature Selection. *Applied Sciences (Basel, Switzerland)*, 12(2), 593. https://doi.org/10.3390/app12020593

Liu, B., Zhang, Y., He, D. and Li, Y. (2017). Identification of Apple Leaf Diseases Based on Deep Convolutional Neural Networks. *Symmetry*, 10(1), 11. https://doi.org/10.3390/sym10010011

Liu, J. and Wang, X. (2021). Plant Diseases and Pests Detection Based on Deep Learning: A Review. *Plant Methods*, 17(1). https://doi.org/10.1186/s13007-021-00722-9

Ma, J., Du, K., Zheng, F., Zhang, L., Gong, Z. and Sun, Z. (2018). A Recognition Method for Cucumber Diseases Using Leaf Symptom Images Based on Deep Convolutional Neural Network. *Computers and Electronics in Agriculture*, 154, 18-24.

Mohanty, S.P., Hughes, D.P. and Salathé, M. (2016). Using Deep Learning for Image-based Plant Disease Detection. *Frontiers in Plant Science*, 7. https://doi.org/10.3389/fpls.2016.01419

Nachtigall, L.G., Araujo, R.M. and Nachtigall, G.R. (2016). Classification of Apple Tree Disorders Using Convolutional Neural Networks. *2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)*.

OECD. (2017). *Safety Assessment of Transgenic Organisms in the Environment, 7: OECD Consensus Documents*. OECD.

Panno, S., Davino, S., Caruso, A. G., Bertacca, S., Crnogorac, A., Mandiæ, A., Noris, E. and Matiæ, S. (2021). A Review of the Most Common and Economically Important Diseases that Undermine the Cultivation of Tomato Crop in the Mediterranean Basin. *Agronomy (Basel, Switzerland)*, 11(11), 2188. https://doi.org/10.3390/agronomy11112188

Rangarajan, A.K., Purushothaman, R. and Ramesh, A. (2018). Tomato Crop Disease Classification Using Pre-trained Deep Learning Algorithm. *Procedia Computer Science*, 133, 1040-1047. https://doi.org/10.1016/j.procs.2018.07.070

Restrepo-Arias, J.F., Branch-Bedoya, J.W. and Awad, G. (2022). Plant Disease Detection Strategy Based on Image Texture and Bayesian Optimization with Small Neural Networks. *Agriculture*, 12(11), 1964. https://doi.org/10.3390/agriculture12111964

Rivarez, M.P.S., Vuèuroviæ, A., Mehle, N., Ravnikar, M. and Kutnjak, D. (2021). Global Advances in Tomato Virome Research: Current Status and the Impact of High-throughput Sequencing. *Frontiers in Microbiology*, 12. https://doi.org/10.3389/fmicb.2021.671925

Rodrigues, M.H.P. and Furlong, E.B. (2022). Fungal Diseases and Natural Defense Mechanisms of Tomatoes (*Solanum lycopersicum*): A Review. *Physiological and Molecular Plant Pathology*, 122(101906), 101906. https://doi.org/10.1016/j.pmpp.2022.101906

Sethy, P.K., Barpanda, N.K., Rath, A.K. and Behera, S.K. (2020). Deep Feature Based Rice Leaf Disease Identification Using Support Vector Machine. *Computers and Electronics in Agriculture*, 175(105527), 105527. https://doi.org/10.1016/j.compag.2020.105527

Singh, U.P., Chouhan, S.S., Jain, S. and Jain, S. (2019). Multilayer Convolution Neural Network for the Classification of Mango Leaves Infected by Anthracnose Disease. *IEEE Access: Practical Innovations, Open Solutions*, 7, 43721-43729. https://doi.org/10.1109/access.2019.2907383

Singh, V. and Misra, A. K. (2017). Detection of Plant Leaf Diseases Using Image Segmentation and Soft Computing Techniques. *Information Processing in Agriculture*, 4(1), 41-49. https://doi.org/10.1016/j.inpa.2016.10.005

Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D. and Stefanovic, D. (2016). Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification. *Computational Intelligence and Neuroscience*, 2016, 1-11. https://doi.org/10.1155/2016/3289801

Tan, M. and Le, Q.V. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In *arXiv [cs.LG]*. http://arxiv.org/abs/1905.11946

Tan, W.X., Zhao, C.J. and Wu, H.R. (2016). CNN Intelligent Early Warning for Apple Skin Lesion Image Acquired by Infrared Video Sensors. *High Technol. Lett.*, (1), 67-74.

Türkoðlu, M. and Hanbay, D. (2019). Plant Disease and Pest Detection Using Deep Learning-based Features. *Turkish Journal of Electrical Engineering & Computer Sciences*, 27(3), 1636-1651. https://doi.org/10.3906/elk-1809-181

Waheed, A., Goyal, M., Gupta, D., Khanna, A., Hassanien, A.E. and Pandey, H. M. (2020). An Optimized Dense Convolutional Neural Network Model for Disease Recognition and Classification In Corn Leaf. *Computers and Electronics in Agriculture*, 175(105456), 105456. https://doi.org/10.1016/j.compag.2020.105456

Wspanialy, P. and Moussa, M. (2020). A Detection and Severity Estimation System for Generic Diseases of Tomato Greenhouse Plants. *Computers and Electronics in Agriculture*, 178(105701), 105701. https://doi.org/10.1016/j.compag.2020.105701

Zaki, S.Z.M., Asyraf Zulkifley, M., Mohd Stofa, M., Kamari, N.A.M. and Ayuni Mohamed, N. (2020). Classification of Tomato Leaf Diseases using MobileNet v2. *IAES International Journal of Artificial Intelligence (IJ-AI)*, 9(2), 290. https://doi.org/10.11591/ijai.v9.i2.pp290-296

Zhang, J., Yu, K., Wen, Z., Qi, X. and Kumar Paul, A. (2021). 3D Reconstruction for Motion Blurred Images Using Deep Learning-based Intelligent Systems. *Computers, Materials & Continua*, 66(2), 2087-2104. https://doi.org/10.32604/cmc.2020.014220

Zhao, Y., Sun, C., Xu, X. and Chen, J. (2022). RIC-Net: A Plant Disease Classification Model Based on the Fusion of Inception and Residual Structure and Embedded Attention Mechanism. *Computers and Electronics in Agriculture*, 193(106644), 106644. https://doi.org/10.1016/j.compag.2021.106644

Zhou, G., Zhang, W., Chen, A., He, M. and Ma, X. (2019). Rapid Detection of Rice Disease Based on FCM-KM and Faster R-CNN Fusion. *IEEE Access: Practical Innovations, Open Solutions*, 7, 143190-143206. https://doi.org/10.1109/access.2019.2943454

**Cite this article as:** Kaan Eroltu (2024). Tomato Leaf Classification Using Computer Vision and Deep Learning: Comparing Different EfficientNets. *International Journal of Artificial Intelligence and Machine Learning*, 4(1), 61-79. doi: 10.51483/IJAIML.4.1.2024.61-79.