Identification and classification of leaf pests within the Indonesian Mango farms using Machine Learning

MSc Research Project
Data Analytics

Ritika Pramod Chendvenkar
Student ID: x19199473

School of Computing
National College of Ireland

Supervisor: Prof. Jorge Basilio

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Ritika Pramod Chendvenkar
Student ID:	x19199473
Programme:	Data Analytics
Year:	2021
Module:	MSc Research Project
Supervisor:	Prof. Jorge Basilio
Submission Due Date:	16/08/2021
Project Title:	Identification and classification of leaf pests within the Indonesian Mango farms using Machine Learning
Word Count:	7010
Page Count:	28

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:	21st September 2021

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):	

Identification and classification of leaf pests within the Indonesian Mango farms using Machine Learning

Ritika Pramod Chendvenkar
x19199473

Abstract

Indonesia being the fourth largest producer of mangoes worldwide, agriculture is considered to be one of the major contributors to the Indonesian economy. However, in the past few years there has been a fall in the contribution of the agricultural field in the national Gross Domestic Product (GDP). The infestation of the Indonesian Mango farms by harmful pests have had a direct impact on the country's economy. And the use of incorrect insecticides and pesticides can double the chances of having a poor yield from the crop and also indirectly affect animals and humans consuming the harvest. Machine Learning techniques applied to large agricultural datasets are capable of extracting valuable insights that can aid farmers in order to perform diagnosis of the leaf diseases. To achieve this, SVM and boosting based models like XGBoost and CatBoost are used for classification. Apart from that, a CNN built from scratch is built. Various experiments were conducted to evaluate the model performance. Convolutional Neural Network outperformed the other models with accuracy of 72.06%. This CNN model was taken as a baseline to create a python-based webpage to identify and classify infested mango leaf images.

Index Terms – **Computer Vision, Image Processing, Leaf Pest Identification, Convolutional Neural Network, Multi-class Support Vector Machine, Machine Learning**

Contents

List of Figures

List of Tables

1 Introduction

1.1 An Overview of Indonesian Mango production

Indonesia is one of the foremost agricultural marketplaces in the Asian continent due to the presence of abundant rich land and favourable climatic conditions. It is not just one of the largest producer of mangoes in the Asian subcontinent, but also globally. Figure 1 shows the production of mangoes in Indonesia from 2010 to 2019, measured in million metric tons. From the figure it can be clearly seen that the production of mangoes has increased considerably over the years.

Figure 1: Production of mangoes in Indonesia from 2010 to 2019[1]

1.2 Motivation

The last decade has shown decrease in the contribution of the agricultural field in the national Gross Domestic Product (GDP) of Indonesia. Figure 2 from the Data World Bank shows the contribution of agricultural field in the total percentage of the Gross Domestic Product (GDP) of Indonesia for the past two decades. The percentage contribution shows a dramatic fall over the years. From nearly 19.6% in 1999, the contribution has plummeted to 12.7% in 2019.

The figures indicated above are alarming as it has a direct impact on the economy of the country. A major issue faced in the field of agriculture is that of pest infestation. The leaves and fruits are prone to get infested by different type of pests. Taking a note of the case study based on Luzon region in the Philippines (Cardinoza G. (2018)), the key factor contributing to the deteriorated yield and quality of the mango fruit was the occurrence of pest and diseases. This accounted for the expensive production owing to the lack of understanding and ineffective pesticide application by the farmers. In the past, prediction tasks were carried out using machine learning tools whereas data modelling largely relied on statistical methods. However, as time passed, researchers have demonstrated that these two methods can be combined together to process and analyse data and to use the input data to train the machine learning models, either for prediction or to make a

[1]Source: `https://www.statista.com/statistics/706514/production-of-mango-in-indonesia/`

[2]Source: `https://data.worldbank.org/indicator/NV.AGR.TOTL.ZS?contextual=default&end=2019&locations=ID&start=1999&view=chart`

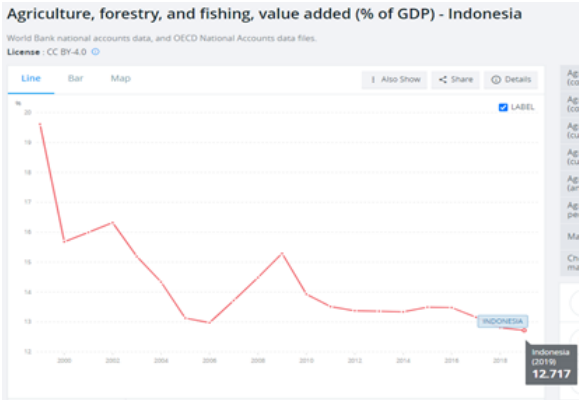

Figure 2: Percentage contribution of agriculture sector in Indonesian GDP [2]

decision. In this work, we would be working on computer vision. We will be deploying and testing a machine learning based framework in image category classification. In this we will be training the machine learning models to predict whether the leaf is infested by a pest, or it is a healthy leaf and then would classify the infected leaf into the type of pest that has infested it, which would help in early identification of the pest and determining the suitable pesticide to be sprayed on and thereby can improve the yield of the crop.

1.3 Research Question

"Can a Multi-class Support Vector Machine model with GLCM based feature extraction statistically outperform Convolutional Neural Network in classifying pest-infested mango leaf images into various categories of pests?"
The main aim of the research is to build deep learning models based on Convolutional Neural Networks that makes use of transfer learning techniques with hidden layers that can be trained, along with image augmentation methods to build an efficient model and compare it with a traditional multi-class supervised machine learning model like Support Vector Machine which uses Grey Level Co-occurrence Matrix (GLCM) technique used for feature extraction in combination with feature selection technique, Principle Component Analysis (PCA). To the best of my knowledge, these techniques are yet to be entirely explored in mango leaf pest classification and can turn out to be of immense value to the farmers in the diagnosis of their crops if desirable results are obtained.

1.4 Research Objective

To meet the above research question, following objectives are defined:

- Critically reviewing the research papers related to Leaf Disease Detection and identifying the best suited techniques.
- Gathering the data consisting of images from the Indonesian mango farms captured using a low-cost sensing equipment, similar to what is used by the farmers.
- Performing data augmentation on the images to expand the dataset in order to train the machine learning models to avoid overfitting.

- Extracting texture features which can best contribute to pest classification model using Gray Level Co-occurrence Matrix (GLCM).
- Performing dimensionality reduction of the features using Principal Component Analysis (PCA) technique.
- Handling class imbalance through various experiments.
- Implementing Support Vector Machine (SVM) model on the final dataset.
- Designing a Convolutional Neural Network from scratch and identify the evaluation parameters.
- Implementing Boosting algorithms like XGBoost and CatBoost to compare performance.
- Comparing the model performance of the designed models.
- Finally, building a webpage to facilitate user to upload images of infected leaf to obtain the class of pests.

1.5 Contribution

Traditional methods of crop disease management involved farmers and crop pathologists, and the diagnostic process used to be carried out in the fields. Not only was this time consuming but also many a times resulted in incorrect diagnosis. With the advent of Computer Vision and Machine Learning, there has been a significant progress in the development of automated models thereby enabling precise and timely detection of plant leaves' disease. Obtaining information about the type of pest that has infested the crop can help the farmers to eliminate the pests in an appropriate fashion. The classification of these pests provides a solution to the farmers in finding and applying the specific type of insecticide and pesticides in the right quantity and at the correct frequency. In a nutshell, early detection of the pests on a crop can minimize the use of pesticides and thus increase the overall productivity of the crop.

1.6 Structure of paper

Further, the paper is structured as follows, Section 2 presents the review of related previously done work, Section 3 explains the methodology adopted for classifying mango leaf pests and specifications related to the design, Section 4 focuses on the end to end steps involved in each stage of the implementation of this research project, Section 5 illustrates the implementation of the machine learning models and discussion on the model evaluation methods, Section 6 discusses the findings of the research. Finally, Section 7 concludes the findings of the conducted research work with suggestions about the future work, with acknowledgement following it.

2 Related Work

2.1 Introduction

This section contains a comprehensive summary of state-of-the-art results achieved by researchers with relevant image processing, traditional and deep learning methods for classification of images from 2011 to 2021. A methodical reviewing approach has been followed and the findings have been summarized. This segment covers the point by point subjective and quantitative review of different research embraced utilizing machine learning based methods in the field of computer vision and image processing.

2.2 Computer Vision and its evolution in the agriculture field

For quite a while now, agriculture has been an area stringently reliant on climate conditions and susceptible to climatic changes. Computer vision has emerged over the years from initially built to mimic the human eye, to being able to recognise patterns and perform machine learning tasks in the last decade, to finally performing high end tasks such as facial expression detection, object recognition, robotic navigation and may others.

In this research, Kakani et al. (2020) investigate emerging technologies such as computer vision and artificial intelligence and their applications in the food and agriculture industry. The research states that until the last decade, farmers would depend on satellite imaging which would take up to 15 days for retrieval of images. With the latest technology, farmers now make use of infrared sensory imaging for identification of disease plants. The study also sheds light on the challenges in the agricultural field such as crop management, pest infestation and management and smart irrigation, among others, that can be tackled using computer vision and artificial intelligence.

In the research conducted by Tripathi and Maktedar (2020), they performed a comprehensive survey and analysis of 98 research papers closely related to computer vision in the field of agriculture. The authors have focused their study on computer vision in fruit and vegetable areas. They have also compared various machine learning techniques and performance evaluation metrics. Their findings indicate that SVM performed the best among all the classifiers giving the highest accuracy. The authors in conclusion opinionated that computer vision has a great potential to handle real time problems in the field of agriculture.

2.3 Use of machine learning in plant disease identification and classification

Traditionally the task of leaf disease diagnosis was performed by experts. This process was time consuming and also very expensive. With the advent of technology, various techniques such as computer vision, deep learning and artificial intelligence have gained popularity not just for reducing the human effort but also for their ability to deal with complex data. They have gained light for their pattern recognition, regression, and classification precision.

The authors have carried out their research based on the available pest identification techniques and then proposed a novel method for classification of crop pests. The proposed methodology involves a Convolutional Neural Network (CNN) applied on a relatively large dataset of 9500 images. This research, similar to ours, also involves multiple classes of pests. The accuracy of the proposed model is found to be 90% which surpassed the traditional classification models. Malek et al. (2021)

Chouhan et al. (2019) proposed a multilayer convolutional neural network for the classification of diseased leaf images from four different trees. The accuracy of the proposed model was found to be 98.58%. The challenges faced by the researchers revolve around noise in the images that include excessive light, temperature, overlapping leaves. It can be inferred that by handling these challenges using the appropriate segmentation and image pre-processing techniques would result in improved accuracy of the model. In a similar study undertaken by Singh et al. (2019), a multilayer convolutional neural network has been implemented exclusively to classify anthracnose disease on the leaves of mango trees. The accuracy was found to be 97.13% with similar constraints of temperature, overlapping and shadowing.

The methodology of data augmentation was further researched upon and implemented by Hong et al. (2020) in their research wherein a comparison has been performed between five deep CNN models which were trained and tested using images of diseased leaves from the tomato crop. The research shed light on the fact that data augmentation has in fact improved the performance of the model by increasing the number of training images, thereby enhancing the efficiency of model training.

Nesarajan et al. (2020) used an approach similar to the one used by Shijie et al. (2017) for the classification of leaf pests on coconut trees. CNN and SVM models were implemented, and the study also focused on image segmentation and feature extraction methods before classification.

It is not always true that a leaf is infested by a single pest. There may be occurrence of multiple pests on a single leaf. (Tetila et al. (2020)) in their research developed a convolutional neural network to count the number of pests on the leaves of soybean crop. Their model achieved an accuracy of 94.89%. This application proves to be useful when the leaves are infested by various pests.

A recent study in the area of pest classification is that done by Swathika et al. (2021) where a CNN based deep learning approach is followed in order to detect pests from the paddy leaves. The CNN model consists of three convolutional layers and their model achieved an accuracy of about 70% after 10 epochs. The images after classification are subject to contour detection. This method helps in the estimation of the exact affected area of the leaves.

2.4 Image Pre-processing techniques

Image Pre-processing a broad term with a number of techniques as a part of it. It includes Image Cleaning, Image Segmentation and Feature Extraction. Image cleaning methods include resizing, median filtering, histogram equalization, noise removal, among others. The most common segmentation methods include K-means clustering and edge detection. Feature extraction is the process of drawing out important factors from the image. The various features are shape, texture, colour, SIFT and SURF features. This sections describes the various image pre-processing techniques used as a part of previously done research.

Iqbal et al. (2018) have compared and reviewed various image cleaning, segmentation, and feature extraction techniques. The comparison has been done based on their technique, performance, pros, and cons. From the research it was inferred that K-means clustering is the most prominent technique used for the segmentation of diseased leaves.

Also, the most import features are the texture features that are fed into the classification model.

As far as Feature Extraction is considered, it is the most crucial step in any machine learning algorithm for classifying images as the classifiers are applied on the extracted features. The study undertaken by Iqbal et al. (2018) the texture feature are the most commonly used features in classification tasks. One such feature extraction technique is Gray Level Co-occurrence Matrix (GLCM) where features such as contrast, energy, correlation etc. of an image are obtained. In the research carried out by Massot-Campos et al. (2013) decision tree classifier has been implemented after extracting features using GLCM in order to classify underwater plants. The research not only gives a high accuracy, but also a low false negative and false positive rate.

Di et al. (2015) used GLCM to analyse texture of the spots on the sunflower leaf. The researchers also used colour features in combination with texture features to enhance the performance. Also, median filtering method is used for image cleaning which has not hampered the image quality. Segmentation has been carried out using edge detection method.

With the passing years, machine learning came into play and image processing techniques in combination with machine learning algorithms had a huge impact on the performance of the model. The extracted features we fed into the model to achieve better results. Bhimte and Thool (2018) in their research combined K-means clustering technique for image segmentation with GLCM for texture feature extraction. The output was then fed into the SVM classifier to detect cotton leaf spots.

Recent advancements in the field of image processing have shown hybrid approach in the pre-processing of images. In one such research by Priyanka and Kumar (2020) a hybrid approach for feature extraction is used on grayscale ultrasonic medical images. In this, GLCM is used for feature extraction and PCA (Principal Component Analysis) is used on the extracted features to select the optimal subset of features from them. Their proposed method consumes less computation time. In our research we are trying to incorporate the similar approach for coloured images of plant leaves.

2.5 Multi-class Support Vector Machine for image classification

In the years when neural networks and deep learning techniques were still in their budding stage, researchers relied on traditional techniques to perform machine learning tasks. These traditional algorithms were very effective in both classification and regression problems and proved to reduce human intervention. The most commonly used traditional machine learning algorithm is the Support Vector Machine (SVM). Numerous research has shown that SVM is a better performing model than many other algorithms.

Early applications of SVM show the use of a single class. One class classification was majorly used for remote sensing and medical images in pattern recognition problems. In the research carried out by Pla et al. (2013) one class SVM has been used to detect the vegetation from remote sensing images. Their model achieved a True Positive rate of 100% and False Positive rate of 9%.

Support Vector Machine is a linear algorithm. A kernel function in SVM plays a key role. It helps a linear model in solving a non-linear problem by mapping the data into a higher dimensional space. Tangthaikwan et al. (2017) experimented with four different kernel functions: Linear, Quadratic, Polynomial and Radial Basis Function (RBF) to classify satellite images according to seven different types of land use. Their findings indicated that SVM with RBF kernel gave the highest accuracy of 90.89% with a sigma value of 1.7. Alam et al. (2018) used a similar approach with the four kernels as used by Tangthaikwan et al. (2017) to classify grayscale ultrasonic lung images to detect cancer. It was observed that the Linear Kernel achieved the highest overall accuracy of 90%. It can be inferred that the Linear Kernel works best with grayscale images whereas for colour images, RBF kernel is the best suited choice. Indrabayu et al. (2019) developed an automated multiclass classification system for strawberry ripeness measurement based on the skin tone using SVM - RBF kernel approach. This multi-class SVM classifier with RBF kernel has displayed a superior classification performance with colour images achieving an overall accuracy of 85.64% which confirms the mentioned inference.

More recently, Tumang (2019) used Multi-class SVM to classify diseased leaves of mango trees. After the cleaning of images, K-means segmentation has been performed and the segmented images are subjected to GLCM feature extraction. The extracted features are applied as input to the SVM classifier. Although the model achieved accuracy of 85%, the shortcoming of this research lies in the exploration of RGB values of the images. In our research we will be carrying out analysis among the RGB values of the leaf images.

While SVM is one of the better performing traditional ML model, one major drawback of the traditional methods is that there is a need of hand-engineered features as they are unable to learn different features in images on their own. This is where deep learning methods come in place. They are broadly used for challenges pertaining to computer vision challenges (Pantazi et al. (2019)) since they are capable of automatically extracting features and usually achieve a higher accuracy as compared to traditional machine learning methods for image classification.

2.6 Boosting methods for image classification

XGBoost is one of the most commonly used boosting algorithm for its computational abilities. Another similar algorithm is CatBoost. It is very similar to Random Forest classifier; however, the only difference lies in the way the individual trees are built and the way the results are combined. It works on the principle of 'boosting'. Boosting combines weak learners sequentially so that each new tree corrects the errors of the previous tree. Gradient boosting only takes numeric inputs, so categorical features need to be transformed. The most commonly used approach to transform categorical data into numeric data is One Hot Encoding (OHE), but there are few potential setbacks. CatBoost uses a new method of transforming categorical to numeric. CatBoost is capable of outperforming XGBoost. To enhance the performance of classification models, a deep neural network based on XGBoost is proposed in the research by Song et al. (2020). From their experiments, it is clear that XGBoost definitely enhances the performance of the models. Their model gave an accuracy of around 80% when used without XGBoost, while XGBoost increases the accuracy to about 93%. Similarly, Fangoh and Selim (2020) uses CNN-XGBoost model to enhance the performance of CNN in identifying COVID-19 based on the chest X-rays. Their model produced a remarkable F1 score of 91.98%, however, with further testing and training, it is possible to achieve better results. Pham et al. (2019) suggested XGBoost classifier to classify melanoma images. Their research yielded a recall of 44% and specificity of 69.36%. Their findings suggested that as the labelling was done in a generic manner, this led to lowering of the values. In the study by Samat et al. (2021), CatBoost was used for hyperspectral image classification. The model was evaluated on its accuracy and the computational properties. Their research compared GPU-based CatBoost and CPU-based CatBoost and it was observed that GPU-based CatBoost outperforms all other methods. In the research carried out by Samat et al. (2020), CatBoost was used to classify remote sensing images for the very first time. In their research, CatBoost reduced the overfitting issue. It was found that CatBoost has excellent performance as compared to GDBT, LightGBM and XGBoost. CatBoost demonstrated high classification accuracy but low computational efficiency.

2.7 Convolutional Neural Networks for image classification

Artificial Intelligence has witnessed an enormous development in overcoming any barrier between the abilities of people and machines. Scientists work on various parts of the field to make astounding things occur. One of numerous such territories is the domain of Computer Vision. The main aim of this field is to empower machines to see the world as people do, see it along these lines and even utilize the information for a huge number of assignments like Image and Video processing, Image Analysis and Classification, Natural Language Processing, and so forth. The evolution in the field of Computer Vision with Deep Learning has been developed and consummated with time, essentially over one specific algorithm — a Convolutional Neural Network.

Convolutional Neural Networks are known for their ability to work on images. They paved their way and had most of the success in the study of medical images. Numerous researchers have applied CNN models in the classification of ultrasonic images. One such research carried out by Betul Oktay (2017) presented a tooth detection method using a CNN on dental X-Ray images. In this paper AlexNet is used for teeth detection. AlexNet is a popular CNN architecture that uses the sliding window technique, and the output of the CNN is four classes. the model achieved an overall accuracy of 90% where the individual accuracy for each class is 94.32% for Molars, 91.74% for Premolars and 92.47% for Canines and Incisors. The limitation observed in this model is the incorrect identification of premolars as the neighbouring canines and molars have a similar tooth structure leading to misclassification.

A Hybrid model has been developed by Hameed et al. (2018) to perform multi-class skin disease classification. The proposed hybrid approach is a combination of Deep CNN and SVM for classifying skin lesion images into one of the five classes. Pretrained complex Deep CNN model, AlexNet is used for feture engineering and the classification task is performed using SVM classifier. The hybrid model produced an overall accuracy of 86.21%. The uniqueness of the research is the use of a relatively larger dataset comprising of around 9000 images and successful handling of issues pertaining to overfitting using k-fold cross validation techniques.

Another study done by Tropea and Fedele (2019) compared five most popular traditional machine learning classifiers to classify images of objects belonging to 256 classes. Multiclass Logistic Regression (MLR), Support Vector Machine (SVM), k-Nearest Neighbours (kNN), Random Forest (RF) and Gaussian Naive Bayes (GNB) were used in a Convolutional Neural Network (CNN) for image classification purpose. The approach used in this study was that CNN would be used for Feature Extraction and then the classification models would be applied. The best performing model was chosen by evaluating the accuracy and the computation time. Using these two parameters, it was observed that kNN had the best performance.

In the recent years, there has been numerous research in the area of leaf pests and disease classification. While some of these have been discussed above in section 2.3, few research that follows demonstrates the exclusive use of Convolutional Neural Nets. The proposed CNN consists of five convolutional layers Truong et al. (2018). A ReLU activation function is used after each conv layer, and MaxPooling approach is applied for each pooling layer. Finally, there is a fully connected layer, after which SoftMax is placed. Data augmentation is also performed to increase the data size. Their model achieved an accuracy of 98.14% on the original dataset and accuracy of 99.35% on the augmented dataset. The results demonstrate the impact of data augmentation technique when used with a Convolutional Neural Network.

As discussed in Wongbongkotpaisan and Phumeechanya (2021), CNN has proved to perform well with augmented data in classifying leaf diseases. In their research, they proposed a simple CNN architecture for classifying left pests and their model gave an accuracy of around 95.65%. We shall be using the same CNN architecture for our research. We are using a CNN that consists of 3 convolutional layers, two of which are followed by max pooling layers. The output from the final convolutional layer is flattened to convert it into a 1-dimensional array and is fed into the final fully connected layer. Finally, the model is compiled, and accuracy is used as the metric to evaluate the performance of the CNN. Convolutional Neural Networks not only provide better accuracy, but also reduces the need for human intervention to a great extent.

3 Research Methodology

CRISP-DM Methodology

The implementation of this research work has been carried out based on the CRISP-DM methodology. Fig. 3 below shows the various stages of this research which resembles the stages of the Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology. The leaf pest identification process as shown in the figure, is spread over 6 main stages namely: understanding the agriculture business and the underlying flow of the pesticide application process, data gathering, data pre-processing, modelling and followed by the evaluation of machine learning model and finally, model deployment onto the right platform to verify its working with the test data. The generated prediction results could then be used to apply pest management technique.

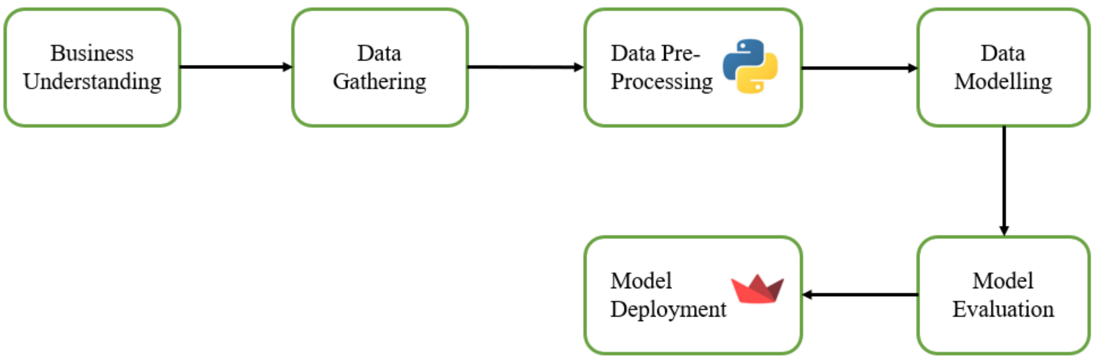

Figure 3: Pest Identification Methodology

3.1 Business Understanding

As suggested by Sfiligoj E. (2020), the farming industry is continually dealing with several challenges, such as extreme climatic conditions, depleting soil health, infestation of pests, lack of knowledge possessed by farmers, and many more. In the business understanding phase, the pesticide application flow needs to be understood to figure out at which stage the crop gets harmed and the soil health degrades. Also, the process of selection of pesticides by the farmers needs to be understood. This will help us achieve our objective in regard to early detection of the type of pest and the use of the appropriate pesticide to target it.

3.2 Data Gathering

The dataset was acquired from Mendeley Data, which is an online data repository wherein data is publicly available for academic research. The dataset consists of images in .jpg format. The datasets consist of images captured using a low-cost sensing device. The total number of classes in the dataset is 16 and the images belong to one of these 16 classes.

Dataset	Number of Images
Training Set	9592
Testing Set	2408
Total	12000

Table 1: Data Structure

3.3 Data Pre-processing

As a part of pre-processing the data, the first and the most important step is data augmentation. The augmentation was performed to increase the size of the dataset, as the number of images in the downloaded dataset were only 510, it would result in overfitting the model, thus some variations were introduced in the images and the final count of images was increased to 12000. The augmentation is performed using 'Augmentor' package in python. This package works by creating a pipeline where the operations to be performed on the images is defined. The operations that were applied on the image were as follows:

1. Zooming: The images were zoomed with a probability of 0.5 with a scale of up to 80%.

2. Adjusting Brightness: Here, the brightness was adjusted with a probability of 0.3, whereas the scale is selected from between 0.3 and 1.2 at random.

3. Adding Distortions: Added random distortions to the images.

4. Rotation: Using the rotate180() function, the image is simply rotated by 180 degrees and saved.

5. Skewness: The image is skewed by tilting the image in a random direction with a probability of 0.7, the magnitude of the tilt is set to 45 degrees.

6. Resizing: Most of the images in the original dataset were of the dimension, 500 x 333. Thus, all the images were resized to 500 x 333.

The next step in data pre-processing is the Feature Extraction. For this research, I have used a combination of two feature extraction techniques, namely, Gray Level Cooccurrence Matrix (GLCM), which is a feature extraction technique that extracts the texture features from the images. The other technique used is the Principal Component Analysis (PCA), which is also a dimensionality reduction method. Using GLCM, we were successfully able to extract 80 features with 5 base texture features which are: homogeneity, contrast, energy, dissimilarity, and Angular Second Moment (ASM). These 80 features were then subject to dimensionality reduction, in which we extract the most relevant features out of the 80, closest to the highest power of 2. So, as a part of this, we have outputted 64 of the most relevant texture features. After the feature extraction part is completed, the data preparation and modelling is done which will be covered in the implementation part, i.e., Section 5 of this report.

4 Design Specification

The architecture followed for this research is described in Figure 4.

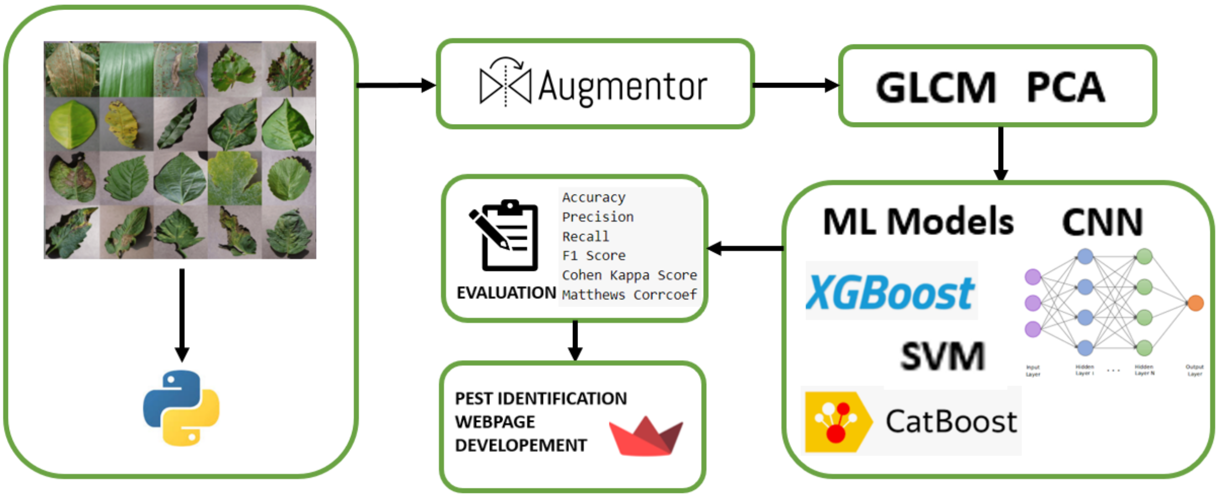

Figure 4: Pest Type Prediction - Design Flow

- In the initial stage, the dataset required to carry out the research work is downloaded from the Mendeley data repository.
- The dataset was then loaded into python where the images are then processed. As the number of images were roughly around 550, the machine learning models were susceptible to overfitting. The images are thus subject to data augmentation techniques which was used to increase the number of images in the dataset by introducing diversity in the data. The data augmentation is performed using the 'Augmentor' package in python and it resulted in 12000 sample images.

- Train-test split is then applied in 80:20 ratio respectively, followed by implementation of machine learning classification models and a convolutional neural network built from scratch.
- For the purpose of implementation of machine learning models, I have chosen to use a combination of two feature extraction techniques, namely Grey Level Cooccurrence Matrix (GLCM) along with Principal Component Analysis (PCA).
- The machine learning models were then evaluated using precision, recall, accuracy, F1 score, Cohen Kappa Score and Matthews Correlation Coefficient.
- Lastly, as an extension to the research, the best performing model was evaluated and used to build a webpage, where user can upload the image of the infected leaf and the name of the pest that has infested the leaf is obtained.

5 Implementation

This section discusses all the steps involved in the end-to-end implementation of this research. The section describes the techniques used for data preparation, feature extraction and modelling of our classification algorithms.

5.1 Data Preparation

For the implementation of machine learning models like SVM, XGBoost and CatBoost, we make use of the final dataset that is obtained after augmentation and application of feature engineering techniques i.e., GLCM and PCA. This final training dataset consists of 9592 rows and 64 columns. The 64 columns represent the texture features with reduced dimensionality and the 9592 rows represent the total number of images. Similarly, the testing dataset consists of 64 columns and 2408 cases. The training data was then split into training and validation set in the ratio 80:20. Table 2 shows size of the train, test, and validation datasets. The class distribution was also observed for the dataset before and after data augmentation. Figure 5 shows the class distribution for dataset before and after the augmentation was performed.

	Total Size
Training Set	7673
Testing Set	2408
Validation Set	1919
Total	**12000**

Table 2: Dataset Size Details

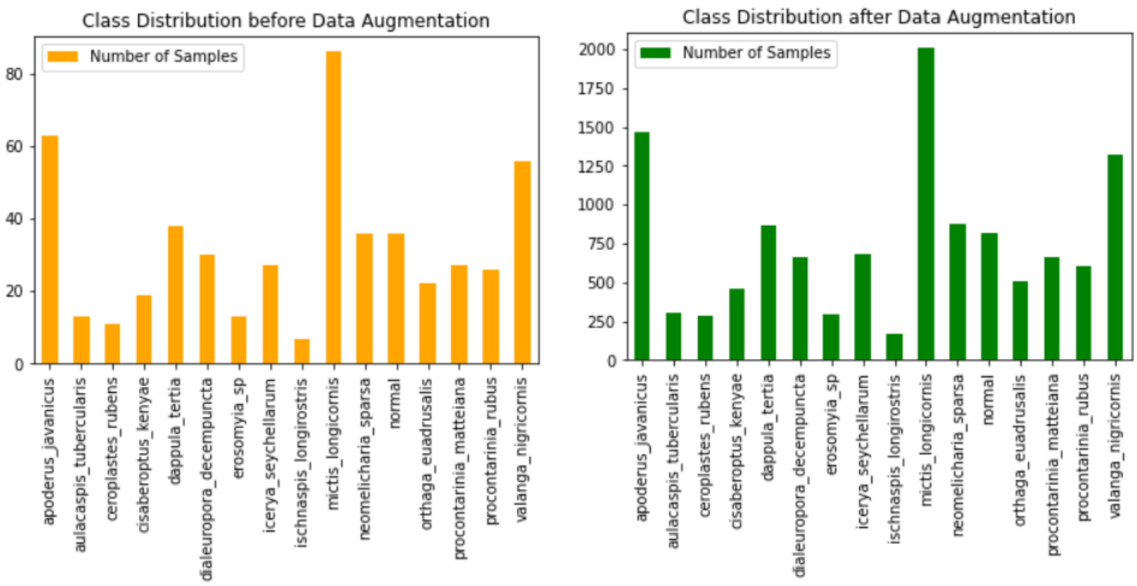

Figure 5: Pest Type Prediction - Class distribution before and after data augmentation was performed

5.2 Support Vector Machine (SVM) Classifier

The Support Vector Machine (SVM) is a supervised machine learning algorithm that performs well with both, regression as well as classification problems. SVM classifier takes into consideration the extremities of the dataset and draws a hyperplane, which is basically a decision boundary. Support vectors are the data points that are closest to the opposing class that the margin pushes up against. In cases where a straight line or a linear function is not feasible to separate the data, we can use a function to transfer our data into higher dimensional space. Once the data is transported to a higher dimension, it is simple to draw a hyperplane. In computing, we use the kernel function to draw the hyperplane after transforming the data into a higher dimensional plane. Kernel function is a function that takes in its input the vectors in its original space and returns the dot product of the vectors in the feature space. Essentially a kernel function is used to transform a nonlinear space into a linear space. Some popular kernels are Polynomial, Radial Basis Function (RBF) and Sigmoid, amongst others.

5.3 Extreme Gradient Boosting Classifier (XGBoost)

It is an ensembling method that perform regression or classification by combining the output from individual trees. It is very similar to Random Forest classifier; however, the only difference lies in the way the individual trees are built and the way the results are combined. It works on the principle of 'boosting'. Boosting combines weak learners sequentially so that each new tree corrects the errors of the previous tree. As discussed in Song et al. (2020), XGBoost has proved to perform better than SVM in classifying breast tumours. The accuracy for their classification model using XGBoost was found to be 92.80%. Their study used a similar GLCM-based feature extraction approach as we are using in our research.

5.4 CatBoost Classifier (Gradient Boosting based)

Gradient boosting only takes numeric inputs, so categorical features need to be transformed. The most commonly used approach to transform categorical data into numeric data is One Hot Encoding (OHE), but there are few potential setbacks. CatBoost uses a new method of transforming categorical to numeric, which is derived from Online Click Prediction where features and models are updated over time with each new click. To begin with, the data is randomly ordered to artificially construct time. Then for each categorical variable, new numeric features are created by taking a mean of the target by taking the mean of the target but only using historical observations to avoid creating biasness. Additionally, we blend against an a-priori to increase stability for early time periods. CatBoost is capable of outperforming XGBoost. The only drawback is that it takes longer to run.

5.5 Convolutional Neural Network (CNN)

Neural Network (NN) works like the human brain to identify pattern and relationship between the data. A convolutional neural network (CNN) is a type of neural network that is most often applied to image processing problems. Since CNNs treat data as spatial, they are different in the way they work. The term convolutional refers to the filtering process that happens in this type of network. Similar to normal neural networks, CNNs are made up of various layers namely Convolution, Pooling, Dense and Fully Connected. As discussed in Wongbongkotpaisan and Phumeechanya (2021), CNN has proved to perform well with augmented data in classifying leaf diseases. They proposed a simple CNN architecture for classifying left pests and their model gave an accuracy of around 95.65%. We have used the same CNN architecture for our research. Table 3 shows the dimension and parameters for input, convolutional and output layers along with the activation function that is used for each of those layers. We are using a CNN that consists of 3 convolutional layers, two of which are followed by max pooling layers. The output from the final convolutional layer is flattened to convert it into a 1-dimensional array and is fed into the final fully connected layer. Finally, the model is compiled, and accuracy is used as the metric to evaluate the performance of the CNN. The loss function used is the 'Sparse Categorical Cross entropy' and 'adam' is used as an optimizer function adjust the learning rate. Figure 6 shows the CNN Model Summary.

Layer	**Input Dimension**	**Activation Function**
Input	224	
1st Convolutional	3,3,32	relu
1st Max Pooling	2,2	
2nd Convolutional	3,3,64	relu
2nd Max Pooling	2,2	
3rd Convolutional	3,3,128	relu
Flatten	-	
Fully Connected	Number of classes (16)	relu / SoftMax

Table 3: Convolutional Neural Network Model Architecture

```
Model: "sequential"
_________________________________________________________________
Layer (type)                Output Shape              Param #
=================================================================
rescaling (Rescaling)       (None, 224, 224, 3)       0

_________________________________________________________________
conv2d (Conv2D)             (None, 222, 222, 32)      896

_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 111, 111, 32)     0

_________________________________________________________________
conv2d_1 (Conv2D)           (None, 109, 109, 64)      18496

_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 54, 54, 64)       0

_________________________________________________________________
conv2d_2 (Conv2D)           (None, 52, 52, 128)       73856

_________________________________________________________________
flatten (Flatten)           (None, 346112)            0

_________________________________________________________________
dense (Dense)               (None, 16)                5537808

=================================================================
Total params: 5,631,056
Trainable params: 5,631,056
Non-trainable params: 0
```

Figure 6: CNN Model Summary

5.6 Parameter Optimization Techniques

For the neural network (CNN), I have made use of Adam optimizer for parameter optimization. It was the best choice as it is easy to implement, computationally efficient and has low memory requirement. The choice of optimizer largely depends on the problem at hand, and Adam was thus, the best choice as it is well-suited for problems dealing with large datasets or large number of parameters. In this case, the dataset was large and hence adam was chosen as it achieves good results with faster computation.

For the SVM model, I have used Grid Search technique for optimization. I have used the GridSearchCV function of the Scikit-learn package. This optimization technique deals with trying out all possible combinations of hyperparameters and evaluating the models based on these combinations. The best performing combination was selected on the basis of the 'score'. In my case, the hyperparameters for SVM model were 'c' and 'gamma'. The possible values of c were specified to be [0.1, 1, 10, 100, 1000] and that of gamma were [1, 0.1, 0.01, 0.001, 0.0001]. Out of all these combinations, it was found that value of c=1 and gamma=0.0001 gave the highest score of 0.422 and thus, these hyperparameters were chosen to implement the SVM model.

5.7 Webpage for Pest Identification using CNN

This neural network is taken as a base to create a webpage for identification of pests. Users simply uploads the image of the diseased leaf, and the webpage is able to classify the image and give the appropriate class of pest. This webpage is made using the python 'Streamlit' package. This package helps to create machine learning based web applications. For this project, we have built a webpage for pest identification using python script and model file and pushing them into streamlit by just pushing it through GitHub using GitBash. This webpage is not just accessible through the laptop / computer but is also compatible to use on mobiles and tablets, thereby making it easily accessible and user friendly.

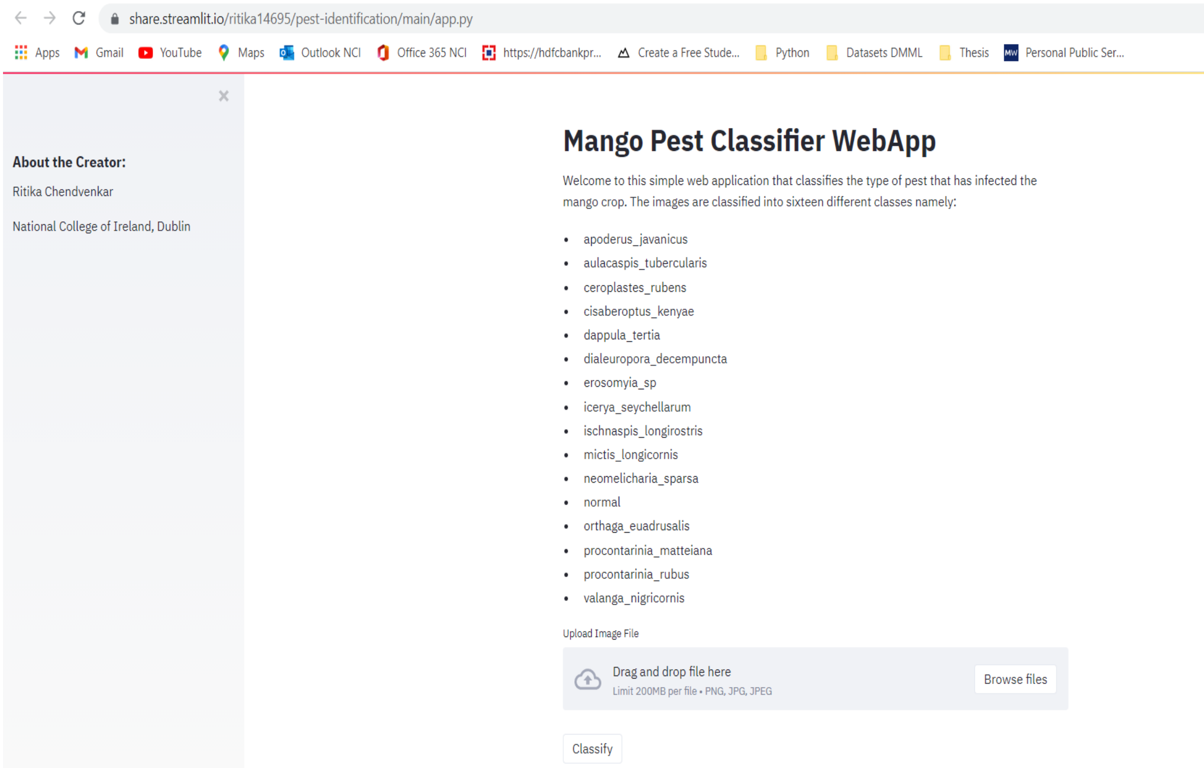

Figure 7: Simple UI of the designed Webpage

6 Evaluation

The models were evaluated on the basis of accuracy, precision, recall, F1 score, Cohen Kappa Score, and Matthews Correlation Coefficient. The obtained results and the various experiments with different machine learning models will be explained in this section of the report.

6.1 Feature Extraction experiment using GLCM and PCA

6.1.1 Experiment with SVM Classifier

The Table 4 below shows the results obtained when the SVM model was run and a combination of both, GLCM and PCA is used for feature extraction. The accuracy of the model was found to be 33.8%, increasing to 42.9% after hyperparameter tuning, which was still very low. And hence, it was decided to only move forward with GLCM based feature extraction and not use PCA.

	Before Hyperparameter Tuning — After Hyperparameter Tuning			
Evaluation Parameters	Test Results	Validation Results	Test Results	Validation Results
Accuracy	0.338	0.3299	0.429	0.4221
Precision	0.2977	0.2637	0.4442	0.443
Recall	0.338	0.3299	0.429	0.4221
F1 Score	0.2859	0.2781	0.4077	0.404
Cohen Kappa Score	0.258	0.2483	0.3664	0.3579
Matthews Corrcoef	0.2633	0.2533	0.3697	0.3616

Table 4: Results of SVM Classifier with GLCM and PCA

6.2 Feature Extraction experiment using only GLCM

Since the combination of GLCM and PCA did not give good accuracy, it was decided to only move forward with GLCM based feature extraction technique.

6.2.1 Experiment with SVM Classifier

Similar to section 6.1.1, SVM classifier was implemented. However, this time we did not make use of Principle Component Analysis (PCA). The accuracy of the model was found to be 30.98% which was still lower than that with GLCM and PCA, but on performing hyperparameter tuning, the test accuracy increased to 43.36%. Table. 5 below shows the test and validation results obtained when the SVM model was run on GLCM based feature extraction.

	Before Hyperparameter Tuning — After Hyperparameter Tuning			
Evaluation Parameters	Test Results	Validation Results	Test Results	Validation Results
Accuracy	0.3098	0.2824	0.4431	0.4336
Precision	0.2297	0.2047	0.4593	0.4615
Recall	0.3098	0.2824	0.4431	0.4336
F1 Score	0.246	0.2193	0.4205	0.4215
Cohen Kappa Score	0.2243	0.1931	0.3816	0.3723
Matthews Corrcoef	0.2312	0.1998	0.3855	0.3751

Table 5: Results of SVM Classifier with GLCM

6.2.2 Experiment with XGBoost Classifier

As the accuracy of the SVM classifier was lower, we decided to use more robust classifiers based on decision trees. We have made use of XGBoost classifier along with GLCM features to classify the images. The XGBoost classifier gave a test accuracy of 64.66%.

6.2.3 Experiment with CatBoost Classifier

As CatBoost is a more enhanced version of XGBoost, we decided to use this algorithm to increase the accuracy. This CatBoost model gave a test accuracy of 65.82%, which was overall the best among all the classifiers. Table 6 below shows the test and validation results obtained for XGBoost and CatBoost. Figure 8 shows a graphical representation of the same.

XGBoost ——— CatBoost				
Evaluation Parameters	Test Results	Validation Results	Test Results	Validation Results
Accuracy	0.6466	0.6524	0.6582	0.6566
Precision	0.6451	0.6504	0.6558	0.6551
Recall	0.6466	0.6524	0.6582	0.6566
F1 Score	0.6429	0.6482	0.6524	0.6511
Cohen Kappa Score	0.611	0.6174	0.624	0.6224
Matthews Corrcoef	0.6115	0.6178	0.6246	0.6229

Table 6: Results of XGBoost and CatBoost Classifiers

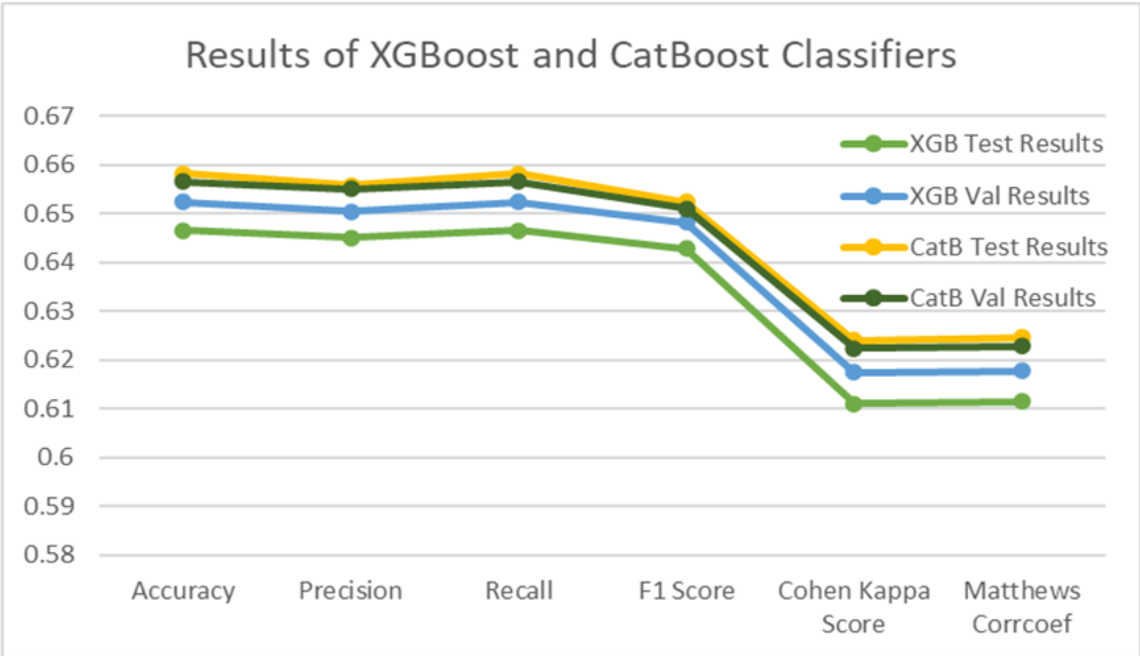

Figure 8: Results of XGBoost and CatBoost Classifiers

6.3 Experiment with Neural Networks

Convolutional Neural Network with 3 convolutional layers, 2 max pooling layers, 1 input and 1 output layer was applied on the data set. obtained from feature extraction with 5 epochs. Performance of neural network was measured using accuracy. The model gave a training accuracy of 92.7% for 5 epochs. And a test accuracy of 72.05%. Overall, CNN with 5 epochs has a better accuracy. Fig. 9 and 10 show the performance of NN in based on the testing and training accuracy respectively.

```
------------------------ Test Set Scores ------------------------
Accuracy: 0.7205
Precision: 0.7378
Recall: 0.7205
F1 Score: 0.7211
Cohen Kappa Score: 0.6933
Matthews Corrcoef: 0.6956
                Classification Report:
               precision    recall  f1-score   support

           0       0.53      0.82      0.65       293
           1       0.87      0.76      0.81        62
           2       0.89      0.82      0.85        57
           3       0.63      0.59      0.61        93
           4       0.60      0.47      0.53       174
           5       0.94      0.88      0.91       133
           6       0.62      0.72      0.67        60
           7       0.90      0.83      0.87       136
           8       0.78      0.80      0.79        35
           9       0.87      0.85      0.86       402
          10       0.85      0.73      0.79       176
          11       0.82      0.72      0.76       165
          12       0.60      0.84      0.70       103
          13       0.63      0.67      0.65       133
          14       0.56      0.50      0.52       121
          15       0.73      0.52      0.61       265

    accuracy                           0.72      2408
   macro avg       0.74      0.72      0.72      2408
weighted avg       0.74      0.72      0.72      2408
```

Figure 9: CNN Test Report

```
history = model.fit(train_ds,epochs=5,class_weight=class_weights)
```

```
WARNING:tensorflow:From C:\Users\91773\anaconda3\lib\site-packages\tensorflow\python\ops\array_ops.py:5043: calling gather (fro
m tensorflow.python.ops.array_ops) with validate_indices is deprecated and will be removed in a future version.
Instructions for updating:
The `validate_indices` argument has no effect. Indices are always validated on CPU and never validated on GPU.
Epoch 1/5
150/150 [==============================] - 299s 2s/step - loss: 2.0734 - accuracy: 0.3707
Epoch 2/5
150/150 [==============================] - 295s 2s/step - loss: 0.9841 - accuracy: 0.6799
Epoch 3/5
150/150 [==============================] - 293s 2s/step - loss: 0.5121 - accuracy: 0.8215
Epoch 4/5
150/150 [==============================] - 291s 2s/step - loss: 0.3390 - accuracy: 0.8832
Epoch 5/5
150/150 [==============================] - 293s 2s/step - loss: 0.2021 - accuracy: 0.9270
```

Figure 10: CNN Training

6.4 An extension to CNN experiment

As an extension to this research work, convolutional neural network was extended to create a Webpage where the user is able to upload the image of the diseased leaf and the application is able to classify the image and give the appropriate class of pest. For the purpose of giving an example, let us upload the image of 'cisaberoptus_kenyae' pest class on the webpage. The application was correctly able to identify the type of pest as seen in Figure 11.

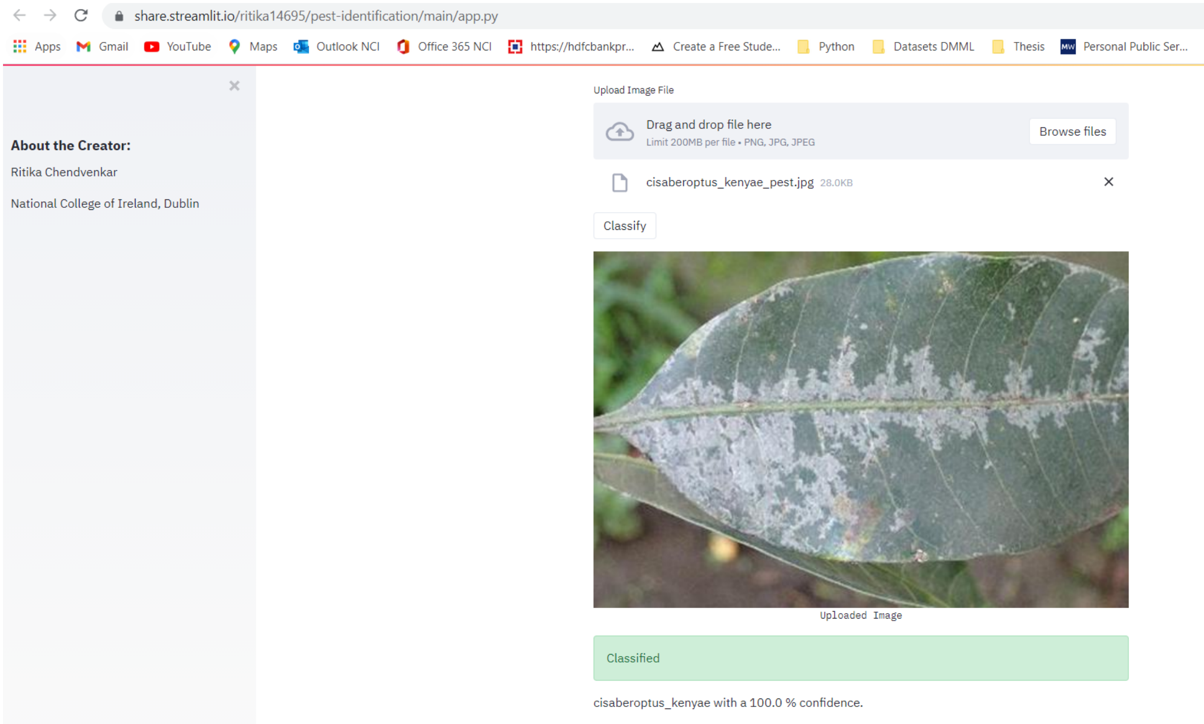

Figure 11: Streamlit app

6.5 Discussion

This research was carried out with the intent to compare traditional machine learning algorithm with the recent neural networks. Both these methods were individually implemented with various experiments. Two of the more recent approaches using decision trees were also implemented and analysed. The main challenges faced include:

1. Bulky Data: The original dataset contained 510 images. However, after augmentation, the number of images is 12000. This has resulted in increasing the number of samples in the data but has also caused the data to be heavy on the processor.

2. Computational Time: The amount of time taken to run neural networks was high. One epoch took around 9 to 10 minutes for execution. Due to this reason, the model was tested onto a maximum of 15 epochs.

Five machine learning models were implemented as a part of this research and evaluated. Different experiments were conducted to try and improve the performance of the models, including feature extraction using GLCM, feature engineering using a combination of GLCM and PCA, implementation of decision trees-based algorithms like XGBoost and CatBoost which work on the principle of boosting. Figures 12, 13 and 14 represents the performance of all models for accuracy, precision and F1-Score respectively.

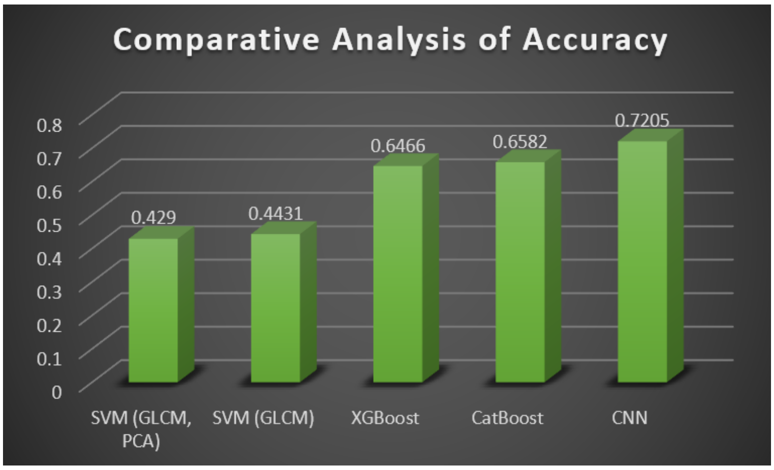

Figure 12: Comparative Analysis for Accuracy

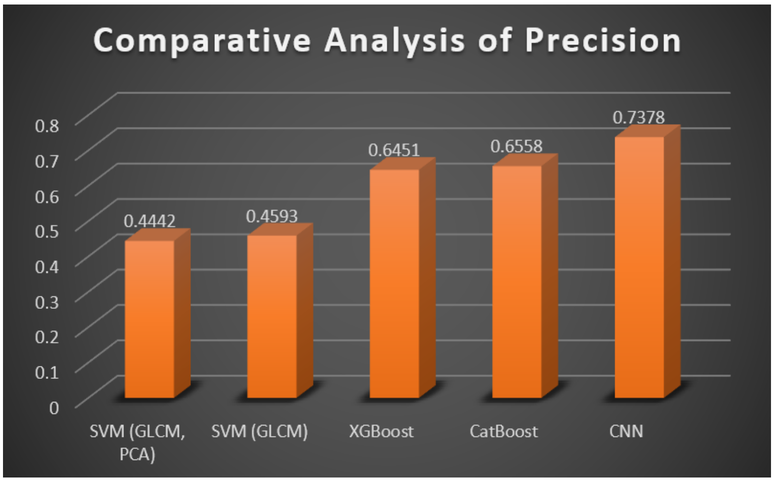

Figure 13: Comparative Analysis for Precision

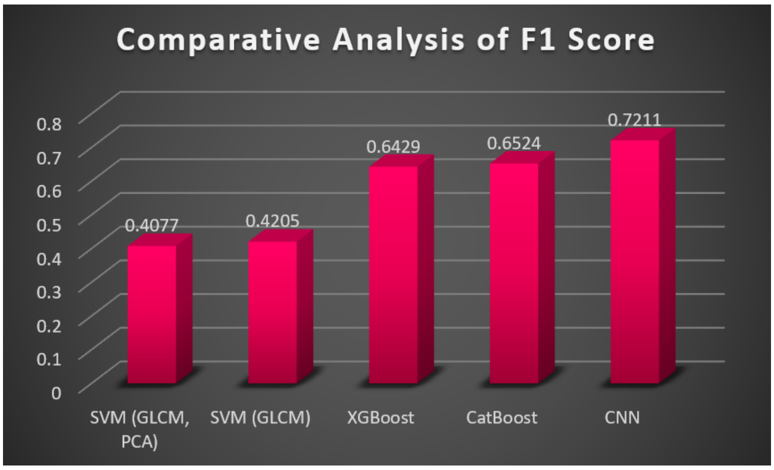

Figure 14: Comparative Analysis for F1 Score

7 Conclusion

To conclude, the implemented research work will come in handy and assist the farmers in crop management by automatically identifying the type of pest or disease and help them specifically in the area of pest management and pesticide application. Although SVM provided a very low accuracy of about 43%, exploring the RGB values could help in enhancing the accuracy. After SVM, models based on boosting were explored. XGBoost and CatBoost were implemented, and they gave a higher accuracy as compared to SVM. The accuracy of boosting algorithms was about 65%. Finally, a CNN was built from scratch, which gave an acceptable accuracy of 72.05%. This CNN was further used to build a python-based webpage where user can upload the image of the infected leaf and the class of the pest is identified. The images used in this research were the pest and diseases identified that were present in the mango farms, specifically in Indonesia.

As a part of future work, the algorithm implemented in this research can be used on realtime data from other parts of the world, in order to further carry out research on mango leaf pests and diseases. As stated above, the RGB values of the images can be explored for getting a higher accuracy. Additionally, the developed webpage could be further enhanced and considered as a base to create an android based application to facilitate portability and ease of use and make it handy.

Acknowledgement

I would like to thank my supervisor, Prof. Jorge Basilio for his constant support, guidance, motivation and assistance for all the queries in each and every one-to-one session throughout the course of this research work. To be able to present this research in an insightful manner, continuous feedback from the supervisor and my fellow batch mates was taken and worked up on. I would also like to thank National College of Ireland for making all the resources available through Moodle and Library materials. Lastly, I would like to thank my family for believing in me.

References

Alam, J., Alam, S. and Hossan, A. (2018). Multi-stage lung cancer detection and prediction using multi-class svm classifie, *2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2)*, pp. 1–4.

Betul Oktay, A. (2017). Tooth detection with convolutional neural networks, *2017 Medical Technologies National Congress (TIPTEKNO)*, pp. 1–4.

Bhimte, N. R. and Thool, V. R. (2018). Diseases detection of cotton leaf spot using image processing and svm classifier, *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 340–344.

Chouhan, S. S., Kaul, A. and Singh, U. P. (2019). A deep learning approach for the classification of diseased plant leaf images, *2019 International Conference on Communication and Electronics Systems (ICCES)*, pp. 1168–1172.

Di, P., Lv, F. and Wang, X. (2015). The research on the feature extraction of sunflower leaf rust characteristics based on color and texture feature, *2015 International Conference on Computational Intelligence and Communication Networks (CICN)*, pp. 457–460.

Fangoh, A. M. and Selim, S. (2020). Using cnn-xgboost deep networks for covid-19 detection in chest x-ray images, *2020 15th International Conference on Computer Engineering and Systems (ICCES)*, pp. 1–7.

Hameed, N., Shabut, A. M. and Hossain, M. A. (2018). Multi-class skin diseases classification using deep convolutional neural network and support vector machine, *2018 12th International Conference on Software, Knowledge, Information Management Applications (SKIMA)*, pp. 1–7.

Hong, H., Lin, J. and Huang, F. (2020). Tomato disease detection and classification by deep learning, *2020 International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, pp. 25–29.

Indrabayu, I., Arifin, N. and Areni, I. S. (2019). Strawberry ripeness classification system based on skin tone color using multi-class support vector machine, *2019 International Conference on Information and Communications Technology (ICOIACT)*, pp. 191–195.

Iqbal, Z., Khan, M. A., Sharif, M., Shah, J. H., ur Rehman, M. H. and Javed, K. (2018). An automated detection and classification of citrus plant diseases using image processing techniques: A review, *Computers and Electronics in Agriculture* **153**: 12–32.
URL: *https://www.sciencedirect.com/science/article/pii/S0168169917311870*

Kakani, V., Nguyen, V. H., Kumar, B. P., Kim, H. and Pasupuleti, V. R. (2020). A critical review on computer vision and artificial intelligence in food industry, *Journal of Agriculture and Food Research* **2**: 100033.
URL: *https://www.sciencedirect.com/science/article/pii/S2666154320300144*

Malek, M. A., Reya, S. S., Hasan, M. Z. and Hossain, S. (2021). A crop pest classification model using deep learning techniques, *2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST)*, pp. 367–371.

Massot-Campos, M., Oliver-Codina, G., Ruano-Amengual, L. and Miró-Juliá, M. (2013). Texture analysis of seabed images: Quantifying the presence of posidonia oceanica at palma bay, *2013 MTS/IEEE OCEANS - Bergen*, pp. 1–6.

Nesarajan, D., Kunalan, L., Logeswaran, M., Kasthuriarachchi, S. and Lungalage, D. (2020). Coconut disease prediction system using image processing and deep learning techniques, *2020 IEEE 4th International Conference on Image Processing, Applications and Systems (IPAS)*, pp. 212–217.

Pantazi, X., Moshou, D. and Tamouridou, A. (2019). Automated leaf disease detection in different crop species through image features analysis and one class classifiers, *Computers and Electronics in Agriculture* **156**: 96–104.
URL: *https://www.sciencedirect.com/science/article/pii/S0168169918310202*

Pham, H. N., Koay, C. Y., Chakraborty, T., Gupta, S., Tan, B. L., Wu, H., Vardhan, A., Nguyen, Q. H., Palaparthi, N. R., Nguyen, B. P. and H. Chua, M. C. (2019). Lesion segmentation and automated melanoma detection using deep convolutional neural networks and xgboost, *2019 International Conference on System Science and Engineering (ICSSE)*, pp. 142–147.

Pla, F., Carmona, P. L. and Sotoca, J. M. (2013). One-class classification techniques in image recognition problems, *2013 12th Workshop on Information Optics (WIO)*, pp. 1–3.

Priyanka and Kumar, D. (2020). Feature extraction and selection of kidney ultrasound images using glcm and pca, *Procedia Computer Science* **167**: 1722–1731. International Conference on Computational Intelligence and Data Science.
URL: *https://www.sciencedirect.com/science/article/pii/S1877050920308486*

Samat, A., Li, E., Du, P., Liu, S., Miao, Z. and Zhang, W. (2020). Catboost for rs image classification with pseudo label support from neighbor patches-based clustering, *IEEE Geoscience and Remote Sensing Letters* pp. 1–5.

Samat, A., Li, E., Du, P., Liu, S. and Xia, J. (2021). Gpu-accelerated catboost-forest for hyperspectral image classification via parallelized mrmr ensemble subspace feature selection, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **14**: 3200–3214.

Sfiligoj E. (2020). The Challenges Facing Agriculture, Going Into 2021, URL: `https://www.croplife.com/crop-inputs/the-challenges-facing-agriculture-going-into-2021/`. [Online; Accessed on: 04 April 2021].

Shijie, J., Peiyi, J., Siping, H. and s. Haibo (2017). Automatic detection of tomato diseases and pests based on leaf images, *2017 Chinese Automation Congress (CAC)*, pp. 2537–2510.

Singh, U. P., Chouhan, S. S., Jain, S. and Jain, S. (2019). Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease, *IEEE Access* **7**: 43721–43729.

Song, R., Li, T. and Wang, Y. (2020). Mammographic classification based on xgboost and dcnn with multi features, *IEEE Access* **8**: 75011–75021.

Swathika, R., Srinidhi., S., Radha, N. and Sowmya., K. (2021). Disease identification in paddy leaves using cnn based deep learning, *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, pp. 1004–1008.

Tangthaikwan, K., Keeratipranon, N. and Agsornintara, A. (2017). Multiclass support vector machine for classification spatial data from satellite image, *2017 9th International Conference on Knowledge and Smart Technology (KST)*, pp. 111–115.

Tetila, E. C., Brandoli Machado, B., Menezes, G. V., de Souza Belete, N. A., Astolfi, G. and Pistori, H. (2020). A deep-learning approach for automatic counting of soybean insect pests, *IEEE Geoscience and Remote Sensing Letters* **17**(10): 1837–1841.

Tripathi, M. K. and Maktedar, D. D. (2020). A role of computer vision in fruits and vegetables among various horticulture products of agriculture fields: A survey, *Information Processing in Agriculture* **7**(2): 183–203.
URL: *https://www.sciencedirect.com/science/article/pii/S2214317318303834*

Tropea, M. and Fedele, G. (2019). Classifiers comparison for convolutional neural networks (cnns) in image classification, *2019 IEEE/ACM 23rd International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, pp. 1–4.

Truong, Q. B., Thanh, T. K. N., Nguyen, M. T., Truong, Q. D. and Huynh, H. X. (2018). Shallow and deep learning architecture for pests identification on pomelo leaf, *2018 10th International Conference on Knowledge and Systems Engineering (KSE)*, pp. 335–340.

Tumang, G. S. (2019). Pests and diseases identification in mango using matlab, *2019 5th International Conference on Engineering, Applied Sciences and Technology (ICEAST)*, pp. 1–4.

Wongbongkotpaisan, J. and Phumeechanya, S. (2021). Plant leaf disease classification using local-based image augmentation and convolutional neural network, *2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pp. 1023–1027.