Journal of Agricultural Sciences (Tarim Bilimleri Dergisi) 2023, 29 (4) : 1003 – 1017 DOI: 10.15832/ankutbd.1230265

Journal of Agricultural Sciences
(Tarim Bilimleri Dergisi)

J Agr Sci-Tarim Bili
e-ISSN: 2148-9297
jas.ankara.edu.tr

JAS

Transfer Learning based Image Classification of Diseased Tomato Leaves with Optimal Fine-Tuning combined with Heat Map Visualization

Sandhya Devi RAMIAH SUBBURAJ[a*], Vijay Kumar VAITHYAM RENGARAJAN[b], Sivakumar PALANISWAMY[c]

[a]*Department of Electrical and Electronics Engineering, Kumaraguru College of Technology, Coimbatore - 641049, INDIA*
[b]*Department of Electronics and Communication Engineering, Anna University Regional Campus Coimbatore - 641046, INDIA*
[c]*Department of Electrical and Electronics Engineering, PSG College of Technology, Coimbatore - 641004, INDIA*

ARTICLE INFO
Research Article
Corresponding Author: Sandhya Devi Ramiah Subburaj, E-mail: sandhyasubburaj@gmail.com
Received: 06 January 2023 / Revised: 25 March 2023 / Accepted: 28 March 2023 / Online: 24 Oct 2023

Cite this article
RAMIAH SUBBURAJ S D, VAITHYAM RENGARAJAN V K, PALANISWAMY S (2023). Transfer Learning based Image Classification of Diseased Tomato Leaves with Optimal Fine-Tuning combined with Heat Map Visualization. *Journal of Agricultural Sciences (Tarim Bilimleri Dergisi),* 29(4): 1003-1017. DOI: 10.15832/ankutbd.1230265

ABSTRACT
Plant disease detection and disease classification at initial stages for sensitive commodities like tomatoes and potatoes is highly mandated as the harvest losses have a direct impact on the price fixation of the vegetables. The most identified limitation in the study of plant pathology is the availability of datasets with visual symptoms that covers all the possible diseases of one crop or plant species. Computer Vision systems and advancements in deep learning-based modeling methodologies gained significant attention in smart farming. It is presumed that the implementation of deep learning algorithms demands a large amount of data to learn complex features automatically and this can pose a challenge for applications with lesser data to achieve generalization. In such cases, Transfer Learning with optimum regularization techniques and fine-tuning mechanisms is the solution to overcome the limitations of smaller datasets. The objective of the work is to develop Tomato Disease Classification System using a transfer learning approach for ten tomato disease classes of the PlantVillage dataset downloaded from the Kaggle platform. Inception V3, a pre-trained transfer learning model is used to classify this multi-class, imbalanced, tomato plant disease based on the leaf symptoms such as dark brown lesions, concentric rings, etc. Geometrical data augmentation is used as a regularization technique to expand the size of the dataset. Significant improvement in the performance metrics is observed when the finetuning is optimum. The training accuracy and validation accuracy of the model before and after fine-tuning are 97.08%, 83.52%, and 98.19%, 95.93% respectively. The average accuracy with augmentation and optimal fine-tuning is 98%. In addition, prediction scores in terms of precision, recall, and F1-score are obtained to visualize the rate of mispredictions across the disease classes. It is observed that the misprediction rate is high across the classes early blight, late blight, and Septoria spot due to similar visual symptoms. Further, activations are used to generate an attention map in the form of Heat Maps which are included as a post-processing step before the classification of the output. Plant Leaf Disease Classification- A web application is deployed using Streamlit Python library and Ngrok services.

Keywords: Image classification, Transfer learning, InceptionV3, Attention maps

1. Introduction

Tomato (Solanum lycopersicum) is the most profitable commercial crop in India. This is a sensitive commodity as the harvest losses have a direct impact on the price fixation of the vegetables. Early disease symptoms appear on leaves and if it is potentially identified it prevents spreading. It is observed that Tomato Leaf Curl Virus has a dreadful impact (Yang et al. 2019) and leads to yield losses. The most identified limitation in the study of plant pathology is the availability of datasets with visual symptoms that covers all the possible diseases pertaining to one crop or plant species. Investigations prove that effective learning happens from intermediate to higher order layers in terms of statistical strength both qualitatively and quantitatively (Bengio et al. 2011). Transfer learning in computer vision applications is based on the previous insight which re-uses a model pre-trained on large image datasets. The knowledge learned from the pre-trained model (Pan et al. 2009) using Transfer Learning (TL) approach works on a lesser number of images (Hussain et al. 2018) and the training time is also reduced substantially. This TL approach lies under two broad categories namely Homogeneous and Heterogeneous transfer learning (Wang et al. 2019) (Zhuang et al. 2020). A homogeneous method aims at reducing the difference between marginal and conditional probabilities of source and target domains and heterogeneous transfer learning (Sukhija et al. 2006). aims at reducing the gap between the feature spaces of the source and the target. The above categories can further be subdivided into a few more classes of approach and one such is feature – based transfer learning. This approach applies to heterogeneous and homogeneous (Feuz & Cook 2015) (Oruba et al. 2014) transfer learning. The heterogeneous Transfer Learning approach is used in this tomato disease classification system.

Frontiers in plant disease classification include the use of typical ML models (Saleem et al. 2020) such as Support Vector Machines (SVM), Decision Tree (DT), K-Nearest Neighbours (K-NN), etc. These models have prominent results with small-scale datasets and the study signifies the need for qualitative and quantitative analysis of disease classification. Since CNN models automatically extract features with no complex pre-processing steps, most of the investigations use either CNN or CNN-enhanced models for disease prediction. Segmentation-based study of lesions and their geometric properties using a sliding window is used to highlight the affected leaf portions. Histogram Equalization, Colour to grey conversion, K-Means clustering, Discrete Wavelet Transforms, and Contouring are also used along with CNN models and SVMs for image classification (Harakannanavar et al. 2022) (Liu et al. 2021). However, misinterpretation occurs as leaves rapid color change occurs due to environmental conditions. SVM classifier for Tea leaves disease detection through feature reduction technique (Hossain et al. 2018) Regional-CNN to segregate weeds from paddy farms based on spatial information (Saleem et al. 2020) semantic segmentation and encoder – decoder model for weed separation (Guo et al. 2018) are also, some of the DL models and methods used in precision agriculture using image recognition. These models however lack quantitative analysis across each disease class such as misprediction rate. A recent study on the classification and prediction of (Jiang et al. 2019) diseased apple plants based on visual symptoms conclude that CNN based Inception Module provides enhanced classification performance on background clutter, occlusion in leaf images, and poor lighting environment (Astani et al. 2022). Inception V3 is a 48- layer dense CNN-based TL model and the model's capability in learning feature representations through the TL approach is studied. It is observed that recognition of cervical cancer cell structures using InceptionV3 (Dong et al. 2020) based on the TL approach has a performance outcome of 98% with better generalization. TL approach-based food images classification usingInceptionV3 has an enhanced classification performance of 98% in 100 epochs (Goh et al. 2021) Another work on face mask detection (Jignesh Chowdary et al. 2020) using InceptionV3 based on the TL approach has a classification accuracy of 99% on masked face dataset. Pulmonary classification using images based on TL and InceptionV3 (Wang et al. 2019) methodology has attained improved sensitivity and specificity scores. With the above insights, the work on the identification of leaf disease symptoms of the Tomato plant using the InceptionV3 model is performed. This multi-class image dataset downloaded from the Kaggle platform covers frequently occurring disease symptoms on tomato leaves such as Bacterial spots, Early Blight, Late Blight, Spider Mites, Target Spots, Tomato Yellow Leaf Curl Virus, and so on. The diseased leaf image dataset is trained using pre-trained weights of InceptionV3 on the target network (Nguyen et al. 2018) (Zhang et al. 2018), and the performance of leaf disease classification is measured. This image classification problem is also deployed as a web-based application using Streamlit python library and Ngrok cloud services. The main contribution of the proposed investigation involves:

- Developing and validating InceptionV3 model through a transfer learning approach to closely suit the real-time field scenarios with uneven distribution of images across disease classes and variable image resolutions.
- Signifying the role of the transfer learning approach for applications with lesser data since it is presumed that deep learning-based investigations demand larger datasets.
- Evaluating heterogeneous transfer learning method between cross domains with appropriate performance indicators.

2. Proposed Methodology

Fine-tuning of a way to transfer learning is performed based on the weights of the previously trained layer to minimize the loss during the training process. The final feature map layer provides the state of the model being trained and it is later fine-tuned and flattened, and the output is fed to the end fully connected layer of the classifier model. A dropout layer is added to the hidden layers for regularization. This form of regularization achieves optimum performance that minimizes the variance in the validation set by preventing dense co-adaptations on training data. NumPy-based argmax prediction is used on random test images to predict the disease class and other performance metrics such as precision, recall, and F1 score. To visualize the performance of the model in the between-layers activation maps as heat maps are being generated for the layer just before the last layer of the model. Th classifier developed is deployed as a web application using Streamlit, a python library. The required libraries are imported, and the application is developed using %%writefile app.py, and the features required on the webpage being defined. The trained model is loaded, to predict the class of output on random images downloaded from the internet. This web application development depends on Ngrok, which enables cross-platform application development, and it is used as a tunnel to the Streamlit port. The technical elaborations of all the key modules are discussed in the forthcoming sections. The flow of the entire process carried out in this paper is shown as block diagram in Figure 1.

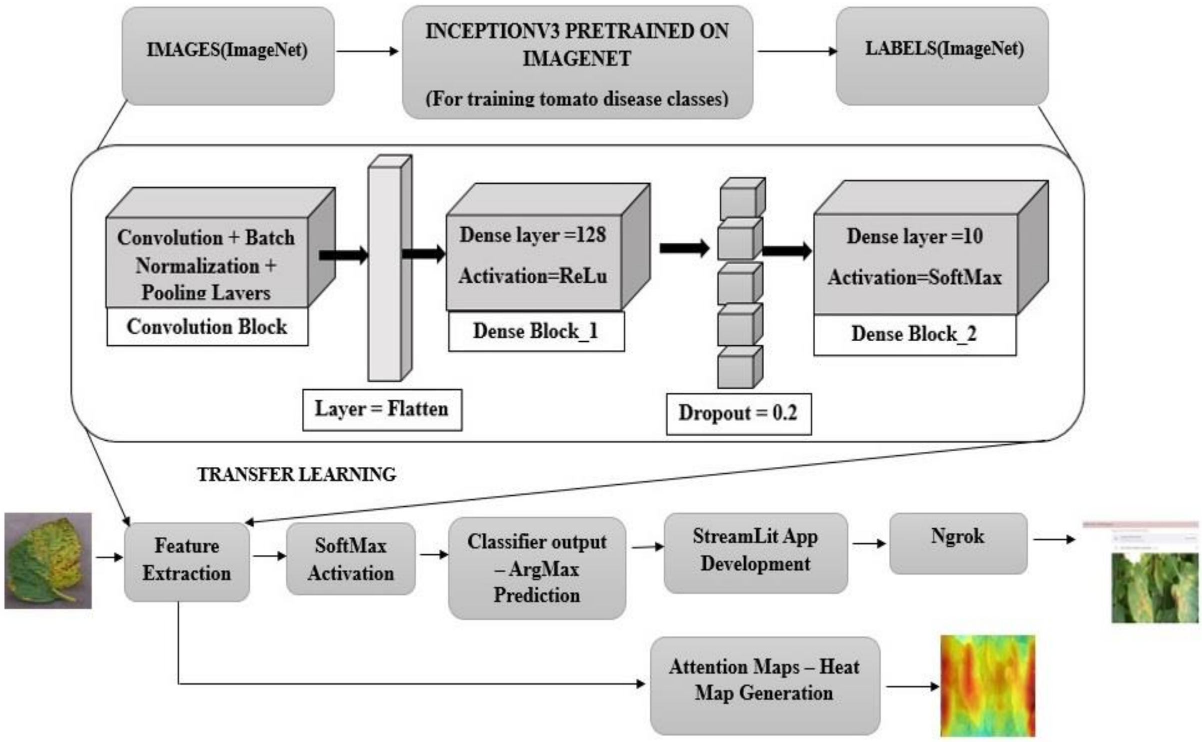


Figure 1- Image Classification of Diseased Tomato Leaves – Methodology

3.1. Convolution Neural Networks (CNN), the base model of Inception V3

3.1. Transfer learning – Notation, definition and approach used

The objective of the transfer learning is to improvise the conditional probability distribution P(Yt|Xt) of the target domain Dt (corresponding learning task is Tt) with the information learned from source domain Ds (corresponding learning task is Ts). Here (Dt≠Ds) and (Tt≠Ts).

In a more generalized way, when two tasks are different the respective label spaces are also different (Yt≠Ys), and its conditional probability distribution (Wiatowski & Bölcskei 2017) is also different. (P(Yt|Xt) ≠P(Ys |Xs)).

In this paper, a feature-based heterogeneous transfer learning approach is performed on the Inception V3 model pre-trained on the ImageNet dataset. The features extracted from the final convolution layer of the InceptionV3 model being utilized for transfer learning. The early convolution layers are frozen as these layers will extract more general or low-level features and as training happens the later layers are focused on specific features.

3.2. InceptionV3 model as image feature extractor

ImageNet is a research resource for the computer vision domain that can label, recognize, and classify around 22,000 object categories. It is one of the largest anthologies of images which is around 10 million with 3.2 million cleanly annotated images as of 2020. This ImageNet fosters the development of many robust and sophisticated models and one such is the Inception V3 model.

The key concept of InceptionV3 is to reduce the computational costs (i.e) the number of deeper networks say filters of size 5x5 or 7x7 into 1x7 and 1x5 smaller filters of asymmetric size as shown in Figure 2 used. A 1x1 filter is used to reduce the channel depth (Szegedy et al. 2016). For example, a 1x1 filter is used to reduce 100 M into 10M by a factor of 10 and thus shrinks the number of channels.

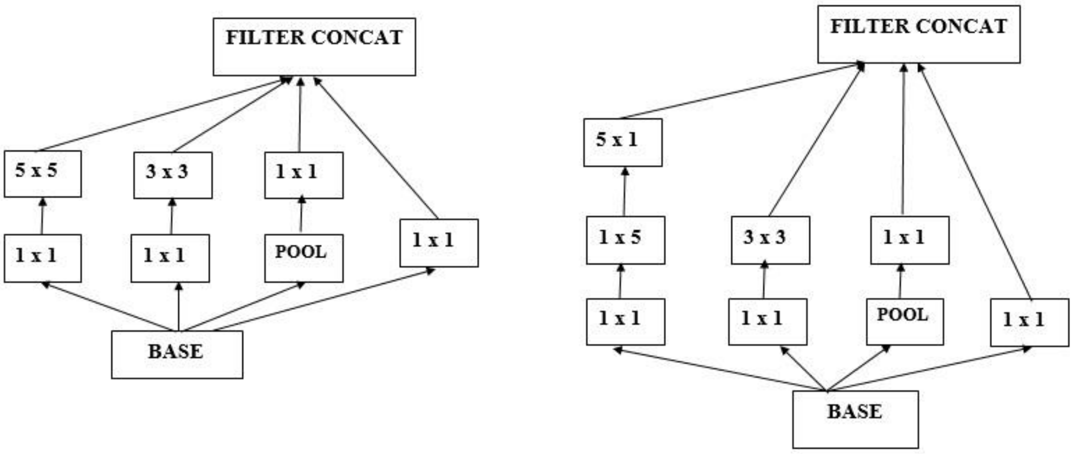


Figure 2- InceptionV3 - Symmetric Factorization and fig(b)InceptionV3 Asymmetric Factorization where 5x5 block is Factorized into 5x1 and 1x5.

The model has attained around 78% accuracy in 170 epochs on the ImageNet dataset and it is a culmination of symmetrical and asymmetrical layers consisting of convolution layers, average and max pooling layers, SoftMax for loss computation, and batch normalization is used throughout for stabilizing the process. In the InceptionV3 factorization model, batch normalization is used in auxiliary classifiers. The role of the auxiliary classifier is it acts as a small CNN inserted during the training and the loss incurred due to the inclusion of this classifier will be added to the main loss. But in Inception V3, this auxiliary classifier acts as a regularizer (Zhang et al. 2019) which aids the loss module to avoid or prevent over-fitting.

The term Accuracy is the ratio of the number of correct predictions to the total number of predictions. This performance parameter assesses the model's ability to function across various classes, the relationship between each parameter, and pattern prediction. The InceptionV3 model has attained 78% Top-1 accuracy on the ImageNet validation dataset. Further, the Top-5 Accuracy is extended to 93.7%. The flip-flop of accuracy is the error parameter and for having the good insight into the model, the accuracy parameter provides a comprehensive perspective. The error rate in terms of accuracy is given as Error Rate= 1-Accuracy. The Top-1 and Top-5 accuracy on ImageNet Validation dataset for various models is given in Table 1.

Table 1- Top-1 and Top-5 accuracy of various models

S.No.	*Model*	*Number of Parameters*	*Top-1 accuracy*	*Top-5 Accuracy*
1	VGG-16	138 357 544	71.8%	90.1%
2	Inception V3	23 851 784	77.9%	93.7%
3	ResNet50	25 636 712	74.9%	92.1%
4	AlexNet	62 378 344	63.3%	84.6%
5	GoogLeNet	23 000 000	74.8%	92.2%
6	InceptionResNetV2	55 873 736	80.3%	95.3%
7	ResNet-152	25 000 000	78.57%	98.2%
8	DenseNet	8 062 504	76.39%	93.34%
9	Xception	22 910 480	79.00%	94.5%

4. Image Dataset, Diseased Leaf Symptoms and Data Augmentation

4.1. Tomato- Diseased image dataset

The Plant Village dataset contains 38 categories of healthy and unhealthy leaf images of apples, potatoes, pepper plants, etc. This paper focuses on 10 categories of Tomato Leaf classes inclusive of the healthy tomato images the train folder consists of 14472 images and the test set consists of 3616 images, both belonging to 10 classes. The distribution of images under 10 categories is shown in Figure 3.

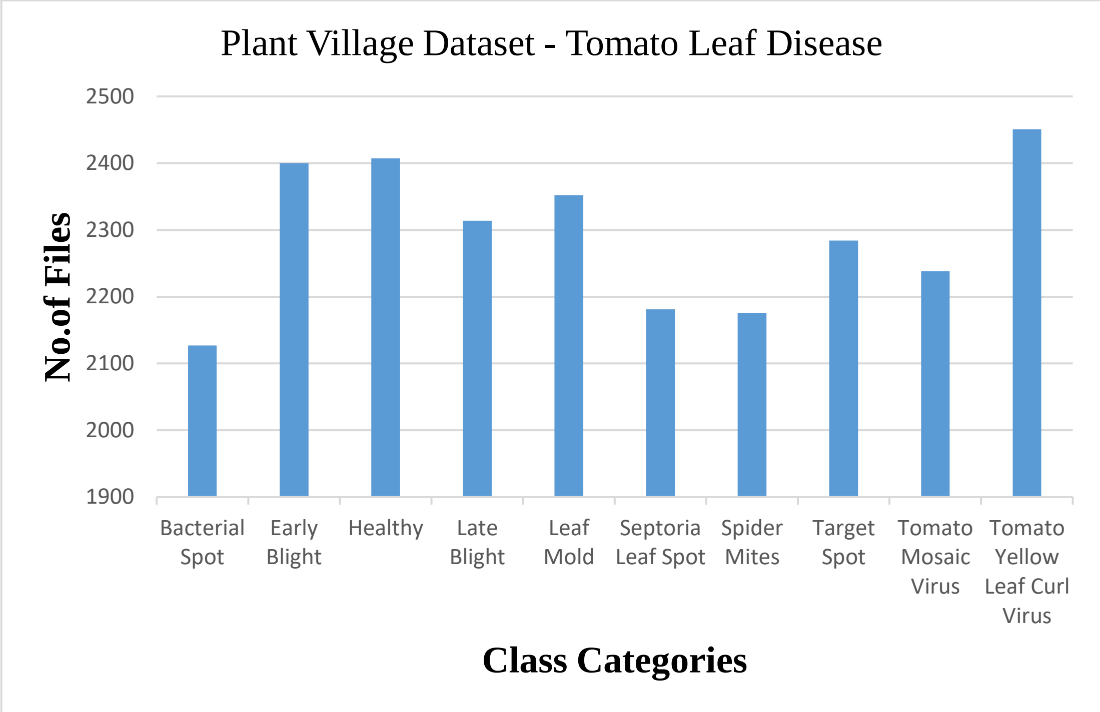


Figure 3- Distribution of Images of PlantVillage Dataset focusing on Tomato Leaf Disease under various classes

4.2. Image data augmentation

Image augmentation is carried out as an initial step to prevent over-fitting and to generalize the model on the output classes (Shorten & Khoshgoftaar 2019). Random and appropriate transformations such as flips, shifts, and zooms are performed on the actual images of the PlantVillage dataset using Keras ImageDataGenerator and it is shown in Figure 4. The data generators act as inputs to the model which also performs normalization operations on the augmented dataset. To create train and test generators flow_from_directory is used.

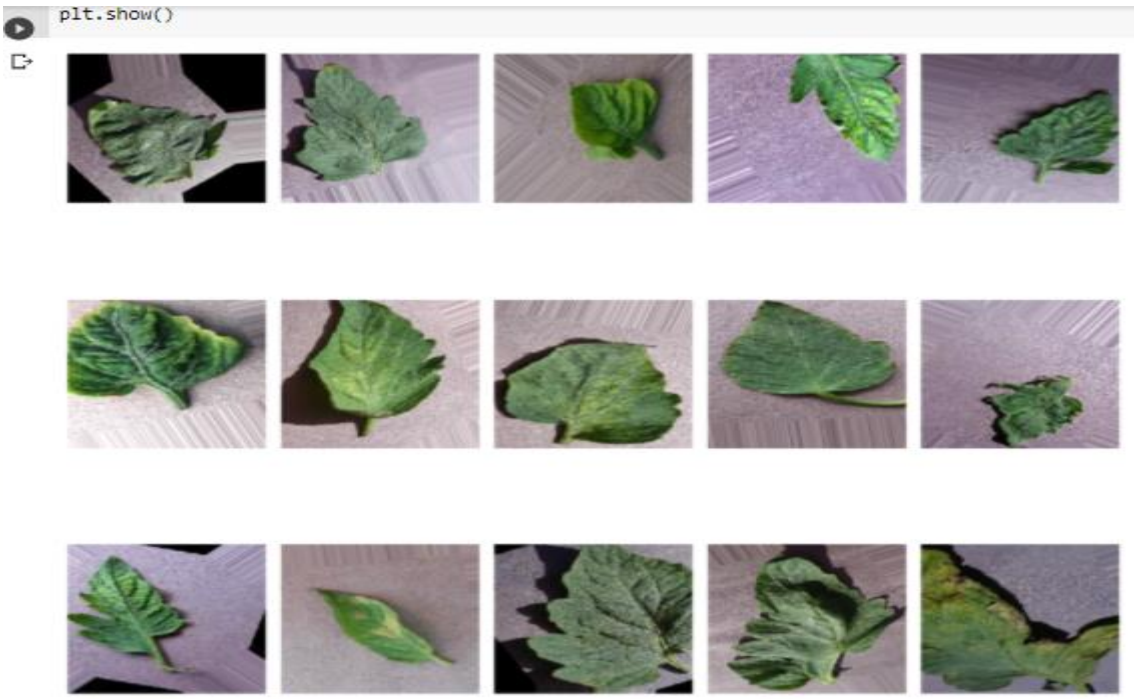
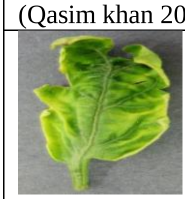


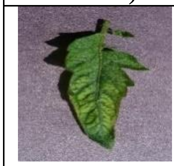
Figure 4- Results of Data Augmentation on a Training Image

4.3. Diseased tomato leaf visual appearance (Symptoms)

Early symptoms appear on the leaves with some changes in foliar structures. Early blight begins with dark brown lesions beneath the leaves and in the case of late blight wet blotches appear it is identified as one of the most destructive diseases since this blight is airborne and the spores spread at a faster rate. Septoria leaf spot and early blight may occur simultaneously. Bacterial spot is another disease with wet large blotches on matured plants and it usually appears on the leaf margins. Thus, each disease has its unique symptoms caused by various pathogens. The 10 broad classes in the image dataset used for the analysis and their visual symptoms are shown in Table 2.

Table 2- Diseased Tomato Leaves and their Visual Symptoms

 Bacterial Spot (Qasim khan 2022)	Scientific Name: *Xanthomonas (X. euvesicatoria, X. gardneri, X. perforans, and X. vesicatoria)* • The spots appear as wet-looking circular areas (Potnis et al. 2015) (Osdaghi et al. 2021) with the scabby wart-like surface. • Initially starts with yellow green discoloration and later turns into brownish red. Since plants with bacterial spots cannot be cured, infected plants should be identified and removed to prevent further spread.
 Early Blight (Qasim khan 2022)	Scientific Name: Alternaria solani • Discoloured spots or rings and a few small brown lesions (Adhikari et al. 2017) appear on the leaves. • Does not affect the fruits initially if it is potentially identified and treated to prevent spreading. • Known as Bull's eye disease as it starts with the appearance of spots and two concentric rings around in matured leaves. These concentric rings can spread to stems and fruits and further the spots can combine and make patches.
 Late Blight (Qasim khan 2022)	Scientific Name: *Phytophthora infestans* (Montagne) Bary • Steady brown spots (Mazumdar et al. 2021) that cover a major part of the fruit • Irregular spots which turn mushy and dark brown or blackish purple lesions • Traces of white fungal growth
 Leaf mold (Qasim khan 2022)	Scientific Name: Passalora fulva • Foliage, pale green, and yellow spots on the upper side of the leaves. The color of the upper side of the leaf is olive green and finally curls. • The leaf will have irregular borders and in severe cases, the spots enlarge, and the fruit is black with rot (Yoshida et al. 2021) in the stem.
 Septoria Leaf Spot (Qasim khan 2022)	Scientific Name: Septoria • A destructive disease where leaf spots with dark brown (Ibrahim 2019) outlines and a greyish centre appears on the lower side of the leaf. • As the disease spreads spots spread and eventually the leaf color turns yellow, later brown, and withers.
 Tomato Yellow Leaf Curl (Qasim khan 2022)	Scientific Name**:** Tomato yellow leaf curl virus (TYLCV) • Stunted growth, leaf size reduces (Yang et al. 2019) (Mariyappan et al. 2013), and leaf curls upwards. • Lead to chlorosis and finally tomato production.

 Spider mites (Qasim khan 2022)	Scientific Name: TETRANYCHUS EVANSI • This polyphagous pest disease lays eggs on the top side of the leaf (Liu et al. 2020). • It lays eggs on the bottom side and the leaf underneath turns yellow or tannin color. • This causes a blotchy color pattern.
 Target spot (Qasim khan 2022)	Scientific name: Corynesporacassiicola • Early symptoms are like early blight and many other fungal spots. • The target spot (Weeraratne et al. 2020) is concentric rings with the innermost brown lesions surrounded by yellow circles. • As the disease spreads the spots enlarge and club with other spots thereby covering a major area of the leaves.
 Tomato Mosaic Virus (Qasim khan 2022)	Scientific Name: Tobamovirus • This pathogenic virus causes irregular ripening of fruits. • The symptoms (Kubota et al. 2003) include leaves with light green and yellow mosaics on the leaves with a prominent reduction of the leaf curvatures. • The slight fern-shaped impaired affect fruit yield by 2 to 23%.

4.4. Performance metrics - Precision, recall, and F1- score

Prediction metrics is significant for evaluating the trained model on multi-class image dataset. RoC Curve is considered efficient for binary classifiers and on balanced datasets. Precision-Recall metrics and F1-Score shown in Table 4, provide better evaluation insights on class predictions irrespective of balanced or imbalanced datasets (Saito & Rehmsmeier 2015) as compared to RoC curves. True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN) shown in Table 3 are the four basic parameters required for calculating the metrics say Accuracy, Precision, Recall, and F1-Score (Liang et al. 2022) shown in Table 4.

True Positives (TP): The model predicts the positive class when both the Predicted and Actual classes are the same.
True Negatives (TN): The model predicts the negative class when both prediction and Actual are No or Negative.

False Positives (FP): Also known as Type 1 error. Wrong prediction of the Negative class occurs when the actual class is Negative, but prediction outcomes are Positive.

False Negatives (FN): Also known as Type II error. Wrong prediction of the Positive class occurs when actual outcomes are Positive, but prediction results are Negative.

Table 3- Basic Parameters required for Computing Performance Metrics

	Predicted Class			$TPR = \frac{TP}{Actual\ Positive} = \frac{TP}{TP+FN}$
Actual Class		**Class = Positive**	**Class = Negative**	$FNR = \frac{FN}{Actual\ Positive} = \frac{FN}{TP+FN}$
	Class = Positive	**True Positive**	**False Negative**	$TNR = \frac{TN}{Actual\ Negative} = \frac{TN}{TN+FP}$
	Class = Negative	**False Positive**	**True Negative**	$FPR = \frac{FP}{Actual\ Negative} = \frac{FP}{TN+FP}$

Table 4- Performance Metrics used for Model's Evaluation

Performance Metrics	
Accuracy	$\frac{TP + TN}{TP + FP + TN + FN}$
Precision	$\frac{TP}{TP + FP}$
Recall	$\frac{TP}{TP + FN}$
F1 Score	$\frac{2 * (Precision * Recall)}{(Precision + Recall)}$

The above values are used to compute True Positive Rate (TPR), False Positive Rate (FPR), True Negative Rate (TNR), and False Negative Rate (FNR). TPR is also known as Sensitivity and TNR is known as Specificity (Altman, Douglas et al. 1994).

- Accuracy: It is the ratio of correctly predicted observations to the total number of observations
- Precision: Ratio of relevant observations to the retrieved observations. It is a measure of quality.
- Recall (Sensitivity): It is the measure of total relevant results correctly classified by the model. It is a measure of quantity.
- F1 Score: It is the weighted average of Precision and Recall. The score is high only when both precision and recall are high.

4.5. Evaluation methodology

While training the data it is quite important to achieve optimum fit over the model. A model initially will learn the correlation between the input samples (x) and the target values(y). This is done by evaluating the model on the test data. In this study, the percentage of data used for training and testing is 70:30 respectively. The evaluation methodology implemented during the model training and validation phase is detailed in the following algorithm.

Tomato Disease Classification Algorithm

Input Data: Tomato Disease Dataset (PlantVillage) images (X, Y); where Y= Predicted class {y/yϵ Tomato Disease Classes}
Pre-processing steps:
Step1: Set the image size s_i and set layer. trainable = False
Step2: Import Keras Sequential Model, Layers, ImageDataGenerators, and optimizer.
Step 3: Perform Geometrical data augmentation on X images and obtain X'
Step4: Gain the information from the source domain D_s and learning task T_s of Inception V3
Step5: Apply the information learned to target Domain D_t with X' and the corresponding target learning task T_t where $D_s \neq D_t$
Step 6: Train the model for N epochs for the augmented dataset X'.
Step 7: Compute the values $y(true)$ and $y(pred)$ for the output classes.
Step 8: Estimate the deviation and apply categorical cross-entropy loss function and RMSProp optimizer on Y predicted classes.
Step 8: Fine-tune the model for M epochs to obtain a balanced fit.

5. Results and Discussion

The entire implementation is performed on Tesla T4 and the features extracted using the InceptionV3 model and Keras API is used for loading the model with pre-trained weights the model compilation requires the parameters optimizer, loss, and performance metrics. The model is optimized on the augmented multi-class dataset using a Categorical cross-entropy loss function, RMSprop optimizer with a learning rate of 0.0001 and decay rate of 1e-6. The values for both the learning rate and decay parameter is fixed based on the over-fitting results obtained during the implementation. Learning rate decay is independent of the optimizer. Model Checkpoint and the callback function is used to save the model and the tensor board information after each epoch. A definite file path is used to save the model in h5 format. The function callbacks list is used in aggregation with model. Fit function. This feature facilitates obtaining performance metrics such as training and validation for both accuracy and loss parameters. The number of epochs is set to 50 and the batch size is 32 for the model training process. The intermediate results during the training process are studied to learn the impact of data augmentation. Optimal fine-tuning is performed for additional 10 epochs based on the accuracy curves. Table 5 shows the training and validation accuracy graphs.

During the training process for the model. Fit function, the steps per epoch during the is fixed based on the length of the training samples divided by batch size and validation steps are fixed based on the length of the validation samples divided by the batch size. The dropout regularization technique of 0.7 was initially used on hidden layers. Since validation accuracy was greater than the training accuracy which depicts the over-fitting scenario, an appropriate and ideal value of 0.5 is used on the hidden layers. NumPy-based argmax and the corresponding prediction scores are initially obtained and the before and after fine-tuning results with and without image data augmentation are tabulated in Table 6. The model showed improved performance on training and validation accuracy after fine-tuning.

Table 5- Image Classification Intermediate Result – Training and Validation Accuracy

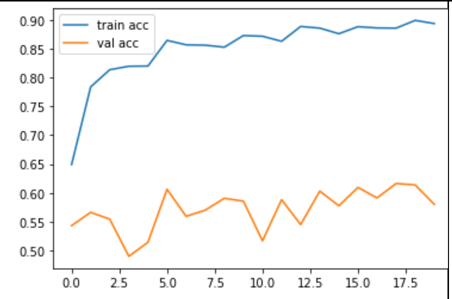
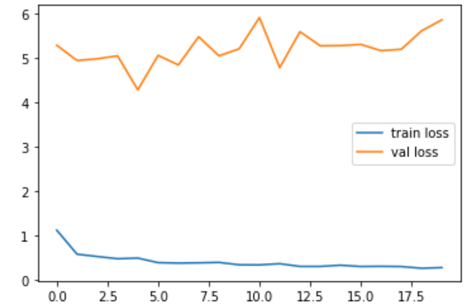
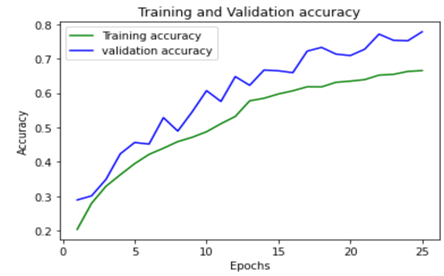
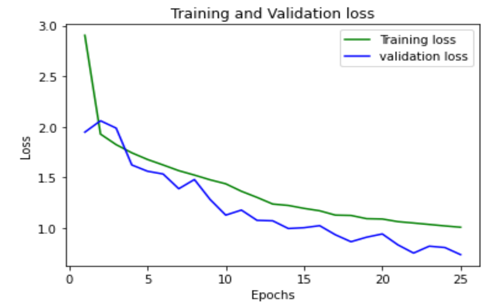
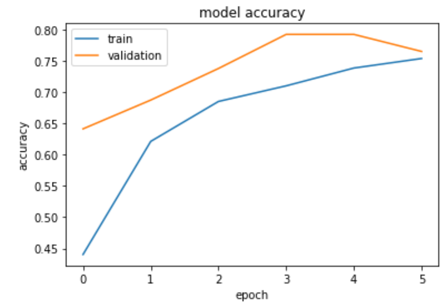
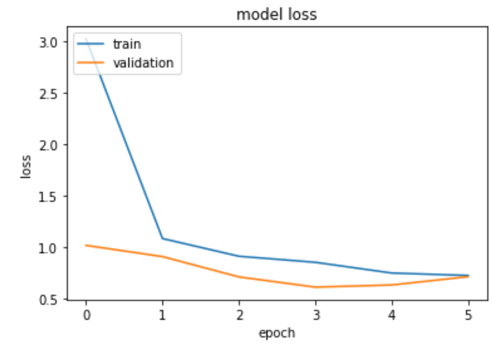
Model	*Accuracy – Training vs Validation*	*Loss – Training vs Validation*	*Observations*
Underfit (Result taken after 17 epochs)			No regularization and data augmentation used. Training accuracy of 80% is obtained. However, validation accuracy is very low.
Balanced fit (Result taken after epoch 25)			Optimal regularization with data augmentation implemented. It is observed that training accuracy is greater than validation accuracy
Over fit (Result taken after epoch 5)			The accuracy declined during the initial epochs due to high regularization and use of all geometrical augmentation techniques on every image.

Table 6- Image Classification Final Results – Before and After Finetuning

Before Finetuning	[ ] # Model evaluation
	scores_train = model.evaluate(train_generator,verbose=1)
	scores_validation = model.evaluate(validation_generator,verbose=1)
	print("Train Accuracy: %.2f%%" % (scores_train[1]*100))
	print("Validation Accuracy: %.2f%%" % (scores_validation[1]*100))
	
	68/68 [==============================] - 41s 593ms/step - loss: 0.6732 - accuracy: 0.9708
	17/17 [==============================] - 10s 605ms/step - loss: 0.9510 - accuracy: 0.8352
	Train Accuracy: 97.08%
	Validation Accuracy: 83.52%

After Finetuning	[ ] # Model evaluation
	scores_train = model.evaluate(train_generator,verbose=1)
	scores_validation = model.evaluate(validation_generator,verbose=1)
	print("Train Accuracy: %.2f%%" % (scores_train[1]*100))
	print("Validation Accuracy: %.2f%%" % (scores_validation[1]*100))
	
	44/44 [==============================] - 365s 8s/step - loss: 0.3768 - accuracy: 0.9819
	11/11 [==============================] - 90s 8s/step - loss: 0.4093 - accuracy: 0.9593
	Train Accuracy: 98.19%
	Validation Accuracy: 95.93%

In addition, the model's performance on this multi-class dataset is evaluated in terms of TP, TN, FP, and FN. The classification metrics say Accuracy, Precision, Recall and F1 score results are shown in Figure 5. The results show the highest sensitivity, precision, and F1 score is recorded against Tomato Yellow Leaf Curl Virus followed by Tomato Mosaic Virus which leads to high yield loss and require immediate attention to prevent further spreading to other plants.

```
report = metrics.classification_report(true_classes, predicted_classes, target_names=class_labels)
print(report)
```

```
                                               precision    recall  f1-score   support

                      Tomato___Bacterial_spot       0.85      0.89      0.87       340
                        Tomato___Early_blight       0.89      0.51      0.65       384
                         Tomato___Late_blight       0.71      0.88      0.79       370
                           Tomato___Leaf_Mold       0.89      0.88      0.89       325
                  Tomato___Septoria_leaf_spot       0.72      0.89      0.80       349
Tomato___Spider_mites Two-spotted_spider_mite       0.80      0.85      0.83       348
                         Tomato___Target_Spot       0.76      0.79      0.77       365
       Tomato___Tomato_Yellow_Leaf_Curl_Virus       1.00      0.87      0.93       392
                 Tomato___Tomato_mosaic_virus       0.95      0.96      0.96       358
                             Tomato___healthy       0.94      0.92      0.93       385

                                     accuracy                           0.84      3616
                                    macro avg       0.85      0.84      0.84      3616
                                 weighted avg       0.85      0.84      0.84      3616
```

Figure 5- Precision, Recall, and F1 Score

Figure 6. shows a 10x10 confusion matrix generated for 10 disease classes where each row corresponds to the predicted disease class and the column corresponds to the actual class. It is known as a statistical error matrix with results for the 10 classes generated in order (class names as listed in Figure 5) between actual values and predicted values. The matrix element at (2, 2) corresponding to Early blight shows 196 correct predictions, 68 early blight images are misinterpreted as late blight and 47 images are misinterpreted as Septoria leaf spot. Similar mispredictions due to close symptom similarities for all the 10 classes are also shown in Figure 6.

```
cm = confusion_matrix(true_classes, predicted_classes)
print(cm)
```

```
[[301   1   8   0  22   0   7   1   0   0]
 [ 32 196  68  10  47  12  18   0   1   0]
 [  2  12 326   9   5   3   8   0   4   1]
 [  1   2  12 285   9   7   4   0   4   1]
 [  5   6  14   4 310   2   5   0   2   1]
 [  0   0   5   3  10 297  25   0   3   5]
 [  7   3  12   1  11  30 288   0   1  12]
 [  6   1  10   3   9  16   4 340   2   1]
 [  0   0   1   4   3   1   3   0 345   1]
 [  0   0   2   0   3   3  19   0   2 356]]
```

Figure 6 - Precision, Recall, and F1 Score The performance of the proposed method is compared with other models, and it is tabulated in Table 7

Table 7- Comparison with Other Models

Authors	*Proposed Model*	*No. of Images/ Tomato Classes*	*Accuracy*	*Mechanism used*	*Performance Metrics*	*Augmentation*	*Fine-tuning*
Jiang Ding et al. 2020	Resnet-50	3000 images/3 classes	98	Leaky ReLu and 11 x 11 convolution layer	Accuracy	Nil	Nil
Al-gaashani, Mehdhar SAM 2022	MobileNetV2 and NASNetMobile	1152 images / 6 classes	97%	Concatenated features of both classifiers used.	Accuracy	Nil	Nil
Agarwal, Mohit, Abhishek Singh et al. 2020	Modified CNN architecture	10 classes	91.2	Augmentations on Modified architecture	Accuracy	✓	Nil
Basavaiah, Jagadeesh, et al. 2020	Decision tree classifier	9 classes	90	-	Accuracy	Nil	Nil
	Random forest classifier		94	-	Accuracy	Nil	Nil
Rangarajan, Aravind Krishnaswamy et al. 2018	VGG16	13 262	97.29	Accuracy dropped as weight and bias learning rate is increased.	Accuracy	Nil	✓
	AlexNet	13,262	97.49	Reduced execution time but fine-tuning decreased performance	Accuracy	Nil	✓
Our Method Transfer Learning Approach	**InceptionV3**	**18 088/10 classes**	**98**	**Optimal Finetuning and Regularization Method improved the performance**	**Accuracy, Precision, Recall, F1-Score**	✓	✓

Attention maps aid in visualizing the models learning traits on the extracted high- and low-level features of an image. These interpretable attention maps provide us an insight into the local fine-grained details which are significant in enhancing the model's accuracy. Attention heat maps are generated using Activation Mapping. For a given input image, obtain the output activations (Zagoruyko & Komodakis 2016) in the form of (H', W', C) where C represents the number of channels. To obtain the spatial attention map g which focuses on the activations within the same layer, convert the image into (H', W'). This attention map is applied to the last convolution layer. The activation A is given as, $A_i = A\ [;\ ;\ i]$. Here I represents the index values of the channel dimension. Then the evaluation (Zhou et al. 2016) is given as

$$g(A) = \sum_{i=1}^{C} |A_i| \tag{1}$$

For IncepionV3 used in plant disease classification, the heat maps are generated for the last convolution layer and a test image of class early blight is used for which the visual leaf disease symptom is very poor. This scenario requires insight into the model's interpretation in the between layers. Compared to the initial heat maps the later heat map images shown in Figure 7 focus more on the weak early blight symptoms and the leaf boundaries as the actual image has shadowing effects.

To examine the generalization of the model's performance, a random image, is downloaded from google images, and heat map visualization is shown in Figure 8. The last rows of the heat maps are centered on the visual symptoms and the leaf boundaries.

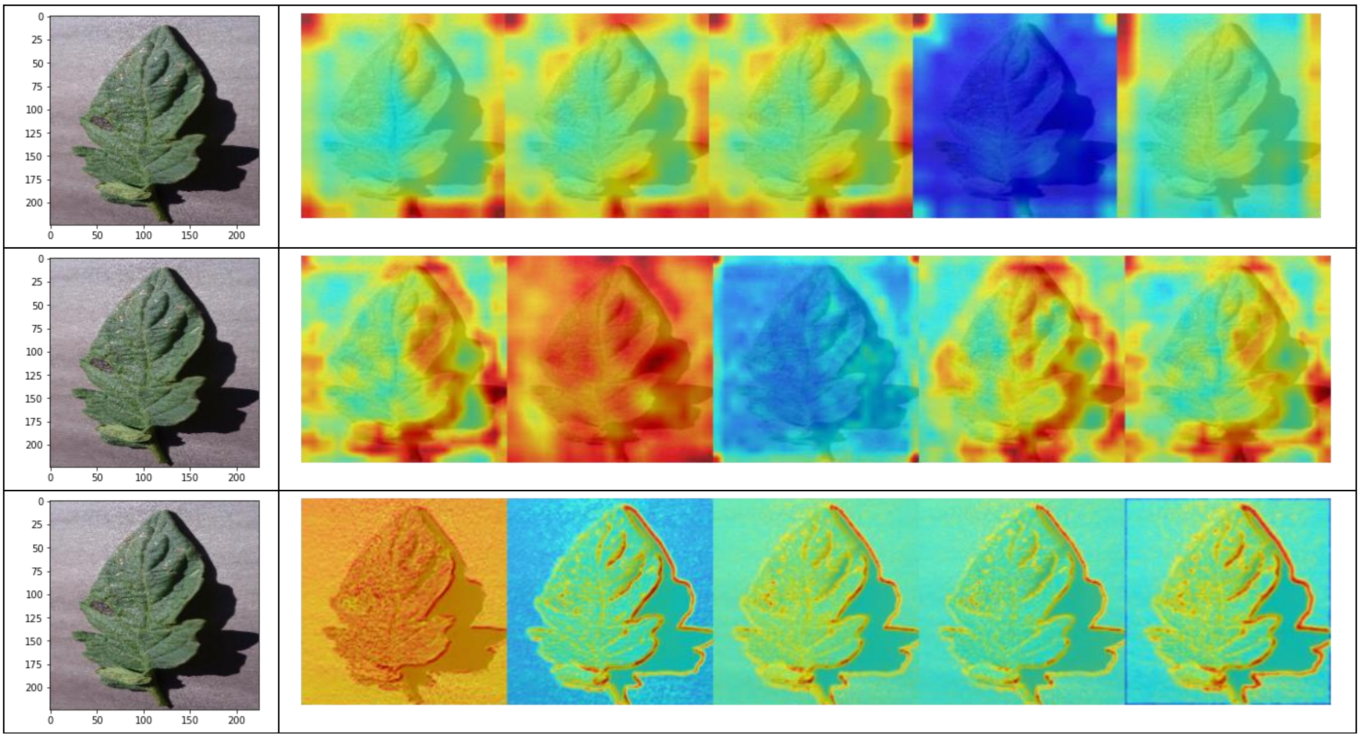


Figure7- Heat Map Generations on a Test Image

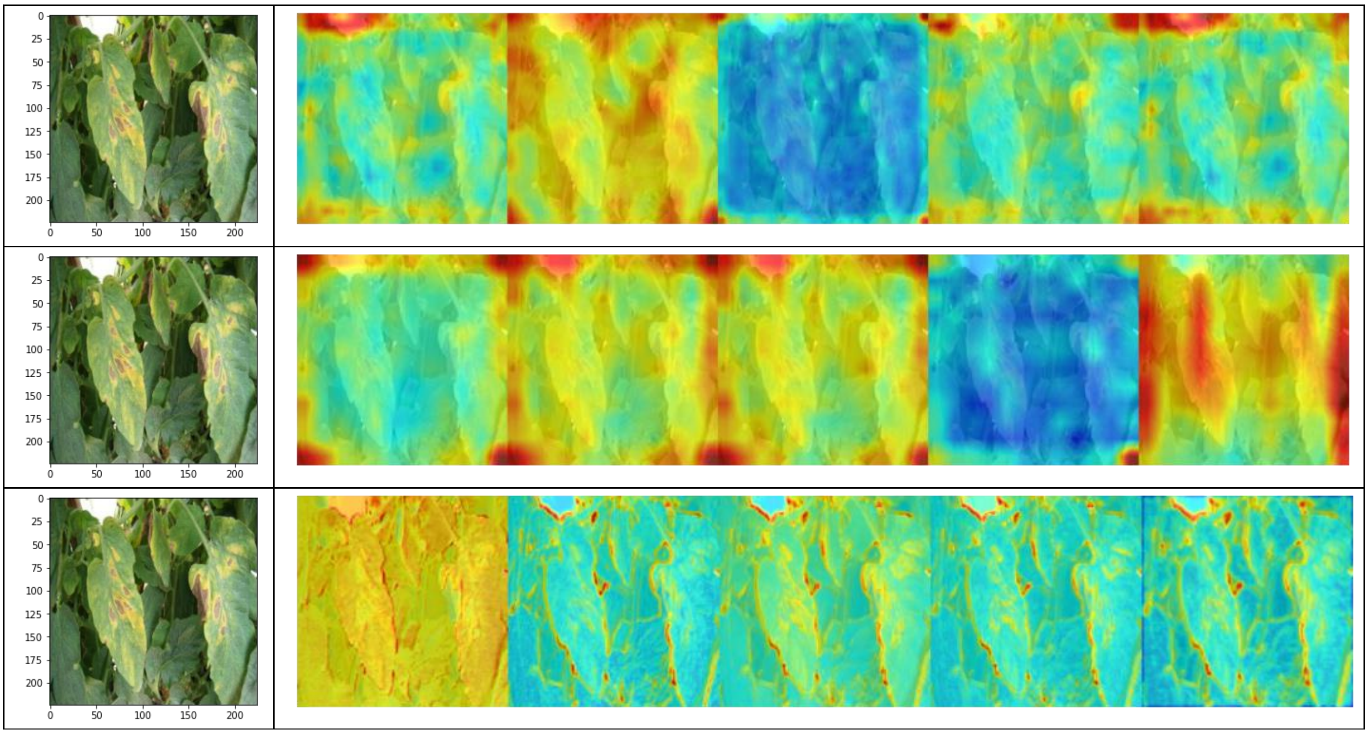


Figure 8- Heap Map Generations on a Random Image

The same random image used in Figure 7 is used as a test image for the web application developed using the Streamlit python library. Ngrok provides a secure tunnel for which an authentication token shown in Figure 7 is obtained and the. yml configuration file is available.

The Ngrok is tunneled to port number 8501. The required web page application features are written using %%writefile app.py and is shown in Figure 9.

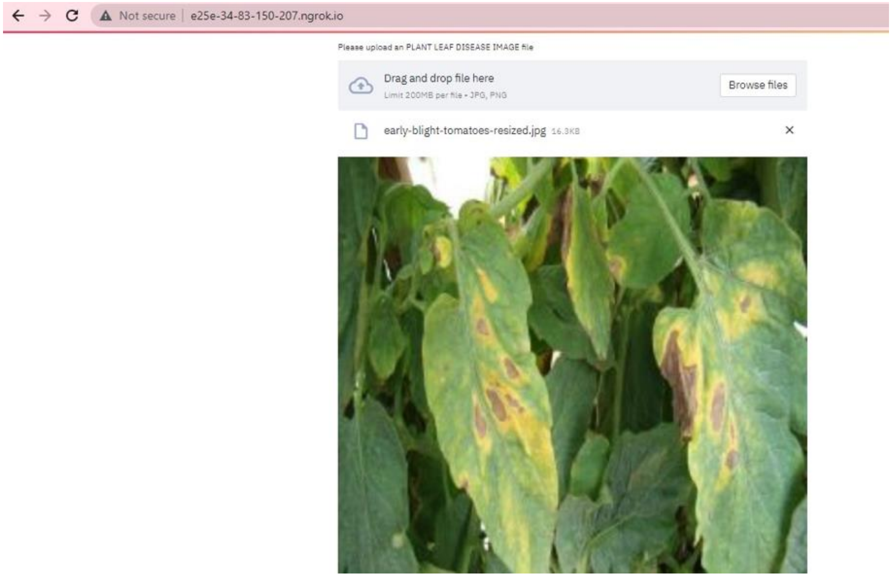


Figure 9- Plant Leaf Disease Classification: Web-based Application Showing Random Images Uploaded from the Internet

6. Conclusions

This paper investigates on the effectiveness of the Transfer Learning approach for classification systems with limited data for cross- domains. The investigation and comparison with related works proves InceptionV3's classification potential is enhanced for multi-class, slightly imbalanced dataset with appropriate regularization and fine-tuning mechanisms. The classification report and confusion matrix show that the highest classification scores are reported for the disease classes Tomato Yellow Curl Virus and Mosaic Virus. Maximum mispredictions due to similar visual symptoms are recorded between Early Blight, Late Blight, and Septoria leaf spot. The overall training and validation accuracy before and after fine-tuning are 97.08%, 83.52% and 98.19%, 95.93% respectively. The results also show that implementing sophisticated methods to improve accuracy during the training process led to over-fitting situations. To evaluate and understand this kind of scenario and the model's insight during the process of training requires visualization attention such as Heat map generation. The classifier model is deployed as a web application. As the next step in this process, the attention mechanisms must be further examined by combining convolution mechanisms and transformer encoders such as Vision Transformer for the areas like image classification and image captioning.

Data availability statement

The 10 tomato disease classes are downloaded from the Kaggle Platform (Link: https://www.kaggle.com/datasets/cookiefinder/tomato-disease-multiple-sources). This is a publicly available dataset with the license CC0: Public Domain.

Disclosure statement

The authors received no specific funding for this study and declare that they have no conflicts of interest to report with the present study.

References

Adhikari P, Oh Y & Panthee D R (2017). Current status of early blight resistance in tomato: an update. International journal of molecular sciences 18(10): 1-22 https://doi.org/10.3390/ijms18102019

Agarwal, Mohit, Abhishek Singh, Siddhartha Arjaria, Amit Sinha & Suneet Gupta (2020). ToLeD: Tomato leaf disease detection using convolution neural network. Procedia Computer Science 167(2020): 293-301. https://doi.org/10.1016/j.procs.2020.03.225

Al-gaashani, Mehdhar S A M, Fengjun Shang, Mohammed S A Muthanna, Mashael Khayyat & Ahmed A. Abd El-Latif (2022). Tomato leaf disease classification by exploiting transfer learning and feature concatenation. IET Image Processing 16(03): 913-925. https://doi.org/10.1049/ipr2.12397

Altman, Douglas G & J Martin Bland (1994). Diagnostic tests. 1: Sensitivity and specificity. BMJ: British Medical Journal 308(6943): 1552. doi: 10.1136/bmj.308.6943.1552.

Astani M, Hasheminejad M & Vaghefi M (2022). A diverse ensemble classifier for tomato disease recognition. Computers and Electronics in Agriculture, 198, 107054. https://doi.org/10.1016/j.compag.2022.107054

Basavaiah, Jagadeesh & Audre Arlene Anthony (2020). Tomato leaf disease classification using multiple feature extraction techniques. Wireless Personal Communications 115(1): 633-651

Bengio Y, Bastien F, Bergeron A, Boulanger–Lewandowski N, Breuel T, Chherawala Y & Sicard G (2011). Deep learners benefit more from out-of-distribution examples. In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics. JMLR Workshop and Conference Proceedings, pp. 164-172

Dong N, Zhao L, Wu C H & Chang J F (2020). Inception v3 based cervical cell classification combined with artificially extracted features. Applied Soft Computing 93: 106311. https://doi.org/10.1016/j.asoc.2020.106311

Feuz K D & Cook D J (2015). Transfer learning across feature-rich heterogeneous feature spaces via feature-space remapping (FSR). ACM Transactions on Intelligent Systems and Technology (TIST) 6(1): 1-27. https://doi.org/10.1145/2629528.

Goh A M & Yann X L (2021). Food-image Classification Using Neural Network Model. Int. J. of Electronics Engineering and Applications 9(3): 12-22

Guo Y, Liu Y, Georgiou T & Lew M S (2018). A review of semantic segmentation using deep neural networks. International journal of multimedia information retrieval 7(1): 87-93. https://doi.org/10.1007/s13735-017-0141-z

Harakannanavar S S, Rudagi J M, Puranikmath V I, Siddiqua A & Pramodhini R (2022). Plant leaf disease detection using computer vision and machine learning algorithms. Global Transitions Proceedings 3(1): 305-310. https://doi.org/10.1016/j.gltp.2022.03.016.

Hossain S, Mou R M, Hasan M M, Chakraborty S & Razzak M A (2018). Recognition and detection of tea leaf's diseases using support vector machine.2018 IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA): 150-154. doi: 10.1109/CSPA.2018.8368703

Zhang H, Zhang L & Jiang Y (2019). Overfitting and Underfitting Analysis for Deep Learning Based End-to-end Communication Systems. 11th International Conference on Wireless Communications and Signal Processing 1-6. doi: 10.1109/WCSP.2019.8927876

Hussain M, Bird J J & Faria D R (2018). A study on cnn transfer learning for image classification. In UK Workshop on computational Intelligence UK, Springer, Cham pp. 191-202. https://doi.org/10.1007/978-3-319-97982-3_16

Ibrahim H (2019). Susceptibility Studies on Two Varieties of Tomato (Lycopersicon esculentum) to Fungal Leaf Spots. EAS Journal of Nutrition and Food Sciences 1(1): 11-15. doi: 10.36349/easjnfs. 2019.v01i01.003

Jiang P, Chen Y, Liu B, He D & Liang C (2019). Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. IEEE Access 7(1): 59069-59080. doi: 10.1109/ACCESS.2019.2914929

Jiang, Ding, Fudong Li, Yuequan Yang & Song Yu (2020). "A tomato leaf diseases classification method based on deep learning." Chinese control and decision conference (CCDC) pp. 1446-1450. IEEE, 2020. DOI: 10.1109/CCDC49329.2020.9164457

Jignesh Chowdary G, Punn N S, Sonbhadra S K & Agarwal S (2020). Face mask detection using transfer learning of inceptionv3. In Big Data Analytics: 8th International Conference, BDA 2020, Sonepat, India, December 15–18, 2020, Proceedings 8: 81-90. Springer International Publishing. https://doi.org/10.1007/978-3-030-66665-1_6

Kubota K, Tsuda S, Tamai A & Meshi T (2003). Tomato mosaic virus replication protein suppresses virus-targeted posttranscriptional gene silencing. Journal of virology, 77(20): 11016-11026. https://doi.org/10.1128/JVI.77.20.11016-11026.2003

Liu J, Chafi R, Legarrea S, Alba J M, Meijer T, Menken S B & Kant M R (2020). Spider mites cause more damage to tomato in the dark when induced defenses are lower. Journal of chemical ecology, 46, pp.631-641. https://doi.org/10.1007/s10886-020-01195-1

Liu J & Wang X (2021). Plant diseases and pest detection based on deep learning: a review. Plant Methods 17(1): 1-18. https://doi.org/10.1186/s13007-021-00722-9

Liang, Jingsai. Confusion Matrix: Machine Learning." POGIL Activity Clearinghouse 3(4) (2022). https://pac.pogil.org/index.php/pac/article/view/304

Mariyappan D, Ganapathy T & Ramalingam R (2013). Survey and incidence of tomato leaf curl virus in Tamil Nadu.Pestology, 37, (7), pp. 10-15

Mazumdar P, Singh P, Kethiravan D, Ramathani I & Ramakrishnan N (2021). Late blight in tomato: insights into the pathogenesis of the aggressive pathogen Phytophthora infestans and future research priorities. Planta, 253(6), pp.1-24. https://doi.org/10.1007/s00425-021-03636-x

Nguyen L D, Lin D, Lin Z & Cao J (2018). Deep CNNs for microscopic image classification by exploiting transfer learning and feature concatenation. In 2018 IEEE International Symposium on Circuits and Systems (ISCAS). pp 1-5. doi: 10.1109/ISCAS.2018.8351550

Osdaghi E, Jones J B, Sharma A, Goss E M, Abrahamian P, Newberry E A, Potnis N, Carvalho R, Choudhary M, Paret M L & Timilsina S (2021). A centenary for bacterial spot of tomato and pepper. Molecular Plant Pathology, 22(12): 1500-1519. doi: 10.1111/mpp.13125

Pan S J & Yang Q (2009). A survey on transfer learning. IEEE Transactions on knowledge and data engineering 22(10): 1345-1359. doi:10.1109/TKDE.2009.191

Potnis N, Timilsina S, Strayer A, Shantharaj D, Barak J D, Paret M L, Vallad G E & Jones J B (2015). Bacterial spot of tomato and pepper: diverse X anthomanias species with a wide variety of virulence factors posing a worldwide challenge. Molecular plant pathology, 16(9): 907-920. https://doi.org/10.1111/mpp.12244

Qasim Khan (2022), Tomato Disease Multiple Sources, CC0: Public Domain https://www.kaggle.com/datasets/cookiefinder/tomato-disease-multiple-sources

Rangarajan, Aravind Krishnaswamy, Raja Purushothaman & Anirudh Ramesh (2018). "Tomato crop disease classification using pre-trained deep learning algorithm." Procedia computer science 133 (2018): 1040-1047. https://doi.org/10.1016/j.procs.2018.07.070

Saito T & Rehmsmeier M (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. PloS one, 10(3), p.e0118432. https://doi.org/10.1371/journal.pone.0118432

Saleem M H, Potgieter J & Arif K M (2020). Plant disease classification: A comparative evaluation of convolutional neural networks and deep learning optimizers. Plants 9(10): 1319. https://doi.org/10.3390/plants9101319

Saleem M H, Khanchi S, Potgieter J & Arif K M (2020). Image-based plant disease identification by deep learning meta-architectures. Plants 9(11): 1451. https://doi.org/10.3390/plants9111451

Shorten C & Khoshgoftaar T M (2019). A survey on image data augmentation for deep learning. Journal of Big Data, 6(1):1-48. https://doi.org/10.1186/s40537-019-0197-0

Sukhija S, Krishnan N C & Singh G (2016). Supervised Heterogeneous Domain Adaptation via Random Forests. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, pp. 2039-2045. http://localhost:8080/xmlui/handle/123456789/2879

Szegedy C, Vanhoucke V, Ioffe S, Shlens J & Wojna Z (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2818-2826.

Wang C, Chen D, Hao L, Liu X, Zeng Y, Chen J & Zhang G (2019). Pulmonary image classification based on inception-v3 transfer learning model. IEEE Access, 7(1): 146533-146541. DOI: 10.1109/ACCESS.2019.2946000

Wang K, Gao X, Zhao Y, Li X, Dou D & Xu C Z (2019). Pay attention to features, transfer learn faster CNNs. In International Conference on Learning Representations, pp. 1-14

Weeraratne W A P G, Wijerathne W M S D K & Dissanayake D M K K (2020). Occurrence of target spot of tomato caused by Corynespora cassiicola in Sri Lanka. *Ceylon Journal of Science* 49(5): 397-400. http://doi.org/10.4038/cjs.v49i5.7807

Wiatowski T & Bölcskei H (2017). A mathematical theory of deep convolutional neural networks for feature extraction. IEEE Transactions on Information Theory, vol. 64, no. 3. pp. 1845-1866. DOI: 10.1109/ACCESS.2020.3024111

Yang Y, Liu T, Shen D, Wang J, Ling X, Hu Z, Chen T, Hu J, Huang J, Yu W & Dou D (2019). Tomato yellow leaf curl virus intergenic siRNAs target a host long noncoding RNA to modulate disease symptoms. PLoS pathogens, 15(1): 1-22. https://doi.org/10.1371/journal.ppat.1007534

Yoshida K, Asano S, Sushida H & Iida Y (2021). Occurrence of tomato leaf mold caused by novel race 2.4. 9 of Cladosporium fulvum in Japan. *Journal of General Plant Pathology* 87(1): 35-38. https://doi.org/10.1007/s10327-020-00963-x

Zagoruyko S & Komodakis N (2016). Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. conference paper at ICLR, arXiv preprint arXiv:1612.03928. https://doi.org/10.48550/arXiv.1612.03928

Zhang H, Zhang L & Jiang Y (2019). Overfitting and underfitting analysis for deep learning based end-to-end communication systems. In 2019 11th International Conference on Wireless Communications and Signal Processing (WCSP): 1-6. DOI: 10.1109/WCSP.2019.8927876

Zhang K, Wu Q, Liu A & Meng X (2018). Can deep learning identify tomato leaf disease? Advances in Multimedia, Special issue pp. 1-10. https://doi.org/10.1155/2018/6710865

Zhou B, Khosla A, Lapedriza A, Oliva A & Torralba A (2016). Learning deep features for discriminative localization. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2921-2929

Zhuang F, Duan K, Xi D, Zhu Y, Zhu H, Xiong H & He Q (2020). A comprehensive survey on transfer learning. Proceedings of the IEEE 109(1): 43-76. DOI: 10.1109/JPROC.2020.3004555