Machine Learning Based Approach in Detection and Classification of Tomato Plant Leaf Diseases

MSc Research Project
Data Analytics

Rajath Ramakrishna
Student ID: x18130721

School of Computing
National College of Ireland

Supervisor: Dr. Muhammad Iqbal

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Rajath Ramakrishna
Student ID:	x18130721
Programme:	Data Analytics
Year:	2019
Module:	MSc Research Project
Supervisor:	Dr. Muhammad Iqbal
Submission Due Date:	12/12/2019
Project Title:	Machine Learning Based Approach in Detection and Classification of Tomato Plant Leaf Diseases
Word Count:	7937
Page Count:	22

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	27th January 2020

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Machine Learning Based Approach in Detection and Classification of Tomato Plant Leaf Diseases

Rajath Ramakrishna
x18130721

Abstract

Tomatoes are one of the most commonly and widely grown crop across the world. As they are one of the staple plant-produces, they are extensively used in every commercial and household kitchen. It is an integral part of the human diet in every continent and culture. Hence tomato cultivation is of great interest to agriculturists. However, tomato plants are not immune to plant diseases. These plant diseases easily and immediately affect the quantity of produce as well as the quality of produce. Therefore, monitoring and analyzing the growth stages of the crop can promote towards the production of disease-free produce with minimal losses to the crop. This research is an attempt at contributing towards the early detection and identification of the onset of diseases in tomato leaves using machine learning. This study intends to analyze the efficiency of algorithms such as XGBoost, Convolutional Neural Network (CNN) and its architectures such as VGG16 and VGG19 in combination with data augmentation and transfer learning over conventional machine learning algorithms such as Support Vector Machines (SVM), Random Forest and test their efficiency in the detection and classification of tomato plant leaf diseases in terms of accuracy, precision, recall and training time. Models training is conducted on publicly accessible PlantVillage dataset consisting of approximately 6500 images of both healthy and diseased tomato plant leaves encompassing five distinct classes. CNN's VGG19 architecture with transfer learning achieved the best result with the overall accuracy of 96 percent compared to other models while having faster training time. Also, it is observed that the performance of the models built improves with increase in the training data. The results achieved on the architectures of CNN while implemented along the transfer learning yields promising results therefore it can be explored more in depth to create an efficient automated leaf disease diagnosis model and thus help farmers and other reserchers in agricultural community in identifying and classifying tomato plant leaf diseases.

1 Introduction

In agriculture, wide variety of diseases affects the cultivation of crops around the world. Plant diseases have always been predominantly analyzed, focusing primarily on biological characteristics of both plant and the diseases. Disease detection is becoming a challenging task that needs to be addressed with special attention as it can cause a significant impact on the ecosystem. Thus, this study focuses on identifying and classifying the diseases that affect the growth of tomato plants. Tomato is one of the world's most cultivated crops and its production has increased substantially over the years. Cultivation of tomatoes can be fruitful for the farmers as it is can be grown in several soil types from black soil to red soil and also in clayey and sandy soil but due to their sensitive nature and changes in the climatic conditions, tomato crops are more vulnerable to various types of diseases during all phases of their development. Mosaic virus, Late blight, Leaf mold and Septoria leaf spot are the most commonly occurred diseases for tomato plants. Diseases constitute to major loss in the cultivation of the crops as well as the most generally affected parts are leaves, stems and the roots of the plant. Detection of tomato plant leaf disease in the early stages is crucial as it benefits in preventing heavy losses and increases the yield. Farmers rely solely on the expert's advice in classifying the diseases accurately, but it takes a lot of effort and is a procedure that takes a lot of time to manually monitor each plant leaf to scan for the disease especially in large farms. Besides there are high possibilities of human error while classifying the diseases, also farmers do not have the experts at their reach. Therefore, the farmer depends mostly on the advice of retail pesticide shops, but the shops sell the higher margin pesticides irrespective of the diseases occurring to loss of the cultivation along with the soil losing its fertility. To address these issues several researches are being performed and smart farming in particular is simple and efficient way that is been adopted recently in the agricultural sector. De Luna et al. (2019) in their study, built a control system to detect and classify tomato diseases automatically using computer vision to monitor the growth of the plants using sensors in motor controlled image capturing box and to identify the diseases developed within the plant but the limitation of such methods are expensive and the motor controlled image capturing box used in the experiment needs expert intervention. However, recent developments in technology have allowed Deep Convolutional Neural Networks and their other applications to evolve and address complex tasks including object recognition and image detection effectively with results illustrating the necessary to study width and depth as the networks generate better results with wider and deeper approach. This study addresses the classification of tomato plant leaf diseases by implementing deep learning models such as CNN and its architectures VGG16 and VGG19 using transfer learning approach along with machine learning algorithms to build a robust model in classifying the tomato leaf diseases to generate high precision levels with minimum system configuration, thus enables to seek answers to questions like "To what extent CNN along with transfer learning technique effective in identifying and classifying diseases in tomato plants?". To the best of my understanding, deep learning models along with eXtreme Gradient Boosting and traditional models like Support Vector Machine and Random Forest are the models still to be addressed jointly and therefore might appear to be of great benefit to the farmers in classifying the tomato plant leaf diseases if favorable results acquired. Objectives of this study includes:

- Applying deep learning algorithms to accurately recognize and classify tomato plant leaf diseases.
- Building multiple models such as CNN and transfer learning models VGG16 and VGG19 based on CNN architecture along with eXtreme Gradient Boost and traditional models like Random Forest and Support Vector Machine to address the solution of classification in identifying the leaf diseases.
- Applying data augmentation technique to generalize the models.

This document is structured as follows: Section 1 is introduction to the study, section 2 covers the related works in the agriculture field in diagnosing the plants, section 3 demonstrates the methodology, section 4 illustrates the design followed in this study, section 5 explains the implementation, section 6 discuss the evaluation and results and section 7 briefs on conclusion and future work.

2 Related Work

It is important to emphasize the previous studies carried out in this area in order to be able to proceed in the right direction. Plant leaf disease identification is an interesting area of interest where both picture recognition and deep learning strategies are widely used to reliably identify the diseased leaves. Plant diseases may damage the crop at any point resulting in major losses. Therefore, plant disease detection is important and machine learning techniques are incorported which analyse the plant by different aspects and classifies into the correct class. This section addresses the most commonly incorporated methods of the previous literatures.

2.1 Classification of plant diseases using transfer learning

Literature review of Mohanty et al. (2016) in their study adopt pre-trained architectures of the Convolutionary Neural Network (CNN) and train on their plant disease dataset only the last few layers of these models to develop a model. The model trained to work with colored data sets achieved best results. Though the model proposed yielded 99.53 percent overall accuracy, this could be observed that the model does not work well against untrained datasets. This has been figured by Barbedo (2018) Arsenovic et al. (2019), in which the findings for the above case fell by 31 percent. Since there are no image augmentation techniques incorporated to enhance the image. This work is further facilitated by author Ferentinos (2018) combined the real world images on the existing images of plant diseases attempting to make the model more prevalent. However, the authors claimed that the reason for achieving such high accuracy by (Mohanty et al. (2016) was that photographs were clicked in a closed and controlled environment and that same model performed poorly when using real-world images. Multiple architectures like AlexNet, ResNet, GoogleNet and VGG have been used and examined for the best fit to solve this problem without incorporating image augmentation techniques. With an accuracy of over 99 percent, VGG architecture produced the best results but the limitation of the model development was that it was very time consuming as each epoch's execution time was over 100 minutes on GPU. The future work of this study includes recommended treatment plans for the diseases that were classified. In another research paper by authors Geetharamani and J. (2019) CNN based transfer learning models using image augmentation techniques are incorporated. Architectures used are VGG16, ResNet, AlexNet and Inception V3. Results show that the max pooling layer in the CNN model is the important layer and tuning it correctly will produce better results in classifying the leaf diseases.

Researchers De Luna et al. (2019) utilizes camera-assisted tomato disease treatment, but using a motor-controlled box to capture images from at least four angles of the plant leaf. The findings showed that the Transfer Learning based CNN model Alexnet performed better than F-RCNN with improved accuracy in the identification of tomato leaf disease since these were collected under laboratory conditions. Using transfer learning techniques author Rangarajan et al. (2018), identified six different types of tomato plant diseases. Then again it was noted that VGG16 (with 97.29 percent) and AlexNet (with

97.49 percent) performed slightly better than VGG16. Just dropout layers were used by previously qualified models to prevent overfitting. The study also attempted to determine the effect on accuracy of different hyperparameters and observed that the learning rate had a significantly greater impact on accuracy while running the epochs. Critique of Fuentes et al. (2017) researched complex meta architectures and deep extraction characteristics to classify real-time plant diseases. In combination with the already trained dataset, VGG16, ResNet50, ResNet101, ResNet152, and ResNeXt504, Faster-R-CNN, R-FCN and Single-Shot Multiplex Detector (SSD) were used. The images were first manually annotated by a bounding box during pre-processing, and after that the techniques of data augmentation were being used to prevent overfitting issues and noticed that VGG16 performed the best. In the same way authors Too et al. (2019) incorporates CNN architectures like ResNet, VGG16, DenseNet and InceptionV4 in the study of identifying and classifying plant diseases using PlantVillage dataset and describes that DenseNet delivered the best result with the accuracy of over 99percent. In this work, standardization was carried out using the technique of batch normalization. The next layer accessed each output layer as an input, thereby enabling higher learning rates and reduces the overfitting. This was merely an attempt to achieve the objective of creating an efficient model with minimum computational requirements. Critique review of authors Ramcharan et al. (2017) incorporated pre-trained CNN models by training only the last few layers with their dataset using SVM as the layer of classification. The CNN architecture InceptionV3 in combination with SVM provided an accuracy of around 92 percent. It was also noted that a measurable impact on the model's performance was found with the change in the dataset's volume by using transfer learning. In this study, image augmentation techniques are used to transform the pre-existing images without needing to use a larger database, thus reducing overall processing time and additional computational requirements that enable us to accomplish the research objective.

2.2 Classification of plant diseases using image processing and machine learning techniques

Agricultural development is the factor on which the economy of nation depends enormously. This is the inspiration to leaves unhealthiness identification that saves plant depletion and production. As a result, image processing methods are applied on unhealthy plant leaf discovery and recognition. This study by authors Dhaware and Wanjale (2017) addresses image pre-processing and image segmentation methods along with the Support Vector Machine (SVM) algorithm is used for automated recognition and analysis on different plant leaf disease classification that could be used to identify leaves disease. Likewise, authors Ranjith et al. (2017) rightly point out that issues caused by lack of supply of water must also be taken into account by farmers. An automatic irrigation system run using the mobile application on the cloud server has built-in embedded function that can be remotely used to evaluate soil moisture content, soil temperature, etc. Image processing based early diagnosis of plant diseases can be achieved using this method. Critique of Ganesan et al. (2018) illustrates image processing and Fuzzy-based segmentation methods that could be used through its images to directly solve the problem of recognising and recovering defective parts of the plant. This approach is assessed using variables of image quality in image processing along with its segmentation to show the potency of the plant leaf disease recognition. Only shortcoming of using this CIELuv is that this is device-dependent, which means that other color spaces like CIELAB or HIS can substitute it for greater reliability.Authors Singh and Misra (2017) also conducted image processing and segmentation methods to study the plant's health. The segmentation method is based on multiple features noted in the picture, indicating that data may be in the form of color discrepancy, surface deformities, or even on their structure / borders. Clustering algorithm is used to accomplish color-oriented segmentation, optimal results are obtained with little computational effort, which indicates the usefulness of the proposed algorithm in the detection and classification of leaf diseases. Francis et al. (2016) also suggested an image processing algorithm for the detection and identification of plant leaf disease using pepper plant leaves as the study specimen. This algorithm has attained constant monitoring of the external indications and their subsequent changes. As a result, the diseases have been successfully recorded and also the healthy leaves have been separated from the infected leaves, thus increasing the pepper crop production.

The research by Abdu et al. (2019) focused on plant disease identification using machine learning technologies which take photos clicked on by long-range cameras like drones or even satellite images. Segmentation of images is achieved by removing the surroundings of the leaf image and by concentrating on the area of interest on the leaf which makes easy to implement to achieve better results. This study uses SVM algorithm as classifier in identifying the plant leaf diseases. Literature review of Pooja et al. (2018) suggests methods of machine learning and image processing mechanisms by initially defining and capturing the infected region and then conducting the image processing. The segments are thereafter obtained, and the area of interest is identified, then the features are extracted on the very same. Finally, the data obtained are sent to SVM for classification and the results indicate that the methodology presented in this paper provides significantly improved results than the previously used techniques of disease detection. In another work, researchers Pantazi et al. (2019) use various images of leaf specimens to demonstrate an automatic means of identifying crop diseases. Local binary patterns (LBPs) as well as one-class classification obtained the retrieval of the leaf features. Each new picture introduced to the database improved the algorithm's accuracy and potential. Significant accomplishment of this method was the ability to spot and recognise not just the currently identified disease, but also new diseases and interpret them as a new section of further reference.

2.3 Classification of plant diseases using CNN and its architectures

Francis and Deisy (2019) in their study uses CNN model is built using photographs of apple and tomato leaves and binary classification of diseased leaves against healthy leaves is performed by incorporating four convolutional layers along with pooling and dense layers and dropout function to avoid overfitting problem using Tesla GPU which uses parallel processing with increasing the model processing time and classifies the plant leaf diseases effectively. Literarure review of Durmus et al. (2017) illustrates the usage of robots in greenhouses and farmlands to capture the images of the leaves. Deep learning-based CNN architectures likes SqueezeNet and AlexNet models are implemented and the results indicate that Squeezenet model performs significantly better than the other architecture solving the issue of classifying the leaf diseases when the real-world images are included. Also, Authors Hidayatuloh et al. (2019) in their research uses SqueezeNet architecture as it performs well with the default parameters making the model small and deployable with producing high accuracy making this model to be deployed on cloud and access it through remote devices because of its smaller bandwidth. Literature review of Türkoğlu and Hanbay (2019) analyzed outcomes employing different methods of deep neural network architecture for plant disease detection. Extraction of deep features and knowledge transfer were among the techniques used to fit the models. Images of pest and plant leaves has been used to assess the result, suggesting that SVM / ELM coupled with deep feature extraction yielded better results than VGG16 and VGG19. Fully Connected layers yielded better results compared to other layers.

Critique of Sardogan et al. (2018) shows the implementation of Learning Vector Quantization algorithm along with the CNN in classifying the tomato leaf diseases with using the RGB values of the pixels in the images to convolution every single image. The LVQ algorithm used full 500 feature vectors to train and test methods acquired from the initial images. Additionally, max pooling and ReLU activation functions are used to improve the model performance which in return classify the images accurately. Another study by Zhang et al. (2019) also used CNN to classify vegetable leaf diseases, by using three-channel convolutional neural network (TCCNN) for every single RGB color on the infected leaf. Finally, Softmax classifier layer classifies the diseases. Literature review of Tm et al. (2018) illustrates the adoption of LeNet architecture of the CNN model making use of the deep feature extraction in classifying the diseases into multiple classes. In this article, the future work suggested was to show the same for various learning rates and enhancement techniques.

2.4 Classification of images using XGBoost

Literature review of Gao et al. (2017) develops a new XGBoost based identification methodology based on the subsampling on the weighted columns for PASCAL VOC 2012 dataset. In this, pretrained CNN model is used and few layers are fine tuned with the new dataset to achieve the accuracy of over 92 percent. Another study by authors Mudgal et al. (2018) illustrates the detection of tumors in the brain through the MRI images using XGBoost classifier in which the accuracy achieved is 100 percent due to the smaller data size of 50 images but for pre processing K-Means clustering is implemented to reduce the noise of the image. Critique of Georganos et al. (2018) develops XGBClassifier to recognize the trend in population growth in the urban areas along with other machine learning algorithms like RF and SVM and witnessed that XGBoost outperformed against the other two alogorithms. As part of future work authors want to combine XGBoost with deep learning models to assess the performance. Summary of the similiar literature reviews are shown in Figure 1

Title	Author	Techniques/Algorithms used	Gaps
Automatic Disease Symptoms Segmentation Optimized for Dissimilarity Feature Extraction in Digital Photographs of Plant	Abdu, A. M., Mokji, M. M., Sheikh, U. U. and Khalil, K. (2019)	Support Vector Machine(SVM)	1. Training and testing data not clearly defined 2. Classifiers cannot classify all the diseases correctly
Diseases Detection of Various Plant Leaf Using Image Processing Techniques: A Review	Santhosh Kumar, S. and Raghavendra, B. K. (2019)	Image processing techniques	1. Too many features for single image 2. No performance comparison
Tomato leaves diseases detection approach based on Support Vector Machines	Mokhtar, U. Ali, M. A. S., Hassenian, A. E. and Hefny, H. (2016)	Support Vector Machine (SVM)	1. Dataset description is missing 2. Feature extraction technique is missing
Plant Leaf Disease Detection and Classification Based on CNN with LVQ Algorithm	Sardogan, M., Tuncer, A. and Ozen, Y. (2018)	Learning Vector Quantization (LVQ)	1. Cannot classify different classes of diseases 2. Dataset size is too small

Figure 1: Summary of Literature reviews

As shown in the related works, there have been found several problems and challenges in the past. The lack of real-world photographs of plants with correct labelling of the disease names were a concern in the past due to which the data doesnt generalize well. This study attempts to build an optimum model with low computational resources in classification of tomato plant leaf diseases.

3 Methodology

As shown in Figure 2, the challenge of data processing grows menacingly beyond as we progress into the age of digital technology because our skill to interpret and comprehend large data sets is a long way behind our ability to collect and store the data. To benefit the extraction of useful information from the rapidly increasing volumes of data, a new conception of computational tools and techniques is needed. These methods and techniques are the topic of the research field of discovery of information in databases also known as knowledge discovery in databases (KDD) stated in the study Fayyad et al. (1996). Knowledge Discovery Database (KDD) is the approach used in classifying the leaf diseases of the tomato plant. KDD is the process through which knowledge is extracted from data. It is widely used to understand, identify the trend and derive intelligent information from the data in machine learning and artificial intelligence. It pursues certain procedures that allows the research to systematically meet the goal. The phases of KDD process are:

- Selection of the data
- Pre-processing of the data
- Transformation of the data
- Integration of suitable data mining techniques
- Interpretation or evaluation of the results

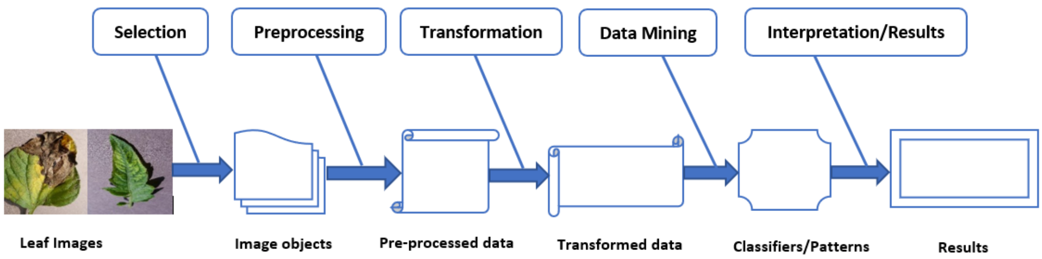


Figure 2: KDD methodology in reference with tomato leaf disease image classification

3.1 Selection of the data

In the interest of creating effective image classification models for tomato plant leaf diseases requires a verified and large number of images, but that is complicated and expensive to acquire datasets that contain images of plant leaves. Until recently such a dataset was difficult to find and the smaller dataset which were found was not accessible free of charge, thus a project named PlantVillage started gathering thousands of images of both infected and healthy plants and made them freely accessible to everyone. Authors Mohanty et al. (2016) in their study used PlantVillage dataset and reported convolutional neural network approach to classify plant diseases by transforming the images into different categories namely grayscale, color, segmented and the authors made this dataset publicly accessible on GitHub [1]. This research uses only the colored images of tomato plant leaves from the publicly accessible PlantVillage dataset from GitHub which contains 8700 images of both healthy and diseased tomato plant as shown in Figure 3. The dataset is categorized into 10 classes of which one class contains images of healthy tomato leaves and other classes contains different diseased tomato leaves namely Tomato mosaic virus, Late blight, leaf mold and Septoria leaf spot among others.

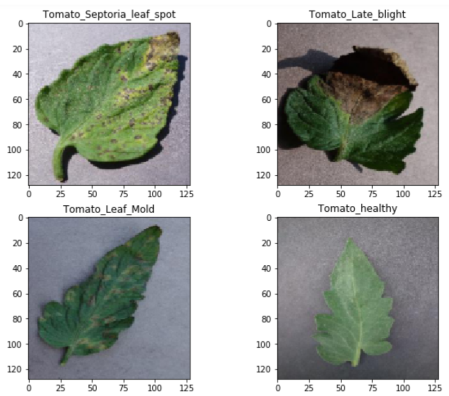


Figure 3: Tomato leaf samples of size 128*128 pixels

3.2 Pre-processing of the data

Dataset in the Github database is stored with 14 different species of plants of which only colored Tomato plant leaves are loaded into the model. This study includes images of both healthy and infected leaves with 9 different types of diseases. Hence image folders of other plants are removed from the downloaded directory. Further, the accompanying classes Bacterial spot, Early blight, Target Spot, Two spotted spider mites and Tomato Yellow leaf Curl Virus are removed as this study concentrates on diseases that are very harmful and most common in tomato plants. This results in the sample size of 6596 images of leaves across five folders. These images are then loaded by parsing the folders iteratively and reading the image files located. Further, the images are resized to meet the dimensionality constraints of the models being implemented and to maintain the uniformity of the images. Label Encoder python library is then used to label the classes of the diseases and histogram is plotted as shown in Figure 4 to see the total number of images in each class. Models are then developed on 80:20 ratio train and test split as it yielded the best results in the study by authors (Ferentinos (2018)).

3.3 Transformation of the data

Initially, the images loaded are transformed to NumPy arrays to normalize the RGB values so that they can be assessed by the classification models. One of the other problems

[1] `https://github.com/spMohanty/PlantVillage-Dataset`

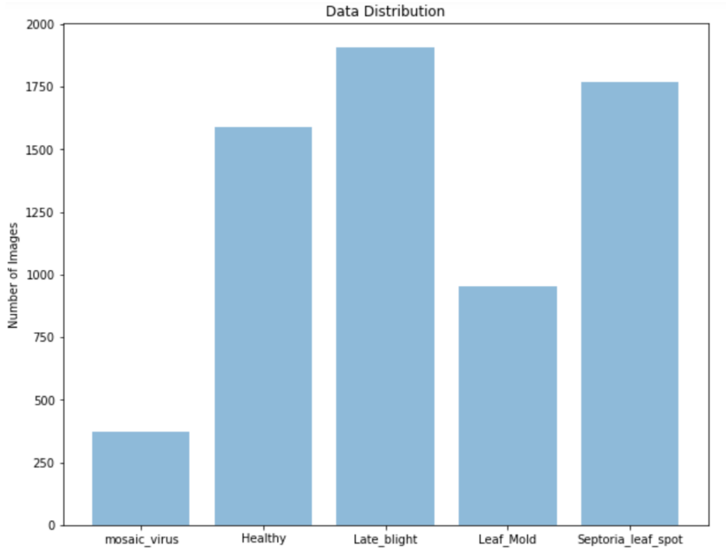


Figure 4: Barchart to visualise total number of images in each class

mostly encountered with image data is the irregularity in images. Few are just too large or too small, and others are rectangular rather than square and so forth. The amount of data in the training set, which often result in overfitting, is another commonly encountered issue. To mitigate such issues data augmentation technique is incorporated in which the images in training set data is transformed to improve the model's ability to classify the image[2]. Along with the augmentation this study uses batch normalization and dropout layers on CNN model to improve the validation accuracy on the test set. This study uses below parameters in image augmentation technique from ImageDataGenerator class of Keras:

- rotation_range: Rotates the actual image as per the defined angle.
- width_shift_range: It defines the width by which the object is to be moved horizontally with the values between 0.0 and 1.0.
- height_shift_range: It defines the height by which the object is to be moved vertically with the values between 0.0 and 1.0.
- shear_range: Slants the image structure by fixing one axis and stretching the image angle as per the defined shear angle.
- zoom_range: Magnifies the image while the value is greater than 1.0 and zooms out the image while the value is less than 1.0.
- horizontal_flip: Image is flipped horizontally.
- fill_mode: Points with the null pixel values in the image will be filled.

In this study, the above-mentioned image augmentation technique is applied on the transfer learning and convolutional neural network approach to generalize the neural

[2] `https://towardsdatascience.com/augmentation-for-image-classification-24ffcbc38833`

networks performance on training set over the traditional machine learning algorithms. This method performs better on unseen data as the test set is not augmented and only transformation performed on the test set is batch normalization for convolutional neural network approach.

3.4 Integration of suitable data mining techniques

Multiple models are developed and assessed to seek an ideal model that performs better in contrast with others both in terms of performance and validation accuracy. Built models are based on two techniques of which first approach is developing the machine learning models Random Forest, Support Vector Machine and eXtreme Gradient Boost that are trained on the tomato plant leaves data to classify the diseases and the second approach is the development of basic CNN and CNN derived deep learning models VGG16 and VGG19 which uses techniques of transfer learning that adopt already learnt features and pre-trained weights of previously trained model from ImageNet dataset. Final few layers have been retrained with the tomato plant leaf data. In addition, the parameters of all the models applied are modified and the experiments are conducted multiple times to find the optimal model in classifying the tomato leaf diseases. Related works in this field indicate that transfer learning technique with training the last few layers of the model with the data and eXtreme Gradient Boost algorithm are the area that needs to be addressed. Hence, these models are applied in order to verify whether they could offer greater insight than the existing models in classifying the diseases of tomato leaves.

3.5 Interpretation or evaluation of the results

Accuracy, precision, recall, and f1 score are the predominant metrics considered in this study to measure performance of the models. Tests are carried out on three CNN models with multiple layers and transfer learning approach along with three other machine learning algorithms SVM, Random Forest and XGBoost to evaluate the effectiveness of the models with the below-mentioned metrics [3].

- Accuracy: The percentage of accuracy is used in the evaluation of results in classification. It estimates the accurate model prediction on the total sample size given as the input to the model and is represented mathematically as in equation (1)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

  where TP=TruePostives,TN=TrueNegatives,FP=FalsePositives,FN=FalseNegatives

- Precision: This is the total number of accurate positive results divided by the total number of positive results predicted by the classifier and is represented mathematically as in equation (2)

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives} \tag{2}$$

- Recall: This is the total number of accurate positive results divided by the total number of all the results predicted by the classifier and is represented mathematically as in equation (3)

[3] https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives} \tag{3}$$

- F1 Score: This is the mean of precision and recall with value ranging from 0 to 1 and it finds right balance between the other two metrics with large number of true negatives and is represented mathematically as in equation (4)

$$F1 - Score = 2 * \frac{1}{\frac{1}{precision} + \frac{1}{recall}} \tag{4}$$

4 Design Specification

In this section, design of the models built and the architecture of all the models are explained in detail. As shown in Figure 5, 2-tier layout is selected since the tomato leaf images are publicly available and are not developed specifically. Python is used as the programming language due to its accessibility to large number of libraries such as Keras which can be readily used to build the models. In the business logic tier, Jupyter IDE is used to preprocess and transform the data and to implement the models. The modles built are then evaluated and the results are analysed in the presentation layer using 'matplotlib' library. Models implemented are as follows

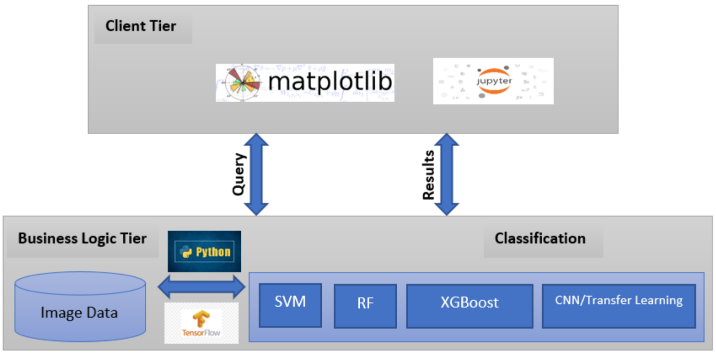


Figure 5: Research Design

4.1 Convolutional Neural Network

First model implemented is basic Convolutional Neural Network also called as ConvNet is an algorithm used in deep learning techniques in which input image is assigned with values like learning weights and biases to the different area in the images and be able to recognize the image. CNN model includes multiple layers like convolutional layer, pooling layer, fully connected layer and activation functions as shown in Figure 6 that are implemented in this study in classifying the tomato leaf diseases. The layers and functions used in the design are discussed by Francis and Deisy (2019) as below

- Convolution layer: This layer receives the colored input image of the tomato plant leaf in the form of RGB and for further processing it collects the output from another layer as input then the received input is interpreted as pixel values to generate feature map that depicts the low level characteristics of the image such as curves and edges and the variable parameter filters are applied to obtain feature map which results in effectiveness of the model.

- Activation function: Non-linearity increases the power of a neural network. After the first convolution layer, a non-linear activation function is immediately applied. In this study rectified linear unit function commonly known as ReLU is applied that makes all the negative values to 0 to increase the non-linearity of the model.

- Pooling layer: Spatial size of the feature is reduced using this layer which in return reduces the computational power and dimensionality to process the data and thereby preserves the model's effective training cycle. This study uses the max pooling layer to retain the maximum value from the covered region of the tomato plant images based on the size of the kernel.

- Fully Connected Layer: This layer recognizes characteristics of very high levels that are highly correlated with an object or class. Fully connected layer output is a single dimensional vector that is achieved by flattening the final pooling layer output.

- Dropout function: Dropout is a feature that enhances generalization by studying multiple pattern representations and overfitting is controlled by applying this layer in which random collection of activations is removed.

- Softmax function: This is the classification layer applied after the dense or fully connected layer used in this study for multi class classification.

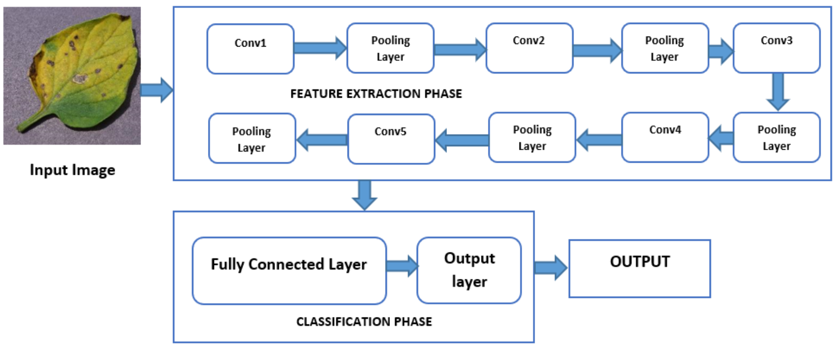


Figure 6: 5 Layered CNN

4.2 Transfer Learning

Transfer learning in deep learning is a technique in which a model of a neural network is first trained on a topic similar to the problem being addressed. In a new model trained one or more layers of the learnt model are then used. Advantage of applying transfer learning is it decreases the model's training time with less computational power as the network is pre-trained. This is achieved by computing few additional dense layers on the already trained network to identify images in the new dataset[4]. In this study VGG16 and VGG19 transfer learning models are applied on the tomato plant leaf image dataset by

[4]https://towardsdatascience.com/a-beginners-guide-to-xgboost-87f5d4c30ed7

using already learnt features and pre-trained weights of previously trained model known as ImageNet that includes millions of images corresponding to over 1000 classes to classify the tomato leaf diseases.

- VGG16 and VGG19: Simonyan and Zisserman (2015) introduced this deep learning architectures that uses only 3x3 convolutional layers that are piled on top of one another with the number of filters depending on the depth of the layer. Further, max pooling is used to reduce the volume size that consists of 5 layers and softmax classifier is included on the two fully connected layers with 4096 neurons. The numbers 16 and 19 represents the number of weighted layers present in the network. Thus, in this study VGG architectures are applied for image recognition in plant leaf disease detection.

4.3 Support Vector Machines

Supervised learning algorithm Support Vector Machine popularly known as SVM is a machine learning algorithm that is used for both classification and regression problems. In this study of classification of tomato plant leaf diseases, SVM is applied in which each tomato leaf image is plotted as a point in n-dimensional space where n is the properties of the image amidst each value of the attribute being the value of a specific coordinate. These attributes are then segregated by hyperplane to classify the images accurately.

4.4 Random Forest

Random Forest is an ensemble model consisting of multiple decision trees. This is a very reliable algorithm which performs random sampling of training data to avoid the overfitting and determines on the negatively correlated trees to surpass single models.

4.5 eXtreme Gradient Boost

XGBoost is an opensource library that provides elevated-performance in implementing gradient-boosted decision trees. This bundle is extremely powerful and quick to deploy. Unlike decision trees where single model is trained on the dataset and the results are predicted, boosting approach is more effective and iterative technique as it combines multiple models and foresee the output of the final model. Being an ensemble technique this approach works successively with each other and the new model is trained to overcome the errors of the earlier model in such a way until no improvements can be done further, hence in this study extreme gradient boosting approach is applied on the images of tomato plant leaf in which the newer models are trained to correct and reduce the residuals of the previous models in classifying the diseases of the plant accurately.

5 Implementation

In this section, implementation of the models built are explained in effort of developing efficient model is detecting and classifying the tomato plant leaf diseases based on the images of both healthy and diseased leaves across five classes of which four common diseases includes Late blight, Leaf Mold, Mosaic virus and Septoria Leaf Spot.

5.1 Environment

In this study, Python version 3.7.3 is used in implementation along with Jupyter notebook as IDE (Integrated Development Environment). TensorFlow is the popular library used in image recognition that helps in building an efficient model. This library is specifically created for Python and hence Python programming language is used in this project. Along with TensorFlow this study uses Keras which is an Application Programming Interface (API) built to use TensorFlow functions. This makes implementing TensorFlow's complex functions easy to implement without additional modifications or plugin as the Keras API is configured to run with Python thus making the implementation straightforward.

5.2 Data Handling

In the interest of creating effective image classification models for tomato plant leaf diseases requires a verified and large number of images such dataset is downloaded from public repository and is preprocessed as mentioned in the section 3.2. To mitigate the issues of irregularity of images which results in overfitting, data augmentation technique from ImageDataGenerator class of Keras is incorporated in which the images in training set data is transformed to improve the model's ability to classify the image. The parameters used in data augmentation technique is explained in the 4.5.

5.3 Architectures

As mentioned in the subsection 4.1, the architectures of all the models built are discussed. First model implemented is basic Convolutional Neural Network also called as ConvNet, is an algorithm used in deep learning techniques in which input image is assigned with values like learning weights and biases to the different area in the images and be able to recognize the image. All the necessary packages like Conv2D, MaxPooling2D, Activation, Flatten, Dropout, Dense etc are imported from TensorFlow library using Keras. Additional packages like numpy, pickle etc. are installed using pip install command. Basic CNN model is developed with multiple layers and default configuration to see how many layered models performs better. Of all the layers ranging from 3 to 5, it is found that 5 layered CNN model produced optimal results in classifying plant leaf diseases. Further, the model is tuned with the hyper parameters. Parameters used in the models are:

- batch_size: 32, 64
- learning_rate: 0.01, 0.001
- epochs: 10, 15, 25
- optimizers: Adam, Adamax, SGD

Dropout layers are added along with the other layers to avoid the overfitting problem of the data. Further, Softmax function is applied as a classifier layer after the fully connected or dense layer to classify the diseases.

Transfer Learning: As mentioned in the subsection 4.2, transfer learning is a technique in which a model of a neural network is first trained on a topic similar to the problem being addressed. In this approach CNN model used is pre trained on ImageNet dataset consisting of millions of images corresponding to over 1000 classes. Advantage of such techniques is it uses minimal computational resources and reduces training time. Therefore, in this study two transfer learning models based on CNN are implemented namely VGG16 and VGG19. Models are run on 128*128 pixel image size due to the limited computational resource as higher pixel images cause memory error. This is then achieved by computing few additional dense layers and freezing the other layers on the already trained network. Along with the global average pooling layer 'softmax' classifier with five classes are used for the classification to identify the tomato plant leaf diseases.

Random Forest: As mentioned in the subsection 4.4, Random Forest is an ensemble method used in this study of classification. To incorporate this model 'RandomForestClassifier' from 'sklearn.ensemble' library is imported. There are several parameters obtainable from the library for this classifier but after performing series of tests it is found that the default parameters performs better with the model for classifying the tomato leaf diseases such that only the 'n_estimators' parameter is set to 100 to perform better. This parameter represents the number of trees included in the random forest model. Also, 'bootstrap' parameter is set to 'true' by default meaning that the sample observations are made with replacement. The model is further developed with the train and test split of 80:20 ratio and are run to analyze the effectiveness of this algorithm.

Support Vector Machine: As mentioned in the section 4.3, this is a supervised learning algorithm incorporated in this study of classifying the leaf diseases. This algorithm is incorporated by importing 'svm' from 'sklearn' library. Like random forest, SVM also includes several parameters as seen in the output in Figure 7 Initially, model built with the train and test split of 80:20 ratio and included 'rbf' kernel with no parameters, but the results were considerably lower. Thus, to boost the model's accuracy, SVM tuning is a performed by changing the kernel to 'linear' without any additional parameters. This resulted in improving the effectiveness by providing the optimal results in classifying the diseases.

```
In [14]:  support_vector = svm.SVC(kernel='linear', probability=True)
          support_vector.fit(x_train, y_train)

Out[14]: SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
             decision_function_shape='ovr', degree=3, gamma='auto_deprecated',
             kernel='linear', max_iter=-1, probability=False, random_state=None,
             shrinking=True, tol=0.001, verbose=False)
```

Figure 7: Implemented SVM code with default parameters

eXtreme Gradient Boost: As mentioned in the subsection 4.5, eXtreme Gradient Boost also known as XGBoost is an ensemble technique that provides elevated-performance in implementing gradient-boosted decision trees. Within the machine learning algorithms, this is one of the popular algorithms used to boost the model's accuracy. This is incorporated in this study by installing XGBoost package but using pip install xgboost command. Further, 'XGBClassifier' from 'xgboost' library is imported and the train and test split of 80:20 ratio is applied on the model. Like random forest model it is seen that default parameters perform better for this algorithm hence there is no additional parameters added. Default parameters included are seen in the output in Figure 8 Parameter 'multi:softprob' is the default objective parameter in the model that uses fuzzy clustering logic to evaluate the probability for all the classes and classifies the plant leaf diseases. Another default parameter is the booster parameter with the value 'gbtree' is used along with the objective function since the dataset contains multiple classes. Thus, the model is built with the default parameters to evaluate the effectiveness against the other models.

```
In [15]:  classifier = XGBClassifier(probability=True)
          classifier.fit(x_train, y_train)

Out[15]: XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
                       colsample_bynode=1, colsample_bytree=1, gamma=0,
                       learning_rate=0.1, max_delta_step=0, max_depth=3,
                       min_child_weight=1, missing=None, n_estimators=100, n_jobs=1,
                       nthread=None, objective='multi:softprob', probability=True,
                       random_state=0, reg_alpha=0, reg_lambda=1, scale_pos_weight=1,
                       seed=None, silent=None, subsample=1, verbosity=1)
```

Figure 8: Implemented XGBClassifier with default parameters

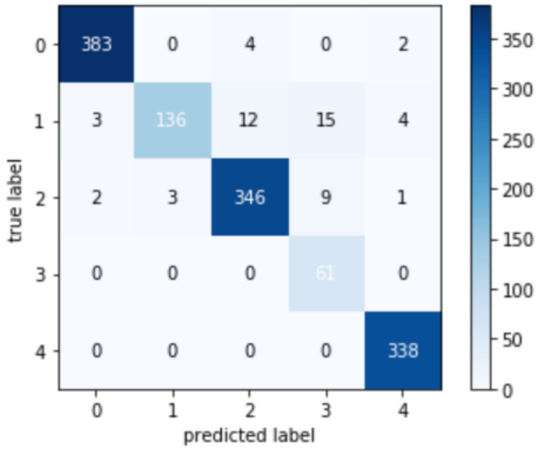
6 Evaluation

In this section all the results obtained in achieving the project objectives are discussed in detail. Models and architectures built in this study try to mitigate the problems faced in the agricultural community in identifying and classifying tomato leaf diseases. Hence, transfer learning techniques based on architectures of CNN are implemented to illustrate the effectiveness and sturdiness of the models built against other machine learning algorithms like Support Vector Machines, Random Forest and XGBoost of which the eXtreme Gradient boosting algorithm is the most recently developed decision tree based ensemble algorithm. This study incorporates accuracy, recall, precision, F1 score and ROC as the evaluation metrics to analyse the results. Then, confusion matrix is generated to determine the correct number of predictions the models made on classifying the tomato leaf diseases based on the negative and positive classifications of each class. This matrix along with the classification report evaluate the performance of the models built. The results summary of all the models implemented in this study is listed in Table 1

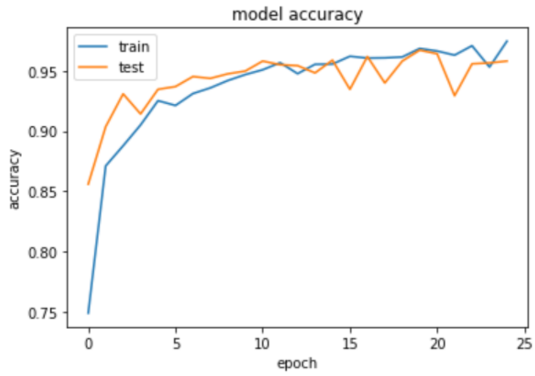
Table 1: Results Summary

Model	**Accuracy**	**Precision**	**Recall**	**F1-Score**
VGG19	96%	92%	95%	93%
VGG16	91%	91%	92%	91%
Random Forest	85%	84%	76%	78%
XGBoost	84%	83%	78%	80%
SVM	82%	82%	80%	81%
5-Layer CNN	81%	86%	74%	77%

From the values in Table 1 we can infer that VGG19 model of CNN architecture using transfer learning performs best in classifying the tomato leaf diseases since it has higher accuracy of 96 percent along with higher precision, recall and F1-scores compared to other models with data split of 80:20 ratio. It also indicates that transfer learning model VGG16 is next best performing model with accuracy of 91 percent after VGG19 meaning that transfer learning techniques works best in classification of images compared to other machine learning algorithms.

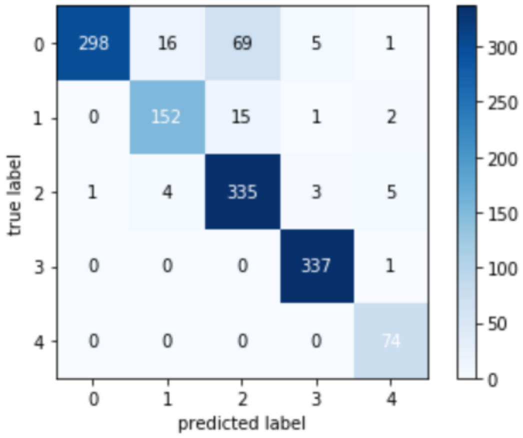


Confusion Matrix

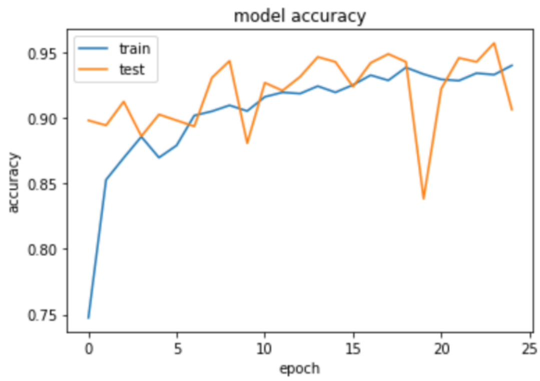


Model Accuracy curve

Figure 9: Confusion Matrix and Model Accuracy curve VGG19



Confusion Matrix

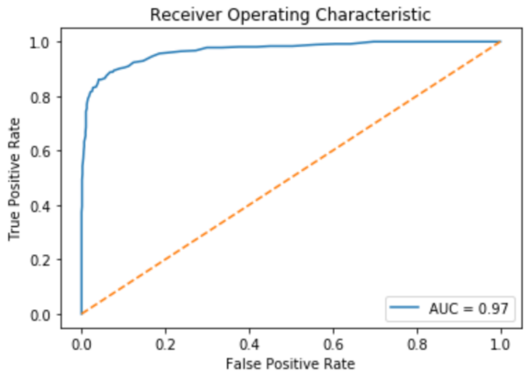


Model Accuracy curve

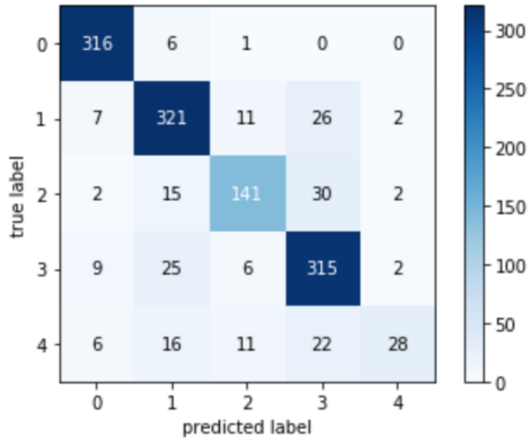
Figure 10: Confusion Matrix and Model Accuracy curve VGG16

6.1 VGG16 and VGG19

Transfer learning approach implemented in this study works best in classifying the images of tomato plant leaves in which the images that were preprocessed was converted into numpy arrays and the pretrained model was retrained with project specific data which included augmented images on the training set. This was then used for classification of unseen data. Epochs, batch size and learning rates of different values were tried and a robust model that predicts best was developed with VGG19 model resulting in accuracy of 96 percent over the VGG16's accuracy of 91 percent. Also VGG19 computational time is relatively less compared to computational time of VGG16. Confusion matrix and model accuracy curve for VGG19 and VGG16 models are shown in Figure 9 and Figure 10 respectively. From the figures it is clearly seen that the pretrained transfer learning models perform almost similarly with 25 epochs. Although, it can be seen from model accuracy curve of VGG16 model that it is unstable even after 20epochs meaning that it should be fine tuned with the hyper parameters. Thus, the models perform better in achieving good results with less computational resources.



ROC curve



Confusion Matrix

Figure 11: ROC curve and Confusion Matrix Random Forest

6.2 5- Layered Convolutional Neural Network

With 5 convolution layers and filter sizes of 32,32,64,64 and 128 this model is trained from scratch with 25 epochs. Series of tests are performed by tuning the hyper parameters to find the optimum learning rate and batch size. The model achieved 81 percent of accuracy for 25 epochs with learning rate set to 0.01 and batch size set to 32. Also, the F1-score achieved is 77 percent. The training time for the model is 1 hour 4 min for 25 epochs and dropout value of 0.25 is given to avoid the over fitting issue. It also shows the confusion matrix obtained for the built model displaying the actual number of classes versus the predicted classes.

6.3 SVM, Random Forest and XGBoost

Although, SVM's are quite popular in classification, they did not perform well in classifying tomato leaf diseases as indicated by their accuracy score of 82 percent. Though, they classified two diseases well, their performance overall was meagre and hence is not recommended for further analysis. Other models like random forest and XGboost were also implemented. However, fine tuning of these models using grid search was not possible due to hardware limitations.Amongst these models, Random Forest performed the best with an overall accuracy of 85 percent and also had lesser training time when compared to other models. The confusion matrix and roc curve for this model is as shown in Figure 11

6.4 Discussion

The research's primary goal was to develop effective models to recognise and classify tomato crop leaf diseases. In order to understand the current limitations and research discrepancies, a comprehensive literature review was carried out indicating the need of models that could generalize the plant diseases effectively on the images data with less computational resources. Hence transfer learning methods like VGG16 and VGG19 were incorported in this study to overcome this problem. Along with the transfer learning approach, 5-layered CNN, boosting algorithm XGBoost and traditional algorithms SVM and RF were also developed to compare the best performing model. The models were then evaluated based on the evaluation metrics like accuracy, precison, recall and F1-

score. Pre processing phase includes resizing the images and converting them to numpy arrays along with the data augmentation on training set to develop a robust model.

Findings showed that the transfer learning models were effective in predicting practically any type of disease it was trained on. On the other hand, standard algorithms along with 5 layered CNN model also performed well but the VGG19 performed the best with 96 percent accuracy. Reason for other algorithms not to perform as better as transfer learning is that the structure of the algorithms developed expect for huge samples of training data. Therefore these algorithms results in good but not best results.

Though the study accomplished its objectives, in the midst of its successfully built models, it encountered few obstacles during the development including the lack of real world data and computational resources. Model might under perform when the real world images are loaded as the images included are the images clicked in controlled environment. Image sizes were reduced mainly because of the training time and memory errors caused due to low computational resources. Apart from the listed shortcomings the models developed are efficient and transfer learning approach reduced the training time and this can be much more optimised by using Graphical Processing Units. By overcoming these shortcomings in the future might result in building much more robust model in classifying the diseases.

6.5 Computational time

Table 2: Computational time

Model	**Training time**	**No. of images**
VGG19	1 hr 4 min	5296
VGG16	3 hr 24 min	5296
Random Forest	1 min 35 sec	5296
XGBoost	1 hr 46 min	5296
SVM	1 hr 19 min	5296
5-Layer CNN	1 hr 4 min	5296

Since only the pretrained transfer learning models behaved the best, the training time needed for each model was calculated to show the effectiveness of the proposed method. As seen in the table 2, Random Forest takes very less computational time of 1 minute but VGG19 model classifies better with 96percent. Thus, Random Forest can be fine tuned with other parameters and GridSearch can be applied to check for the performance to explore further.

7 Conclusion and Future Work

As mentioned in the discussion section, the conclusion drawn encourages in using CNN based transfer learning model VGG19 for classification of tomato plant leaf diseases, thereby answering the research question and fulfilling the objectives of this research. Image augmentation technique incorporated in this study makes the VGG19 architecture more efficient than it actually is, that helps in performing better in classifying new data. Building the models with Graphical Processing Units might reduce the training time.

Random Forest, XGBoost, SVM and 5 layered basic CNN algorithms can also be chosen by fine tuning the models and adding hyper parameters as the performance of these models were also powerful. The work carried out can be further extended by evaluating other viable transfer learning models. As part of future work, object detection technique can be implemented on the built model which segments the image based on the area of interest and can detect the disease spots earlier that the current model. Also the model built trains on the images of single leaf and as a scope of improvement the images of multiple leaves can be added. The models can be further deployed on the remote devices using cloud servers which could help farmer to identify the disease as early as possible by clicking a picture on mobile device enabling to monitor the crop and treat with suitable remedies in a timely manner and helping the agricultural community as a whole.

8 Acknowledgment

My most sincere thanks to our college, NCI and department of MSc in Data Analytics for providing an opportunity for me to carry out this research work successfully for the past three months. I would like to thank my supervisor and mentor Dr. Muhammad Iqbal for his guidance during the project. His unvarying support helped me complete this thesis and present it. I thank him greatly for his time and especially for his useful comments. I find an ideal mentor in him. Lastly, I would like to immensely thank my friends and family back home for their support and for boosting my confidence during this research period.

References

Abdu, A. M., Mohd Mokji, M., Sheikh, U. U. and Khalil, K. (2019). Automatic Disease Symptoms Segmentation Optimized for Dissimilarity Feature Extraction in Digital Photographs of Plant Leaves, *Proceedings - 2019 IEEE 15th International Colloquium on Signal Processing and its Applications, CSPA 2019*.

Arsenovic, M., Karanovic, M., Sladojevic, S., Anderla, A. and Stefanovic, D. (2019). Solving Current Limitations of Deep Learning Based Approaches for Plant Disease Detection, *Symmetry* .

Barbedo, J. G. A. (2018). Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification, *Computers and Electronics in Agriculture* .

De Luna, R. G., Dadios, E. P. and Bandala, A. A. (2019). Automated Image Capturing System for Deep Learning-based Tomato Plant Leaf Disease Detection and Recognition, *IEEE Region 10 Annual International Conference, Proceedings/TENCON*.

Dhaware, C. G. and Wanjale, K. H. (2017). A modern approach for plant leaf disease classification which depends on leaf image processing, *2017 International Conference on Computer Communication and Informatics, ICCCI 2017*.

Durmus, H., Gunes, E. O. and Kirci, M. (2017). Disease detection on the leaves of the tomato plants by using deep learning, *2017 6th International Conference on Agro-Geoinformatics, Agro-Geoinformatics 2017*.

Fayyad, U., Piatetsky-Shapiro, G. and Smyth, P. (1996). The KDD Process for Extracting Useful Knowledge from Volumes of Data, *Communications of the ACM* .

Ferentinos, K. P. (2018). Deep learning models for plant disease detection and diagnosis, *Computers and Electronics in Agriculture* .

Francis, J., Anto Sahaya Dhas D and Anoop B K (2016). Identification of leaf diseases in pepper plants using soft computing techniques.

Francis, M. and Deisy, C. (2019). Disease Detection and Classification in Agricultural Plants Using Convolutional Neural Networks - A Visual Understanding, *2019 6th International Conference on Signal Processing and Integrated Networks, SPIN 2019*.

Fuentes, A., Yoon, S., Kim, S. C. and Park, D. S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition, *Sensors (Switzerland)* .

Ganesan, P., Sajiv, G. and Megalan Leo, L. (2018). CIELuv color space for identification and segmentation of disease affected plant leaves using fuzzy based approach, *ICONSTEM 2017 - Proceedings: 3rd IEEE International Conference on Science Technology, Engineering and Management*.

Gao, X., Fan, S., Li, X., Guo, Z., Zhang, H., Peng, Y. and Diao, X. (2017). An improved XGBoost based on weighted column subsampling for object classification, *2017 4th International Conference on Systems and Informatics, ICSAI 2017*.

Geetharamani, G. and J., A. P. (2019). Identification of plant leaf diseases using a nine-layer deep convolutional neural network, *Computers and Electrical Engineering* .

Georganos, S., Grippa, T., Vanhuysse, S., Lennert, M., Shimoni, M. and Wolff, E. (2018). Very High Resolution Object-Based Land Use-Land Cover Urban Classification Using Extreme Gradient Boosting, *IEEE Geoscience and Remote Sensing Letters* .

Hidayatuloh, A., Nursalman, M. and Nugraha, E. (2019). Identification of Tomato Plant Diseases by Leaf Image Using Squeezenet Model, *2018 International Conference on Information Technology Systems and Innovation, ICITSI 2018 - Proceedings*.

Mohanty, S. P., Hughes, D. P. and Salathé, M. (2016). Using deep learning for image-based plant disease detection, *Frontiers in Plant Science* .

Mudgal, T. K., Gupta, A., Jain, S. and Gusain, K. (2018). Automated system for Brain tumour detection and classification using eXtreme Gradient Boosted decision trees, *2017 International Conference on Soft Computing and its Engineering Applications: Harnessing Soft Computing Techniques for Smart and Better World, icSoftComp 2017*.

Pantazi, X. E., Moshou, D. and Tamouridou, A. A. (2019). Automated leaf disease detection in different crop species through image features analysis and One Class Classifiers, *Computers and Electronics in Agriculture* .

Pooja, V., Das, R. and Kanchana, V. (2018). Identification of plant leaf diseases using image processing techniques, *Proceedings - 2017 IEEE Technological Innovations in ICT for Agriculture and Rural Development, TIAR 2017*.

Ramcharan, A., Baranowski, K., McCloskey, P., Ahmed, B., Legg, J. and Hughes, D. P. (2017). Deep learning for image-based cassava disease detection, *Frontiers in Plant Science* .

Rangarajan, A. K., Purushothaman, R. and Ramesh, A. (2018). Tomato crop disease classification using pre-trained deep learning algorithm, *Procedia Computer Science*.

Ranjith, Anas, S., Badhusha, I., Ot, Z., Faseela, K. and Shelly, M. (2017). Cloud based automated irrigation and plant leaf disease detection system using an android application, *Proceedings of the International Conference on Electronics, Communication and Aerospace Technology, ICECA 2017*.

Sardogan, M., Tuncer, A. and Ozen, Y. (2018). Plant Leaf Disease Detection and Classification Based on CNN with LVQ Algorithm, *UBMK 2018 - 3rd International Conference on Computer Science and Engineering*.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition, *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

Singh, V. and Misra, A. K. (2017). Detection of plant leaf diseases using image segmentation and soft computing techniques, *Information Processing in Agriculture* .

Tm, P., Pranathi, A., Saiashritha, K., Chittaragi, N. B. and Koolagudi, S. G. (2018). Tomato Leaf Disease Detection Using Convolutional Neural Networks, *2018 11th International Conference on Contemporary Computing, IC3 2018*.

Too, E. C., Yujian, L., Njuki, S. and Yingchun, L. (2019). A comparative study of fine-tuning deep learning models for plant disease identification, *Computers and Electronics in Agriculture* .

Türkoğlu, M. and Hanbay, D. (2019). Plant disease and pest detection using deep learning-based features, *Turkish Journal of Electrical Engineering and Computer Sciences* .

Zhang, S., Zhang, S., Zhang, C., Wang, X. and Shi, Y. (2019). Cucumber leaf disease identification with global pooling dilated convolutional neural network, *Computers and Electronics in Agriculture* .