

EfficientNet-Based Robust Recognition of Peach Plant Diseases in Field Images

Haleem Farman¹, Jamil Ahmad^{1,*}, Bilal Jan², Yasir Shahzad³, Muhammad Abdullah¹ and Atta Ullah⁴

¹Department of Computer Science, Islamia College Peshawar, 25120, Pakistan

²Department of Computer Science, FATA University Kohat, 26100, Pakistan

³Department of Computer Science, University of Peshawar, 25120, Pakistan

⁴Agriculture Research Institute, Mingora, Swat, 19130, Pakistan

*Corresponding Author: Jamil Ahmad. Email: jamil.ahmad@icp.edu.pk

Received: 27 March 2021; Accepted: 15 June 2021

Abstract: Plant diseases are a major cause of degraded fruit quality and yield losses. These losses can be significantly reduced with early detection of diseases to ensure their timely treatment, particularly in developing countries. In this regard, an expert system based on deep learning model where the expert knowledge, particularly the one acquired by plant pathologist, is recursively learned by the system and is applied using a smart phone application for use in the target field environment, is being proposed. In this paper, a robust disease detection method is developed based on convolutional neural network (CNN), where its powerful features extraction capabilities are leveraged to detect diseases in images of fruits and leaves. The features extraction pipelines of several state-of-the-art pretrained networks are fine-tuned to achieve optimal detection performance. A novel dataset is collected from peach orchards and extensively augmented using both label-preserving and non-label-preserving transformations. The augmented dataset is used to study the effects of fine-tuning the pretrained networks' feature extraction pipeline as opposed to keeping the network parameters unchanged. The CNN models, particularly EfficientNet exhibited superior performance on the target dataset once their feature extraction pipelines are fine-tuned. The optimal model is able to achieve 96.6% average accuracy, 90% sensitivity and precision, and 98% specificity on the test set of images.

Keywords: Peach diseases; EfficientNet; data augmentation; transfer learning

1 Introduction

The importance of fruits, vegetables and related food products is undeniable in providing essential nutrients to the human body to generate energy for its functioning. Fresh and healthy food items with vital nutrients, especially fruit products bring high health benefits that are essential for the maintenance of human life. Healthy, fresh, unprocessed and disease-free fruits help build up body immunity, make the body strong, keep the body hydrated, fight free radicals, and strengthen all body organs and ensure



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

their proper functioning [1]. However, low nutrient or inadequate fruits consumption can have adverse effects on the human body which may lead to chronic diseases such as cardiovascular, diabetes, renal and retinopathy [2]. Fruit farming is therefore an important area of research that plays a fundamental and healthy role in fresh fruit production and in the management of various fruit diseases. These diseases are a major contributor to the crop losses if they are left untreated which may lead to severe damage to fruit plants. Around 10–15% of the yield is wasted due to diseases [3], which is a big concern for farmers.

The Food and Agriculture Organization of the United Nations (FAO) forecasts a 70% rise in agriculture demand by 2050. Due to this growing increase in demand, it is inevitable to minimize the effects of diseases on crops especially on fruits. FAO warns that peach crops are confronted with severe diseases, such as bacterial canker, bacterial spot, crown gall, and peach scab causing inadequate health and quality problems [4]. Peach is Pakistan's second most important fruit after plum, which faces the same severe threat due to lack of technological resources in predicting early infections to prevent and control threats and also reduce financial liability. Machine learning and Internet of Things (IoT) have significantly improved smart agriculture where diseases are identified in real-time by a single tap with acceptable accuracy. Considerable progress has been made in the field of computer vision, in particular in the detection and recognition of objects [5–8]. Variety of Convolutional Neural Network (CNN) techniques are available in literature for the detection and identification of fruit diseases [9]. However, CNN is computationally complex and resource hungry, particularly when deployed over resource constrained devices. CNN's performance is enhanced by Transfer Learning (TL) techniques on large datasets to process input data and classify into modified classes, with no hand-engineered features extraction involved [10,11]. One of these modern approaches is the use of ICT assisted disease diagnosis and management thus eliminating hurdles in conventional disease control approaches. The technology is an end-to-end application that can be used efficiently on readily available devices such as smartphones or tablets. In this paper, our main contributions are as follows:

- Collected and annotated peach images dataset containing healthy and infected fruits and leaves by capturing images under true field conditions.
- Trained and fine-tuned a wide variety of powerful deep learning model for disease identification in peach plants utilizing extensive data augmentation techniques to search for optimal model.
- Experimented with frozen and fine-tuned models to determine optimal features extraction pipeline for disease detection in peach plants. We also utilized class activation maps (CAM) to demonstrate the effectiveness of the feature extraction of the optimal model.

The rest of the paper is organized as follows: Relevant literature is briefly discussed in Section 2. Section 3 presents the proposed method in detail, followed by experimental results and discussions in Section 4. The paper concludes with limitations of the proposed method along with future work suggestions in Section 5.

2 Related Work

In literature, deep learning approaches have been widely adopted to identify diseases in crops. Most of the techniques have used image recognition using a classifier to get the desired results [6]. In [12], the authors carried out a detailed survey to determine fruit type and estimate yields. All existing research is summarized for easy realization and implementation to choose the best crop detection system. The authors recommend the use of neural networks due to their capability to detect and recognize objects. Neural networks are also capable of detecting and learning rudimentary

features from visual inputs such as shapes and patterns. In addition, the authors recommend transfer learning as a basic approach in primary layers to find optimal weights for parameters by tuning the hyperparameters like learning rate, momentum, initialization, and activation function.

Syamsuri et al. [13] proposed a detection system with optimal performance and latency for both personal computers and mobile phones. The authors investigated MobileNet [14], Mobile NASNet [15], and InceptionV3 [16] for resource constrained devices for the development of various applications. Resource utilization comparison is made for memory, CPU, and battery use. The coffee leaf infection dataset is extracted from the PlantVillage repository. Accuracy and time delay results are compared to the models mentioned above. The authors recommend the use of smartphones for the detection of plant disease due to easy handling, low resource utilization, and negligible degradation of accuracy compared to desktop CPUs. In another work targeting efficiency for resource constrained devices, Duong et al. [17] developed an expert system for the recognition of fruits from input images through image processing and machine learning. The authors used two classifiers namely EfficientNet [18] and MixNet to identify fruits using limited computational resources in a real-time environment. Performance is evaluated using a real dataset of 48,905 images for training purposes and 16,421 images for testing purposes through randomization and transfer learning approaches. The authors also endorse the role of pre-trained weights in transfer learning in the detection of plant disease.

An EfficientNet based method is introduced in [19] to identify and classify maize plant leaf infections. A trivial dataset sample is extracted from the AI Challenger dataset and few web images of maize disease. Images are initially processed for cleaning and screening in order to prepare a sample dataset, augmented using scaling, translation, and rotation transformations. A total of 9279 images are collected of which 6496 are used for training and 2783 are used for testing. Transfer learning is used to improve the accuracy and recognition speed based on the EfficientNet model. The proposed model scores 98.85% accuracy compared to EfficientNet, VGG-16 [20], Inception-V3, and Resnet-50 [21].

In [22], Liu et al. proposed a model namely Leaf Generative Adversarial Networks (Leaf GAN) to identify grape leaf disease. The model generates rich grapes leaf disease images in four categories and is also capable of distinguishing between fake and true disease images. The dataset extracted from PlantVillage consists of 4062 images of grape leaf disease mixed with 8124 images generated. The overfitting problem is recovered by data augmentation and deep regret analytic gradient penalty. The proposed method has a maximum accuracy of 98.7% on the Xception model alongside other classification models.

The authors in [23] proposed a method to identify rice disease in a fast, automatic, less expensive, and accurate manner. The authors preferred to benefit from the DenseNet [24] and Inception modules. Dataset is extracted from ImageNet which is used in transfer learning a new Inception module called DENS-INCEP. The authors take 500 images having 120 color leaf images of rice plant diseases with uneven illumination intensities. The images are initially processed for resizing, edge filling, and sharpening through Photoshop to be configured in a proper RGB model with size adjustments. The proposed model has an accuracy of 98.63% compared to DenseNet and Inception.

In a similar study [25], the authors proposed an identification technique to detect rice and maize diseases. The proposed method enhances learning capability through the use of deep CNN with transfer learning. The dataset is extracted from the PlantVillage dataset consisting of 54306 plant leaf images. The authors prefer to plan positioning and visualization with pre-trained MobileNet-V2 [26] using two-fold transfer learning. The first approach is aimed at concluding initial weights by keeping the lower layers of CNN frozen, while the second is to retrain weights of transfer learning by loading

the first phase of a trained model. The complete process revolves around changing the size of images, image pre-processing, model training/testing, and its validation. The model has an average accuracy of 99.11%, a sensitivity of 92.92%, and a specificity of 99.52% for rice and maize diseases.

Authors in [27] proposed an identification model based on simplified CNN that can take up less storage for tomato crop disease. The authors also explained that CNN has an edge over other machine learning techniques in addition to its computational complexity. The dataset is extracted from PlantVillage with 55000 leaf images of 14 crops in which the authors experimented with 10 classes of tomato crop disease. The proposed method has an accuracy of 98.4 percent compared to traditional machine learning techniques using VGG16, Inception V3, and MobileNet.

In [28], the authors investigated the detection and improvement of plant lesion features using transfer learning with deep CNN. The features of both MobileNet and squeeze-and-excitation network called SE-MobileNet helps to identify plant disease. Two datasets are extracted from the public PlantVillage and real rice disease dataset. The dataset of PlantVillage consists of 54306 plant leaf images, while the real dataset consists of 600 images of rice plant diseases. Both dataset's image inconsistencies are recovered by Photoshop for RGB pattern and image scaling. The authors used a two-fold transfer learning approach, such as loading pre-trained weights on ImageNet and retraining on target datasets. The proposed method scores 99.78% compared to InceptionV3, VGGNet-19, DenseNet121, NASNetMobilet, and MobileNetV2. A summary of the previous works in the domain of plant disease detection with deep learning methods is provided in [Tab. 1](#).

Table 1: A summary of plant disease detection approaches using deep learning

Reference	Proposition	Dataset	No. of images	Compared with	Accuracy
EfficientNet based recognition of maize diseases by leaf image classification	Detect maize plant leaf infection	AI Challenger	9279	EfficientNet, VGG-16, Inception-V3 and Resnet-50.	98.85%
A data augmentation method based on generative adversarial networks for grape leaf disease identification	Detect grape plant leaf infection	PlantVillage	4062 8124- (generated)	AlexNet, VGG, ResNet, DenseNet, Xception, ResNext, SEResNet, and EfficientNet	98.7%
Detection of rice plant diseases based on deep transfer learning	Detect rice plant leaf infection	ImageNet	500	DenseNet and Inception	98.63%

(Continued)

Table 1: Continued

Reference	Proposition	Dataset	No. of images	Compared with	Accuracy
Identifying plant diseases using deep transfer learning and enhanced lightweight network	Detect rice and maze plants leaf infection	PlantVillage	54306	InceptionV3, VGG-Net19, ResNet50, DenseNet121, MobileNetV2, ResNet50 + SVM, and MobileNetV2 + SVM	99.11%
Development of Efficient CNN model for Tomato crop disease identification	Detect tomato plant leaf infection	PlantVillage	55000	VGG16, InceptionV3, and Mobilenet	98.4%
Identification of plant disease images via a squeeze-and-excitation MobileNet model and twice transfer learning	Detect rice plant leaf infection	PlantVillage	54306, 600 (real)	InceptionV3, VGGNet-19, DenseNet121, NASNetMobilet, and MobileNetV2	99.78%

3 Materials and Methods

In this work, we investigated the effectiveness of CNNs in detecting diseases in Peach plants images captured from fields under varying illumination conditions. Several state-of-the-art CNNs including AlexNet [28], Inception, ResNet, MobileNet, and EfficientNet were considered. The proposed framework for disease detection is provided in Fig. 1. Pretrained models were used as backbone feature extractors as well as for fine-tuning on the target dataset for recognition of peach plant diseases. Further details of each component in the proposed framework are illustrated in the subsequent sections.

3.1 Dataset Collection

The objective of dataset collection was to develop a challenging dataset suitable for training CNNs that can robustly predict diseases in field images captured on a smartphone camera. The dataset consists of peach fruit, leaf, and stem images captured using smartphone cameras under a variety of environmental and lighting conditions. A total of 2500 images were captured from different regions of Khyber Pakhtunkhwa province in Pakistan. Some of the images contained more than one fruit or leaf, having the possibility of both healthy and diseased fruits in a single image. Image level annotation would result in a noisy dataset. Though slightly noisy dataset helps in robust model training, further

refinement was needed to clean the dataset. The collected dataset was named as Peach Diseases Dataset Islamia College Peshawar (PDDICP).

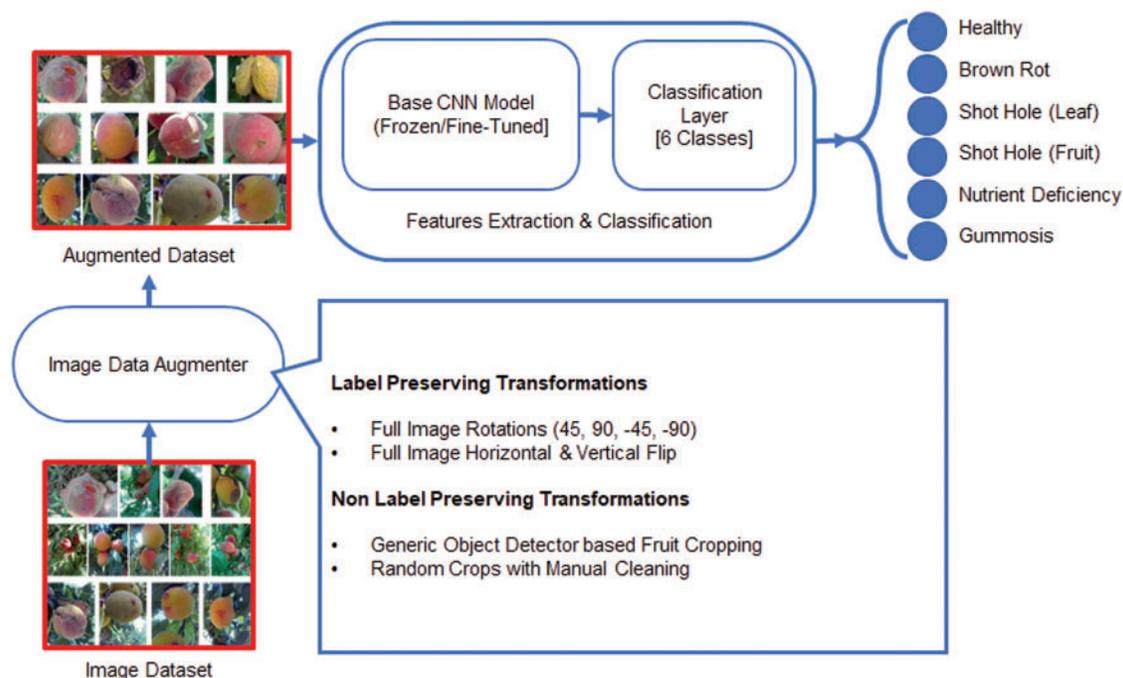


Figure 1: Proposed framework for disease detection in Peach images

3.2 Data Annotation

The dataset was annotated by an expert plant pathologist with focus on disease detection even in the presence of healthy fruits or leaves in the same image. For this purpose, an image was annotated having disease even if it contained healthy fruits too. Such annotations were performed with the hope of obtaining more robust and disease focused detection models. The images were annotated and categorized into 6 groups including healthy, brown rot, gummosis, nutrient deficiency, shot hole leaf, and shot hole fruit. Each category contains around 400 images in each category in the original dataset. Samples of collected images are shown in Fig. 2. To further improve the size, diversity, and complexity of the dataset, extensive data augmentation was performed.

3.3 Data Augmentation

Deeper CNNs usually require large datasets to converge. In most cases, datasets are expanded artificially using data augmentation techniques [9]. This is achieved by applying label preserving transformations like rotations, scaling, translations, flipping, and cropping. Sometimes, cropping is non label preserving because the cropped part may or may not contain an object of interest which may actually render the previous annotation incorrect. In the present study, images captured from the fields often contained multiple fruits and leaves. Image level annotations were primarily performed on images with focus on disease in them. That is, the image was labeled to have a disease even if multiple healthy fruits were present beside the diseased one. Training CNNs with slightly noisy datasets often yield robust detection performance and becomes capable of spotting object or region of interest in the

presence of noise. To this end, we applied various transformations on the dataset in order to not only increase the number of samples, but to create more realistic samples simulating real-world imperfect images captured from field by naïve users. Firstly, we isolated objects (fruits) from images utilizing a pre-trained object detector known as single shot multi-box detector (SSD) [29]. This detector was trained on a number of fruits including apple, lemon, and avocado. Though the model has not been trained on peach images, the visual similarities in peach and apple and avocado helped it being detected by this model with considerable accuracy. We decided to utilize this model as a generic detector to detect and isolate individual peach fruits from the images. Each image was propagated through this model to estimate object positions in the image. The detected bounding boxes were then used to isolate fruits from the image, followed by manual cleaning of labels for the isolated objects. Consequently, a relatively cleaner subset of samples, with slight occlusions, was created from the original dataset. Secondly, four random center crops were taken from each image in the original dataset. Rotated and flipped versions of these images were also created to further increase the dataset size. The augmented images were then inspected and annotated by the plant pathologist to avoid any noisy labels.



Figure 2: Peach images dataset

3.4 State of the Art CNN Architectures as Base Models

Convolutional neural networks were developed decades ago, however, their superior performance was demonstrated in the current decade when a deep CNN outperformed all other traditional approaches in large scale image recognition problem in 2012 [30]. Since then, researchers have been extensively investigating CNNs to solve challenging problems in computer vision and other fields. Krizhevsky et al. [30] won the ImageNet Large Scale Visual Recognition Challenge (ILSVRS) [31] with his AlexNet architecture, which consisted of five convolutional layers and three fully connected layers. Deeper models were later developed exhibiting superior performance compared to shallower models [20]. In He et al. [21], showed that increasing the depth does not always guarantee superior performance. They exhibited that performance plateaus or even declines when depth increases after achieving a peak performance. In their work, they introduced residual connections and proved that deeper networks can be designed with residual blocks without loss of performance as the network depth increases. Highly complex CNN architectures have been developed over the years, where researchers exhaustively attempt to tune hyper-parameters like network width, depth, resolution, and number and size of convolutional kernels. Recently, research has been carried out to investigate and develop hyper-parameter optimization methods which can automatically tune them on sample datasets. In this regard, Zoph et al. [15] developed Neural Architecture Search (NAS) method which allowed researchers to automatically develop convolutional building blocks, balancing the tradeoff between performance and efficiency. Recently, EfficientNet architectures were developed by Tan et al. [18] who proposed systematic model scaling by identifying optimized network depth, width and resolution, thereby leading to superior performance and higher efficiency than existing state-of-the-art methods. EfficientNets have exhibited superior performance in image classification and object detection in a wide variety of challenging image datasets. Their superior feature extraction capabilities have also been used in transfer learning to address a variety of computer vision and related problems.

3.5 Disease Detection with EfficientNets

EfficientNets have been introduced recently as a group of 8 models named from B0 to B7. The model B7 have exhibited state-of-the-art performance on the ImageNet dataset by achieving 84.4% Top-1 accuracy while utilizing only 66 Million parameters. With EfficientNets, a new activation function was introduced known as Swish $f(x) = x \cdot \text{sigmoid}(x)$ [32]. This activation function is known to perform better than ReLU activation in a number of networks/datasets. EfficientNet architectures were developed by searching a grid for optimal performer under a fixed resource constraint. Utilizing the neural architecture (NAS) mechanism, a suitable scaling factor for depth, width, and input resolution is determined for a particular dataset. The network is then scaled according to the target dataset in order to achieve optimal performance at optimal cost.

The main building block in EfficientNet is the inverted bottleneck block (MBConv), which was initially introduced with MobileNetV2. These blocks first expand the channels and then compresses them, thereby reducing the number of channels for the subsequent layer. Furthermore, this architecture uses in-depth separable convolutions that further reduces computational burden by a factor of k^2 , where k is the kernel size.

In this study, we utilized the EfficientNet architectures to detect diseases in Peach plants in order to determine the best model for deployment on a mobile device. All the models from B0 to B5 were fine-tuned using transfer learning on the augmented dataset. The reason for not training them on the original dataset was that it mostly consisted of images having more than one fruit, often belonging to different categories. Using that dataset was prone to create a model with greater confusion. Therefore,

only the augmented dataset was used in this study. Transfer learning of the models were performed based on the parameters given in Tab. 2. All models under consideration were first fine-tuned on the target dataset while keeping the base models frozen (i.e., keeping the parameters of the base model unchanged). In this case, only the classification layer was optimized and the backbone network was used as a pre-trained feature extractor. Later on, the entire model was fine-tuned to observe the performance difference as a result.

Table 2: Parameter values for training deep learning models

Model	Input size	Optimization method	Batch size	Learning rate (Frozen base)	Learning rate (Fine-tuned)	No. of parameters
AlexNet	227 × 227	SGD (Momentum = 0.9)	128	0.001	0.0001	60.95 M
ResNet50	224 × 224	Adam (Beta1 = 0.9, Beta2 = 0.999)	32	0.001	0.0001	25.64 M
Inception V3	299 × 299	Adam (Beta1 = 0.9, Beta2 = 0.999)	32	0.001	0.0001	23.85 M
MobileNet-V1	224 × 224	Adam (Beta1 = 0.9, Beta2 = 0.999)	32	0.001	0.0001	2.26 M
MobileNet-V2	224 × 224	Adam (Beta1 = 0.9, Beta2 = 0.999)	32	0.001	0.0001	2.26 M
EfficientNet-B0	224 × 224	Adam (Beta1 = 0.9, Beta2 = 0.999)	64	0.001	0.0001	5.33 M
EfficientNet-B1	224 × 224	Adam (Beta1 = 0.9, Beta2 = 0.999)	64	0.001	0.0001	7.86 M
EfficientNet-B2	224 × 224	Adam (Beta1 = 0.9, Beta2 = 0.999)	32	0.001	0.0001	9.18 M
EfficientNet-B3	224 × 224	Adam (Beta1 = 0.9, Beta2 = 0.999)	32	0.001	0.0001	12.32 M
EfficientNet-B4	299 × 299	Adam (Beta1 = 0.9, Beta2 = 0.999)	32	0.001	0.0001	19.47 M
EfficientNet-B5	299 × 299	Adam (Beta1 = 0.9, Beta2 = 0.999)	32	0.001	0.0001	28.53 M

4 Experiments and Results

4.1 Experimental Setup

The experiments were conducted on a system running Windows 10, equipped with a Core i5 CPU, 16 GB RAM, and a GeForce RTX 2060 Super GPU with 8 GB of VRAM. Images were captured using a variety of smartphone cameras including Samsung Galaxy S8 (16 MP), Huawei (5 MP), Oppo A33 (8 MP), and some (4 MP) cameras so that a varying degree of image quality could be obtained. Captured image resolutions were 3456×4608 (16 MP), 1920×2560 (5 MP), 2400×3200 (8 MP), and 2560×1440 (4 MP) in size, respectively. TensorFlow v2.4.1 [33] library was used to train and evaluate all the CNN models used in this study.

4.2 Experiment Design

A number of experiments were designed to evaluate performance of the proposed framework in terms of robustness in disease detection and capability of avoiding misdetections. The augmented dataset was used to train and evaluate several state-of-the-art CNN models. The original dataset, though had sufficiently large number of images per category, we felt it would be best to make it more challenging in terms of simulating real world image capturing scenarios in the field. In the first experiment, we fine-tuned a number of pre-trained CNNs by training only the classification layer and freezing the rest of the layers using the augmented dataset and evaluated their performance on the test set. This experiment was designed to evaluate the features extraction pipeline of the CNNs for detection of diseases in peach plants. Though all CNNs have very capable features extraction layers, their suitability for the proposed study required detailed assessment. In the second experiment, we fine-tuned the entire networks with a smaller learning rate for a limited number of epochs on the target dataset in order to allow the network to learn specific features across the feature extraction layers. This could help boost the performance of networks.

4.3 Performance Evaluation Metrics

Performance of all the CNNs used in this study were measured using a variety of metrics including accuracy, precision, sensitivity, and specificity. All these metrics are determined from the confusion matrix consisting of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN). Accuracy indicates the rate of correctly classified samples out of all the samples in a test set for a particular class. Sensitivity measures the ratio of accurately predicted positives to all true positives. This determines the robustness of the model to detect diseases in positive image samples. Specificity on the other hand measures the ratio of correctly classified negative samples out of all true negatives, exhibiting the capability of the model to avoid misdetections in samples. Precision determines the rate of correct prediction of positive samples out of all positive identifications. The wider set of evaluation metrics ensure the robustness and effectiveness of detections as well as avoiding misdetections. For each class k , these metrics are measured as follows.

$$Acc(k) = \frac{TP(k) + TN(k)}{TP(k) + FN(k) + TN(k) + FP(k)} \quad (1)$$

$$Sen(k) = \frac{TP(k)}{TP(k) + FN(k)} \quad (2)$$

$$Spec(k) = \frac{TN(k)}{TN(k) + FP(k)} \quad (3)$$

$$Prec(k) = \frac{TP(k)}{TP(k) + FP(k)} \quad (4)$$

4.4 Performance with Frozen Base Model

This experiment was carried out to evaluate the features extraction performance of various pretrained models. The base model, consisting of the features extraction layers was frozen (learning rate set to zero), which effectively prevented these layers from any modification during the transfer learning process. All the models were trained with the augmented dataset where 80% of the dataset was used for training and the remaining 20% was used for testing. Disease detection performance on the test set is shown in Tab. 3. It can be seen that the EfficientNet-B2 architecture performed the best, yielding the most balanced and optimal performance on the test set. The larger and computationally expensive EfficientNet models B3 to B5 tend to overfit after a few epochs and the rest of the models yield considerable performance, however, the EfficientNet-B2 provides a good balance between computational requirements and performance. The rest of the CNNs like ResNet50, InceptionV3, and MobileNetV2 showed acceptable performance, whereas AlexNet being a relatively shallow network exhibited lowest performance. Confusion matrix of EfficientNet-B2 is provided in Tab. 4.

The feature extraction pipeline of EfficientNet-B2 is very capable given its excellent recognition performance on the target dataset even without any fine-grained tuning of its parameters. The true positive rate of nutrient deficiency which is determined from discoloration in leaves is the highest at 0.94. Considering diseases on leaves, a 5% confusion was observed between nutrient deficiency and shot hole leaf. Similarly, 11% miss-detections were noticed between brown rot and healthy samples due to high degree of visual similarity among the ripe fruits. The generic feature extractors in pretrained models do not discriminate a healthy peach from an unhealthy one. Evidently, their feature extraction pipelines are capable of extracting fine-grained visual features allowing the classifier to discriminate among the classes. Results of EfficientNet-B2 with frozen base model are depicted in Fig. 3. The overall detection accuracy is high; however, sensitivity and precision are low, particularly in case of brown rot and shot hole. The precision of healthy class is also low, which is expected because some images in the dataset with healthy samples were labeled as disease when unhealthy fruits were present around it in the same image.

Table 3: Disease detection performance of various pretrained CNN models

Model	Avg. Acc. (%)	Avg. Sens. (%)	Avg. Spec. (%)	Avg. Prec. (%)
AlexNet	0.903	0.748	0.941	0.769
ResNet50	0.932	0.813	0.960	0.830
Inception V3	0.941	0.832	0.966	0.841
MobileNet-V1	0.911	0.802	0.946	0.817
MobileNet-V2	0.924	0.813	0.959	0.822
EfficientNet-B0	0.934	0.844	0.963	0.851
EfficientNet-B1	0.942	0.849	0.960	0.860
EfficientNet-B2	0.948	0.857	0.968	0.863
EfficientNet-B3	0.951	0.852	0.962	0.861

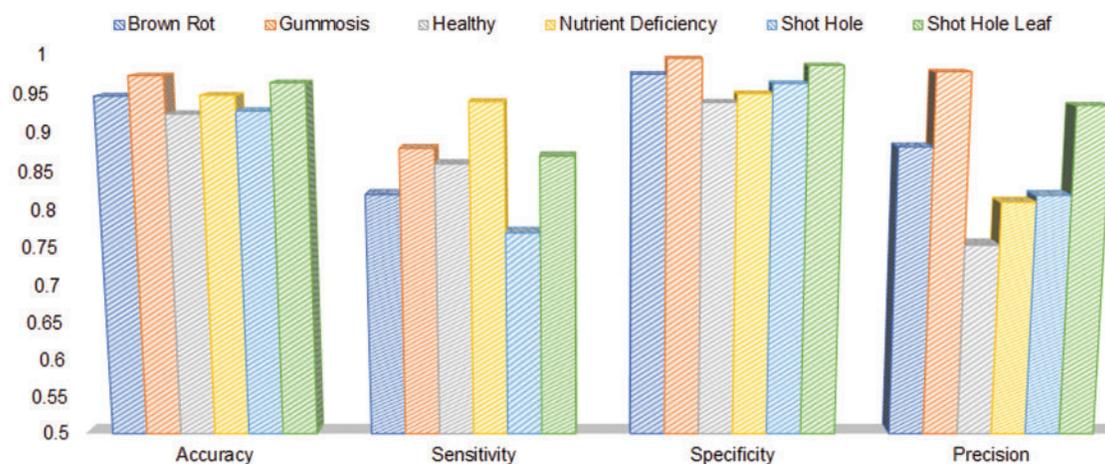
(Continued)

Table 3: Continued

Model	Avg. Acc. (%)	Avg. Sens. (%)	Avg. Spec. (%)	Avg. Prec. (%)
EfficientNet-B4	0.941	0.846	0.955	0.842
EfficientNet-B5	0.947	0.844	0.962	0.859

Table 4: Confusion matrix obtained using EfficientNet-B2 with frozen base model

Predicted label		Brown rot	Gummosis	Healthy	Nutrient deficiency	Shot hole	Shot hole leaf
True label	Brown rot	0.82	0.02	0.11	0.0	0.05	0.0
	Gummosis	0.06	0.88	0.06	0.0	0.0	0.0
	Healthy	0.01	0.0	0.86	0.01	0.11	0.01
	Nutrient deficiency	0.0	0.0	0.0	0.94	0.01	0.05
	Shot hole	0.04	0.0	0.03	0.16	0.77	0.0
	Shot hole leaf	0.0	0.0	0.08	0.05	0.0	0.87

**Figure 3:** Per-class accuracy, sensitivity, specificity, & precision using EfficientNet-B2 with frozen base model

4.5 Performance with Full Fine-Tuning

In the previous experiment, we witnessed that utilizing the feature extraction pipeline of pretrained CNNs can yield considerable performance without requiring extensive fine-tuning of the entire network. However, dataset specific feature optimizations can be achieved if the features extraction layers are allowed to fine-tune on the target dataset. In these experiments, we allowed the entire networks to fine-tune for 30 epochs with early stopping strategy. This enabled us to obtain better

overall results on the test set as shown in Tab. 5. The confusion matrix using the best performing network EfficientNet-B2 is shown in Tab. 6. A lot of the confusion is reduced as a result of fine-tuning, owing to enhancing the capability of feature extraction pipeline to discriminate between healthy and unhealthy fruit. Considerable improvements have been noticed after fine-tuning the entire network on the target dataset. Fig. 4 shows the performance of EfficientNet-B2 with fine-tuned feature extractor. The sensitivity and precision have been considerably improved indicating a much robust detection model after fine-tuning. The larger models including B3, B4, and B5, tend to overfit when the features extraction layers were allowed to change during training.

4.6 Field Testing Results

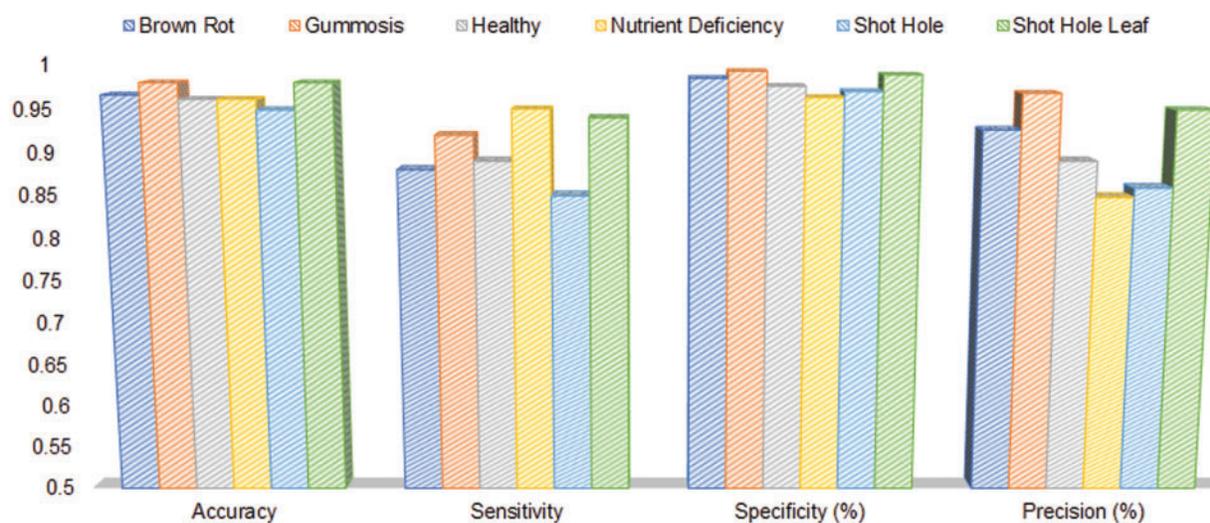
To appropriately assess the effectiveness and robustness of the disease detection model, we captured some images from the field and tested them using the optimized model without any preprocessing. Results shown in Fig. 5 exhibit the capability of the model to detect diseases in challenging situations and even in the presence of healthy samples in the same image. For each test image, we have also included a class activation map to showcase the salient part of the image on which the network's attention is focused. These class activations exhibit that the model is capable of identifying regions which causes high probability of a particular class. In the top row, activations of the brown rot image are focused on the infected areas of the fruit. Though in the first top left most image, some parts of the healthy fruit are also visible, but they have been ignored. Similarly, in the second row, high activations exist in the location of the healthy fruits in the image. In case of gummosis, the upper region with the gum has high activations. Though the other parts having gummosis is ignored, the prediction was made accurately by the model. In the third row, both images had nutrient deficiency in some leaves, which have been correctly identified by the CNN. The other healthy leaves present in the same image have been ignored, which shows the robustness of the model in disease detection even in the presence of healthy fruits and leaves. Activations in the last row also reveal identification of infected regions on both the fruit and leaves.

Table 5: Disease detection performance of various fine-tuned CNN models

Model	Avg. Acc. (%)	Avg. Sens. (%)	Avg. Spec. (%)	Avg. Prec. (%)
AlexNet	0.922	0.820	0.963	0.818
ResNet50	0.952	0.861	0.971	0.875
Inception V3	0.960	0.893	0.975	0.882
MobileNet-V1	0.931	0.824	0.969	0.836
MobileNet-V2	0.945	0.835	0.968	0.852
EfficientNet-B0	0.951	0.843	0.971	0.867
EfficientNet-B1	0.962	0.895	0.973	0.892
EfficientNet-B2	0.966	0.905	0.980	0.907
EfficientNet-B3	Overfit			
EfficientNet-B4	Overfit			
EfficientNet-B5	Overfit			

Table 6: Confusion matrix obtained using EfficientNet-B2 with fine-tuned base model

Predicted label		Brown rot	Gummosis	Healthy	Nutrient deficiency	Shot hole	Shot hole leaf
True label	Brown rot	0.88	0.03	0.04	0.0	0.05	0.0
	Gummosis	0.04	0.92	0.04	0.0	0.0	0.0
	Healthy	0.01	0.0	0.89	0.01	0.08	0.01
	Nutrient deficiency	0.0	0.0	0.0	0.95	0.01	0.04
	Shot hole	0.02	0.0	0.03	0.10	0.85	0.0
	Shot hole leaf	0.0	0.0	0.00	0.06	0.0	0.94

**Figure 4:** Per-class accuracy, sensitivity, specificity, & precision using EfficientNet-B2 with fine-tuned base model

Gradient based class activation maps (Grad-CAM) in Fig. 5 indicate that EfficientNet-B2 is a highly capable architecture and can accurately spot diseases in peach fruit under challenging conditions. These activations can be exploited to locate infected regions of fruit and leaves for precision farming applications like applying pesticides to infected fruits or removing them from trees to avoid the spread of infections to other nearby fruits and trees.

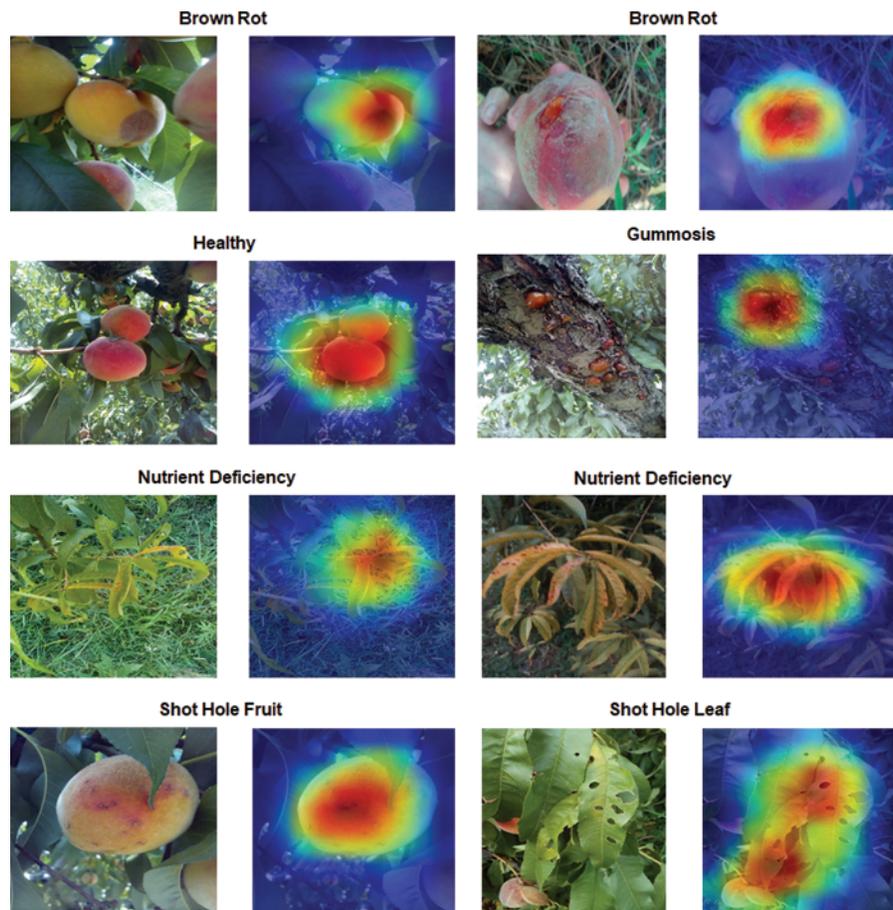


Figure 5: Class activation maps of various categories in PDDICP dataset using EfficientNet-B2

5 Conclusion and Future Work

In this work, we investigated the effects of fine-tuning on an augmented dataset of peach disease images. Dataset was collected from several areas in Khyber Pakhtunkhwa province and expanded artificially utilizing label-preserving transformations like rotations and flips as well as non-label-preserving transformations like generic object detector based cropping and random center crops. It was observed that fine-tuning the features extraction backbone significantly improves performance as opposed to utilizing them as fixed feature extractors. Furthermore, the dataset was augmented in a manner to simulate imperfect image capturing in the field on a smartphone. The augmented dataset helped us in training robust detection models which were capable of detecting diseases in challenging field images.

In future, we intend to label the augmented dataset for simultaneous object detection and classification using end-to-end deep learning approaches. Furthermore, we also aim to utilize auto augmentation procedures to determine the best set of data augmentation transformations for the target dataset.

Funding Statement: We are highly grateful for the financial support provided by Sustainable Development Unit, Planning & Development Department, Government of Khyber Pakhtunkhwa, Pakistan under the program “Piloting Innovative Ideas to Address Key Issues of Khyber Pakhtunkhwa”.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] H. I. Ali, S. G. Al-Shawi and H. N. Habib, “The effect of nutrition on immune system review paper,” *Food Science and Quality Management*, vol. 90, pp. 31–35, 2019.
- [2] K. R. Siegel, “Insufficient consumption of fruits and vegetables among individuals 15 years and older in 28 low and middle income countries: What can be done?,” *The Journal of Nutrition*, vol. 149, pp. 1105–1106, 2019.
- [3] R. Cerda, J. Avelino, C. Gary, P. Tixier, E. Lechevallier *et al.*, “Primary and secondary yield losses caused by pests and diseases: Assessment and modeling in coffee,” *PloS One*, vol. 12, pp. e0169133, 2017.
- [4] W. Alosaimi, H. Alyami and M.-I. Uddin, “Peachnet: Peach diseases detection for automatic harvesting,” *Computers, Materials & Continua*, vol. 67, pp. 1665–1677, 2021.
- [5] J. Ahmad, K. Muhammad, S. Bakshi and S. W. Baik, “Object-oriented convolutional features for fine-grained image retrieval in large surveillance datasets,” *Future Generation Computer Systems*, vol. 81, pp. 314–330, 2018.
- [6] J. Ahmad, K. Muhammad, I. Ahmad, W. Ahmad, M. L. Smith *et al.*, “Visual features based boosted classification of weeds for real-time selective herbicide sprayer systems,” *Computers in Industry*, vol. 98, pp. 23–33, 2018.
- [7] J. Ahmad, I. Mehmood and S. W. Baik, “Efficient object-based surveillance image search using spatial pooling of convolutional features,” *Journal of Visual Communication and Image Representation*, vol. 45, pp. 62–76, 2017.
- [8] M. Khan, B. Jan and H. Farman, *Deep Learning: Convergence to Big Data Analytics*, Singapore: Springer, 2019. [Online]. Available: <https://www.springer.com/gp/book/9789811334580>.
- [9] J. Ahmad, B. Jan, H. Farman, W. Ahmad and A. Ullah, “Disease detection in plum using convolutional neural network under true field conditions,” *Sensors*, vol. 20, no. 19, pp. 1–18, 2020.
- [10] J. Ahmad, H. Farman and Z. Jan, “Deep learning methods and applications,” in *Proc. Deep Learning: Convergence to Big Data Analytics*, Singapore, Springer, pp. 31–42, 2019.
- [11] B. Jan, H. Farman, M. Khan, M. Imran, I. U. Islam *et al.*, “Deep learning in big data analytics: Comparative study,” *Computers & Electrical Engineering*, vol. 75, pp. 275–287, 2019.
- [12] A. Koirala, K. B. Walsh, Z. Wang and C. McCarthy, “Deep learning method overview and review of use for fruit detection and yield estimation,” *Computers and Electronics in Agriculture*, vol. 162, pp. 219–234, 2019.
- [13] B. Syamsuri and G. P. Kusuma, “Plant disease classification using lite pretrained deep convolutional neural network on android mobile device,” *International Journal of Innovative Technology and Exploring Engineering*, vol. 9, no. 2, pp. 2796–2804, 2019.
- [14] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang *et al.*, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” arXiv preprint arXiv: 1704.04861, pp. 1–9, 2017.
- [15] B. Zoph, V. Vasudevan, J. Shlens and Q. V. Le, “Learning transferable architectures for scalable image recognition,” in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, pp. 8697–8710, 2018.
- [16] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, “Rethinking the inception architecture for computer vision,” in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, pp. 2818–2826, 2016.

- [17] L. T. Duong, P. T. Nguyen, C. Di Sipio and D. Di Ruscio, "Automated fruit recognition using efficientNet and mixNet," *Computers and Electronics in Agriculture*, vol. 171, pp. 1–10, 2020.
- [18] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. on Machine Learning*, Long Beach, CA, USA, pp. 6105–6114, 2019.
- [19] J. Liu, M. Wang, L. Bao and X. Li, "Efficientnet based recognition of maize diseases by leaf image classification," in *Proc. Journal of Physics: Conf. Series*, Inner Mongolia, China, pp. 012148, 2020.
- [20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv: 1409.1556, pp. 1–14, 2014.
- [21] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, pp. 770–778, 2016.
- [22] B. Liu, C. Tan, S. Li, J. He and H. Wang, "A data augmentation method based on generative adversarial networks for grape leaf disease identification," *IEEE Access*, vol. 8, pp. 102188–102198, 2020.
- [23] J. Chen, D. Zhang, Y. A. Nanekaran and D. Li, "Detection of rice plant diseases based on deep transfer learning," *Journal of the Science of Food and Agriculture*, vol. 100, pp. 3246–3256, 2020.
- [24] F. Iandola, M. Moskewicz, S. Karayev, R. Girshick, T. Darrell *et al.*, "Densenet: Implementing efficient convnet descriptor pyramids," arXiv preprint arXiv: 1404.1869, pp. 1–11, 2014.
- [25] J. Chen, D. Zhang and Y. Nanekaran, "Identifying plant diseases using deep transfer learning and enhanced lightweight network," *Multimedia Tools and Applications*, vol. 79, pp. 31497–31515, 2020.
- [26] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, pp. 4510–4520, 2018.
- [27] M. Agarwal, S. K. Gupta and K. Biswas, "Development of efficient CNN model for tomato crop disease identification," *Sustainable Computing: Informatics and Systems*, vol. 28, pp. 1–12, 2020.
- [28] J. Chen, D. Zhang, M. Suzaudola, Y. A. Nanekaran and Y. Sun, "Identification of plant disease images via a squeeze and excitation mobileNet model and twice transfer learning," *IET Image Processing*, vol. 15, pp. 1115–1127, 2020.
- [29] J. P. Vasconez, J. Delpiano, S. Vougioukas and F. A. Cheein, "Comparison of convolutional neural networks in fruit detection and counting: A comprehensive evaluation," *Computers and Electronics in Agriculture*, vol. 173, pp. 105348, 2020.
- [30] A. Krizhevsky, I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. 25th Int. Conf. on Neural Information Processing Systems*, vol. 1, Lake Tahoe, Nevada, 2012.
- [31] J. Deng, W. Dong, R. Socher, L. -J. Li, K. Li *et al.*, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Miami, FL, USA, pp. 248–255, 2009.
- [32] P. Ramachandran, B. Zoph and Q. V. Le, "Searching for activation functions," arXiv preprint arXiv: 1710.05941, pp. 1–13, 2017.
- [33] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis *et al.*, "Tensorflow: A system for large-scale machine learning," in *Proc. 12th USENIX Symposium on Operating Systems Design and Implementation*, Savannah, GA, USA, pp. 265–283, 2016.