# Ai based plant disease detection and recommendation system.

# By

# Tharusha Sachindra

Submitted to

**The University of Roehampton**

In partial fulfilment of the requirements

for the degree of

**BACHELOR OF SCIENCE IN COMPUTING**

Declaration

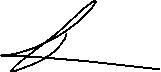
I hereby certify that this report constitutes my own work, that where the language of others is used, quotation marks so indicate, and that appropriate credit is given where I have used the language, ideas, expressions, or writings of others.

I declare that this report describes the original work that has not been previously presented for the award of any other degree of any other institution.



Tharusha Sachindra Peramunagamage Don

Date: 14/05/2025



Acknowledgements

I would like to thank everyone especially by family, friends and lectures for supporting for my journey.

My heartfelt gratitude goes to my aunty and uncle for helping me and guiding me through the journey.

A special thanks goes to Dr. Ogechukwu Okonor for advising me from the very beginning of journey in the university.

My supervisor, Mohammad F. Khan and my second marker Fakherldin Saeed for grading me.

Abstract

Crop diseases are a major threat to the economies worldwide. Detecting the diseases in the plants traditionally took time, labour and expertise but with the rise of the new technology such as machine learning, deep learning and artificial intelligence has made this task simpler and efficient with less of human intervention.

The aim of this project was to provide a system with the ability to classify the diseases in potato and wheat plants and provide a recommendation for the users as to what must be done. A webapp was developed to provide the recommendations for diseases that were identified by the system after the users upload a photo of the diseased leaf. Additionally, a robotic car was also created for further enhancing the use of the models.

In this project, a Convolutional Neural Network is created for the detection of diseases. This system helps the farmers to identify the diseases in the plants and follow the recommendations of that is provided from the webapp. The wheat model provides an accuracy of 0.80 and the potato model provides an accuracy of 0.86 and the robotic car can drive with the user’s command that is provided from the keyboard and take the picture, analyse and provide the disease name.

Keywords- *Plant disease detection, Machine Learning, Deep Learning, Robotics, Convolutional Neural Networks.*

Contents

[Introduction 7](#_Toc198122409)

[Problem description, context and motivation 7](#_Toc198122410)

[Aims 8](#_Toc198122411)

[Objectives 8](#_Toc198122412)

[Legal 8](#_Toc198122413)

[Social 8](#_Toc198122414)

[Ethical 9](#_Toc198122415)

[Professional 9](#_Toc198122416)

[Background 9](#_Toc198122417)

[Project overview 9](#_Toc198122418)

[Literature review 10](#_Toc198122419)

[Technology review 13](#_Toc198122420)

[Summary of outcomes of literature review 13](#_Toc198122421)

[Summary of outcomes of the technological review 14](#_Toc198122422)

[Methodology 15](#_Toc198122423)

[Design 16](#_Toc198122424)

[Testing and evaluation 17](#_Toc198122425)

[Project management 18](#_Toc198122426)

[Technologies and processes 18](#_Toc198122427)

[Implementation 18](#_Toc198122428)

[Wheat model development 19](#_Toc198122429)

[Observations and implementations for the wheat model 24](#_Toc198122430)

[potato model development 25](#_Toc198122431)

[Observations and implementations for the potato model 27](#_Toc198122432)

[Results & Evaluation 28](#_Toc198122433)

[Wheat model 28](#_Toc198122434)

[Potato model 33](#_Toc198122435)

[Web app 38](#_Toc198122436)

[Robotic Car 39](#_Toc198122437)

[Related work 40](#_Toc198122438)

[Conclusion 41](#_Toc198122439)

[Future work 42](#_Toc198122440)

[Reflection 43](#_Toc198122441)

[References 44](#_Toc198122442)

[Appendices 47](#_Toc198122443)

[Appendix A: project proposal 47](#_Toc198122444)

[Appendix B: evidence of project management. 49](#_Toc198122450)

[Appendix C: Additional Technical details 49](#_Toc198122451)

[Appendix D: Supervisor meetings. 49](#_Toc198122452)

# Introduction

Crops provide us with food that is used as a staple food all over the world. However, managing the plant diseases can be a very hard topic especially if the farmer Is new to whole industry and do not know what must be done to prevent and control the diseases. Due to this problem with the proper disease identification a solution can be used. The use of the deep learning where a model is trained to find out the signs of diseases to prevent the diseases. This report presents an analysis of the plant disease detection methods using deep learning that can be used and evaluates their effectiveness solving the problems faced by farmers.

## Problem description, context and motivation

Plant diseases can cause damages which can be irreversible if they are not identified at the right moment. Traditional methods such as relying on the human vision could be Too late for working out a solution as there are time consuming and the human eyes may not notice the subtle notes of the early stages of the diseases to prevent these.

If the diseases are not identified and treated there can be many stakeholders that can get effected. The main group includes the farmers as they will lose what they have invested to the crop because of this loss consumers will have to pay higher prices which can put strain on the governments as they will have to arrange solutions for these problems. Moreover, other parties such as food processors may also have to face the ripple effects of these problems.

The problem of plant diseases is common occurance when it comes to crops and can happen anywhere in the world and can depend on the weather, plant type, and how much the area is affected by plant diseases.

Solving this problem can ensure that the plants get the necessary treatments at the necessary times to ensure that the farmers get the full yield of their crops and to ensure the food security. Moreover, building a system that can identify the diseases can help new farmers who are entering the industry to correctly identify diseases and use the appropriate amounts of pesticides which is good for both the consumers and the farmers too.

## Aims

The primary aim of this project is to develop an accurate and efficient system for classifying the plant diseases especially in potato and wheat plants as these are the most common types of crops that are in the United Kingdom. Furthermore, this system also provides the recommendations for the users on what must be done to overcome the situation. Moreover, the project aims to make a web application for the users to upload the photos and get a classification of the diseases and get a recommendation for the diseases that is model is trained on.

## Objectives

The main objective of this project is to make a web application for identifying plant diseases in wheat and potato and provide recommendations to the users. To achieve the main objective the project needs to collect the necessary datasets that is needed to train the models which is intended to be done through the free to access websites such as Kaggle and free to use to use datasets. Furthermore, after collecting the datasets the training of Convolutional Neural Network (CNN) and testing the models for their accuracy. Additionally, after all the objectives above has been achieved the recommendation system will be built to provide the necessary recommendations for the user.

## Legal

From the legal standpoint data protection laws must be taken into consideration such as collecting the data from the users and other free to use websites. Intellectual property rights must also be taken into consideration such as when using open-source tools and algorithms.

## Social

This project aims to close the gap between farmers and the latest technology that is available by making detection systems more efficient and affordable in the ways of correctly identifying the diseases and providing the necessary solutions for the farmers before it is too late for something can be done or vast amounts of pesticides have to be used in order to eradicate the diseases which can be harmful for everyone that is involved. This system also helps new farmers as they may lack the experience that is needed for the detection of the diseases.

## Ethical

When dealing with the ethical concerns of this project the datasets the researcher collect must ensure that it contains equal amount of information in all the classes that they want to find out otherwise, the system may have bias introduced into the dataset which can lead to improper classification of the diseases in the trained models. Additionally, this project uses images of plants which must be collected from a human controlled environment which means that we must take the legal consent form the farmers before moving forward.

## Professional

From the professional perspective, this project aims to keep record of what has been done in the forms of flowcharts, diagrams and other suitable formats to achieve the same results if this was to replicate by another party. This project also aims to connect with the other professionals in the field to get the necessary recommendations for the purpose of providing a proper outcome for the users.

## Background

This problem of plant diseases is a challenge that has been there since the start of agriculture which even have led countries to shortages of food supplies in the past. In fact, [1, pp. 837–842] shows that 40% of the damage is caused by the plant diseases which can cost up to US$220 billion worldwide. However, with the evolving technology concepts such as using deep learning is getting widely popular which can help to classify the diseases and provide the necessary steps for solving the problem where it is impossible for humans to detect the diseases until the diseases are spread to a wider area of the crop.

## Project overview

This report is organized as follows, the introduction will give the basic outline of the problem, the aims it tries to achieve and objectives of this project. The document also highlights the legal social, ethical and professional aspects of the project and finally the background. The literature review of this project will highlight the key research that has been done in the area and the existing methods for plant disease detection. The technology review of this project will also highlight what technologies are being used to achieve this project. The methodology section will help the readers understand how the technology that is talked about in the technology review is applied.

# Literature review

Plant diseases pose a critical risk for the plant health which can reduce or destroy the crop output which may also have effects on other sectors which depend on the agriculture. Early detection of the plant diseases is the key for reducing the impact of plant diseases this will ensure that the farmers and other stakeholders can necessarily actions to mitigate the problems that are faced by these.

Before the introduction of machine learning and deep learning farmers and other stakeholders used to visual inspection, which is simple, yet time consuming process given the size of the area they may decide to investigate before giving a final wording of the disease/s. Furthermore, the decisions that is taken by stakeholders can be different from one to another which can lead to multiple types of unwanted treatments being applied on the crop which can increase the cost for farmers and unwanted use of pesticides can be harmful.[2, pp. 569–596] shows that use of recommended amount pesticides can alter the natural soil properties and harm good soil microorganisms which can lead to reduced soil fertility and productivity. Furthermore, [3] shows that the use of pesticides has linked to causing chronic kidney diseases in humans.

Introduction of machine learning, deep learning and computer vision the detection of plant diseases completely changed. Before the introduction of deep learning, which is a subset of machine learning was the one that was used to do disease identification, but deep learning dramatically increased the accuracy and the effectiveness of plant disease detection [4, pp. 468-].

When it comes to data collection which is considered as a main part of the system. Some researchers tend to collect the data from the real-world scenarios while some researchers tend to grow the plants by themselves and inoculate them with the virus [5, pp. 56683–56698]. However, the images that has been collected from the real world may have noise in them. According to [6, p. 1177225] noise can decrease the model accuracy by 5% to 40% in tasks such as classification. Additionally, the model even achieves a significant accuracy just by looking at the background of the image which can indicate the reliance on the background pixels rather than the actual disease features[7]. However, to overcome the problem of fitting to the noise different pre-processing techniques can be considered such as image clipping where the cropping to get the interested image region after that image smoothing can be done to smooth out the picture and image enhancement which is done by increasing the contrast [8, pp. 768–771].

The accuracy of the plant disease classification mainly depends on how much of data that can be captured from the plant leaves. [9, pp. 229–242] researchers have used multiple types of imaging techniques such as hyper spectral imaging where they identify the plant stress due to the fungi in the oilseed rape. Also, thermal imaging was used for tea leaves and can be used for the potato and wheat plants. According to researchers the by using the techniques they have been able to identify the early and late blight in potatoes and fungal diseases in wheat. The researchers also point out the use of the chlorophyll fluorescence imaging for identification of the photosynthetic performance of plants.

The other problem that is highlighted by the other researchers is that the unavailability of enough datasets to train the models. Due to that the researchers have used data augmentation, where it increases the training data which can help to reduce the problem of overfitting and improving generalization to unseen data[10, pp. 913–920]. For the plant disease detection [5, pp. 56683–56698] these researchers’ purpose two methods. First method is the use of traditional augmentation where they purpose methods such as the use of tensile rotation, adjustment resolution image translation distribution balance the repercussion of using this method includes poor quality, inadequate diversity and unevenness. The second method they purpose Is the use of Generate Adversarial Networks (GANS), a method that has been introduced by [11, pp. 139–144] it is an artificial intelligence algorithm that solve the generative modelling problem by studying the training examples. The main reason this method is used is because to generate synthetic samples with the same characteristics as the given training distribution. This method Is being used extensively to generate samples and many other variations based on the original GANS has been published such as DCGAN (Deep Convolutional Generative Adversarial Network), CGAN (Conditional Generative Adversarial Network).

Collection and pre-processing of the images is an important process. However, it is also important to find out what kind of model or how to make a model that can correctly classify the pictures of the diseased plants. A common approach Is to create a Convolutional Neural Network which can predict the correct plant disease with an accuracy of 80% or higher with the right parameters [12, pp. 109–113]. The advantages of using CNNs include the automatic feature learning, reduced computational cost. However, [13, pp. 64–68] shows that due to the unavailability of large, labelled datasets and poor-quality images or datasets with limited variations can make the model to generalize too poorly.

Due to the large amounts of data needed to train CNNs [5, pp. 56683–56698] researchers suggest using transfer learning, where it enables the use of other pre-trained CNNs such as AlexNet, VGG, ResNet[4, pp. 468-] Transfer learning makes the adaptation of these pre-trained models by retraining them with smaller datasets rather than using that data for training a network from scratch. Using transfer learning has some significant advantages over training a network from the scratch such as transfer learning can significantly improve the detection accuracy and reduce the false positive rate and they generalize well too and can achieve a higher validation accuracy, somewhere around 90% to 95%[14, p. 105393].

When all these materials are considered transfer learning seems to be the most appropriate choice for detecting plant diseases as they pre-trained with other similar images such as AlexNet which has 60 million parameters and VGG which has 133 million to 144 million parameters[4, pp. 468-].

# Technology review

From the literature review it was understood that the researchers mainly concrete on the dataset that contains pictures, pre-processing and selection of the model. When it comes to the dataset this project uses images that has been available for the public for free to use through Kaggle. The benefits of using the dataset from the Kaggle is that the data is already labelled, which is a tedious task to do manually. However, the data in the dataset can be of lesser quality which could be harder to use as they may require pre-processing. Also, the classes that are available in the dataset

Selection of a proper way to train a model is also important as the data collection process. For this project there were two methods that was considered a CNN model and a transfer learning model. For this project a CNN model and transfer learning model will both be tested as there can be benefits and drawbacks of both systems such as CNNs architecture can be designed according to the needs of the model trainers. However, pre-trained models can be better option as it requires a small number of datasets[15, p. 5930].

Creation of the CNNs is done using the help of the libraries such as TensorFlow, keras, matplotlib for the charts.

## Summary of outcomes of literature review

|  |  |  |
| --- | --- | --- |
| Aspect | Benefits | Limitations |
| Traditional methods | Simplicity, low cost. | |  | | --- | | Subjective, limited scalability. | |
| Imaging Technologies | High accuracy in early detection. | Expensive, requires expertise. |
| AI Models | Accurate, scalable for different crops. | High computational needs, reliance on labelled datasets. |
| IoT Integration(purposed) | Enables real-time monitoring. | Connectivity issues, device reliability. |
| Recommendation Systems | |  |  | | --- | --- | |  | Provides actionable advice, integrates with mobile apps. | | Limited customization for diverse crops and diseases. |

*Table 01: summary of Outcomes*

## Summary of outcomes of the technological review

|  |  |  |
| --- | --- | --- |
| Tool | Benefits | Limitations |
| TensorFlow | good library for building, training, and deploying CNN models. | Could be hard for beginners to implement. |
| Pytorch | Flexibility in model design and dynamic computation graphs. | less user-friendly for deployment compared to TensorFlow. |
| Keras | Simplified API for rapid prototyping. | Limited customization compared to TensorFlow or PyTorch. |
| OpenCV | Preprocessing of images and feature extraction for CNNs. | Not a complete deep learning framework, requires integration with others. |
| Google colab | GPU support or training CNNs. | Limited resources and slower training for large-scale models. |

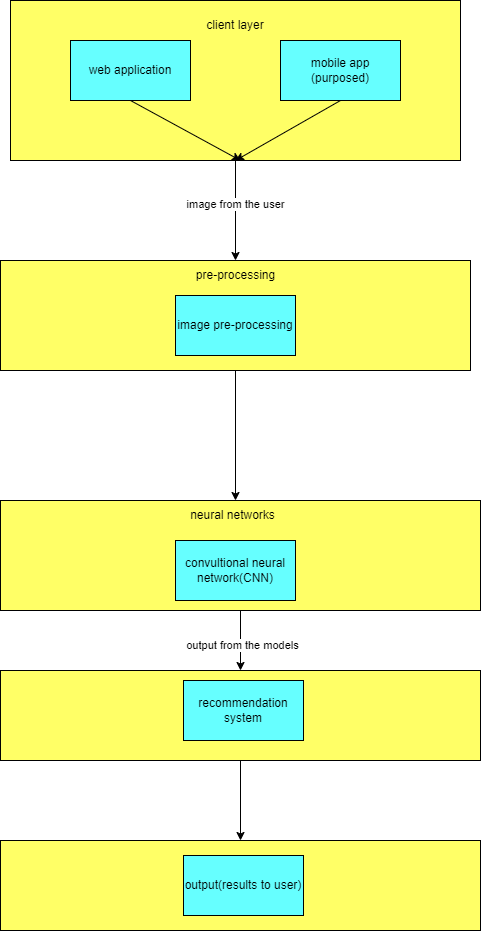
*Table 02: summary of Outcomes of the Technological review.*

# Methodology

The project concentrates on two types of plants, wheat and potato which are the most grown crops in the UK. The datasets are taken from free-to-use websites such as Kaggle. Furthermore, after obtaining the datasets from Kaggle the data must be pre-processed for any inconsistencies such as imbalanced datasets where the data must be filled by synthetic forms.

After the research, selection and pre-processing process the data can be used to train a model, which can be a CNN model or a transfer learning method. For this project both CNN and Transfer learning model is used to achieve these results. The reason for selecting both types of models is because the advantages each of the models give such as CNNs can be customized in the researchers want the results [16, p. 106125]. The benefit of using the transfer learning methods is that the models are pre-trained and only must be tuned according to the user’s preference. However, this method may not always be suitable for the problem due to that techniques such as custom deep learning models will be trained and hyper parameter tuning will be done until the validation accuracy comes to acceptable rate.

## Design



*Figure 01: system architecture diagram*

The figure 1 shows the system architecture diagram. The diagram shows how the webapp will be working. After the photo has been uploaded by the user it will be pre-processed according to the way the pictures were uploaded to the model, so it matches the model’s expected criteria. After going through the pre-processing process, the uploaded pictures will go through the CNN model that has been trained using the selected datasets which it will produce the most confident class that matches the output and based on that the web app will produce the recommendations for the user as what must needs to be done.

A diagram of a plant disease detection system

Description automatically generated

*Figure 02: use case diagram.*

The figure 02 shows the use case diagram for the web application. This shows how the users will be interacting with the final system and using the system functionalities. In this case the user uploads the picture and can identify the diseases and get the recommendations and the robot(purposed) can also do the same. However, only the admin has the privilege of updating the web app.

## Testing and evaluation

For testing the webapp the researcher can upload a picture of a diseased wheat or potato leaf and check if the model is making the right output, the plant disease and the recommendation. Evaluation of the model can be done with the validation accuracy of the model.

After the full completion of the project the webapp will be tested in the real-life conditions for the accuracy of the models. User feedback will also be used to access the quality of the output and make further improvements to the system.

## Project management

This project will follow the agile methodology which makes sure the flexibility and iterative development. The timeline of the project is given below:

Phase 01: Data collection, pre-processing and planning (1-2 weeks).

Phase 02: model development and training (1.5 months).

Phase 03: recommendation system development (1 month).

Phase 04: system integration and deployment (1 month).

Phase 05: evaluation, documentation and final presentation (1 month).

For the project management task, a kanban board is used for tracking the progress of the project which is provided through the GitHub and git through the GitHub is used for the version control.

## Technologies and processes

for the creation of the model the several technologies are used such as keras, TensorFlow as the libraries and the python as the main coding platform. CNNs are used for the training of the model at the start and the use of transfer learning will also be tested for achieving a better model performance. At the end the best performing models will be used for the moving forward.

# Implementation

After having a discussion with the supervisor, it was decided that it will be better that concentrating on only the crops that are mostly grown in the UK to be used due to the easily testable and can be deployed. As it was suggested in the project proposal the data was going to be collected from the real world it could not be done due to time constraints. Instead, datasets from the Kaggle were used to train the dataset. The link for the [wheat dataset](https://www.kaggle.com/datasets/kushagra3204/wheat-plant-diseases/data) and this is the link for the [potato dataset](https://www.kaggle.com/datasets/warcoder/potato-leaf-disease-dataset/data).

For the task of detecting the diseases in potato and wheat two models were developed to classify the diseases properly. Methods such as transfer learning was used to create architectures for the models. However, transfer learning did not perform as expected. Due to that a Convolutional Neural Network was created. After creating and testing several models one model was chosen based on the validation accuracy and the validation loss of the model. All the models that were used for this project was trained on laptop with intel core i7 12th gen processor and no Graphical Processing Unit.

## Wheat model development

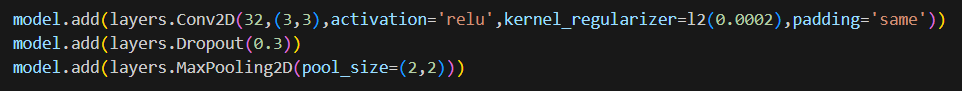
After selecting the dataset several iterations had to be tested to find the best model in terms of the validation accuracy of the model. Several models were tested with layers deep as 2-5 and filters per layer ranging from 8-512. As the depth and the complexity was increased the time that was taken to train the model also grew exponentially. However, as bigger models take time and compute resources it was not feasible to train models with larger filters and depth.

A graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of

AI-generated content may be incorrect.

*Figure 03: model training results for the first model.*

The figure 03 shows the accuracies and the losses for a model that had a structure of 32 and 64 filters accordingly with 5 layers deep. This model was trained only for 20 epochs. The model learns well with the training data, but the validation accuracy fluctuates and lags which suggests the model is overfitting where the model memorizes the training data but struggles to generalize to unseen data. Based on this model this model performance actions that were taken includes regularization, in this case L2 regularization method was used where the sum of squared values of the model’s coefficients is added. Furthermore, dropout feature was used in different layers of the model. What dropout feature will do is randomly sets zero to fraction of the neurons in a layer on each of the passes where it helps the model to generalize better to the unseen images. The figure 04 shows how the dropout and L2 regularization is applied in the code.



*Figure 04: L2 regularization and padding in the code.*

.A graph of different colored lines

AI-generated content may be incorrect.

*Figure 04: model training for the second model.*

This figure 04 shows the model that was trained with 64 and 128 filters accordingly and 5 layers deep and no other changes have been made to the model from above. This model was only trained only for 10 epochs to check the performance where if the performance is acceptable the number of epochs will be increased and will be trained. However, such models with higher number of filters in the model the time that it takes to trains the model will also increases exponentially that is another reason why this model was trained with a lower number of epochs in the testing phase.

A graph of a line and a line

AI-generated content may be incorrect.

*Figure 05: model training for the fourth model.*

The figure 05 shows another iteration of the model where the L2 regularization was introduced to the model. Consistent with the previous findings the model learns well with the training data but fails to generalize to the unseen data making the model overfitting. Furthermore, the validation loss does range from 2.4 to 2.6 during the period of training. With this finding the model’s penalty was increased from 0.0001 to 0.0002.

A graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of

AI-generated content may be incorrect.

*Figure 06: model training for the fifth model.*

This shows another model that was trained with only changes to the batch size from 32 to 15 and everything else was kept the same for the training period. With this model it was understood that the model needed more hyperparameter adjustments.

A graph of a number of lines

AI-generated content may be incorrect.

*Figure 07: model training for the sixth model.*

The above model is an example of a transfer learning model that was used. The transfer learning model that was used here is the ResNet50 model which is 50 layers deep which is pretrained on 1.2 million images which belongs to 1000 classes. Same parameters that were used in the previous models were kept for this model. However, for this case the transfer learning technique does not fit. Due to that the custom CNN model was preferred.

A graph of different colored lines

AI-generated content may be incorrect.

*Figure 08: model training for the sixth model.*

From all the data that was collected from training models with different parameters the model results shown in the figure 09 had one of most tuned parameters. The model was trained with L2 regularization = 0.0001 and batch size was adjusted from 32 to 15, this means the model will update the weights more regularly which can help the model to generalize better. After training the model it was evident that the model is generalizing well compared to the other models so far.

A graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of

AI-generated content may be incorrect.

*Figure 09: model training for the third model.*

The figure 06 was 3 layers deep and comprised of layers of 32,64,32 filters accordingly and a dense layer of 64 neurons. Upon testing the models with the new hyper parameters, the model showed some progress than the other tested models and decided to train for a longer period hence, 50 epochs were used to train the model that took around 5-6 hours to complete the training the model. From this model it was understood the general settings for the model was fine and hyper parameters were the ones that had to be further tuned to achieve the best model performance.

## Observations and implementations for the wheat model

after training nine models for the wheat disease identification with changes to the layers in a network to the filters in the network. From the observations L2 regularization was used to mitigate the problem of over fitting the model where the model memorizes the training images rather than learning the features of the images which leads to poor validation accuracy and a higher validation loss. Use of padding seen as another reason for the increase in the validation accuracy of the final model. Learning rate adjuster was also used for the final model with it set to 0.0002. With the mentioned hyperparameters the model was trained for the final model.

## potato model development

After selecting the dataset from Kaggle a custom model was made and tested with the lessons that was learned from the previous model trainings that was done on the wheat model due to the experiences that was gained from training the previous model.

A graph of different colored lines

AI-generated content may be incorrect.

*Figure 10: The first potato disease detection model.*

The figure 10 shows the first model that was trained for the potato model. This model has an architecture 8,16,32, and 64 filters and comprised of 4 layers. No regularization was used in this model and the batch size is 32 which is the default batch size of the framework, keras. The model’s training accuracy increases steadily while the validation accuracy plateaus around 0.8 and fluctuates. The training loss decreases steadily while the validation accuracy plateaus in the 20th epoch which then starts to fluctuate. This means the model starts to memorize the training data rather than generalizing.

A graph of a graph of a graph

AI-generated content may be incorrect.

*Figure 11: The second potato disease detection model.*

This is another model that was trained with the filters of 8,16,32 and 3 layers deep. The model was trained for 50 epochs. The model learns well form the training data however, the model does not generalize well to the validation data very well and the model does fluctuate. The best validation accuracy the model gains is around 0.4 and the validation loss goes to a minimum of around 1.2 to 1.3.

A graph of different colored lines

AI-generated content may be incorrect.

Figure 12: The third potato disease detection model.

This shows a model that has a structure 64 and 128 filters stacked accordingly in a 6-layer deep network which is then fed through 64 neurons fully connected dense layer and the default batch size was used,32. The model was trained for 100 epochs and after 60 epochs the validation accuracy of the model starts to fluctuate more and the difference between the accuracies and the loss starts to widen and plateau where the validation accuracy is at around 0.78 to 0.79.

## Observations and implementations for the potato model

The potato model was the one with the lowest number of iterations that was used which was three due to the information that was learned from the training the wheat model. Moreover, compared to the wheat model finding a better fitting model was much easier. The model only needed filters of 64 and 128 filters and the network was 6 layers deep and padding was also used in conjunction with batch normalization where it optimizes the learning of the model. For a perfect fitting model. However, learning rate adjuster was also used in this model where it had a patience of only 5 epochs where if the validation loss does not go down it will automatically reduce the rate by 0.5. after adjusting these parameters, the final model was trained.

# Results & Evaluation

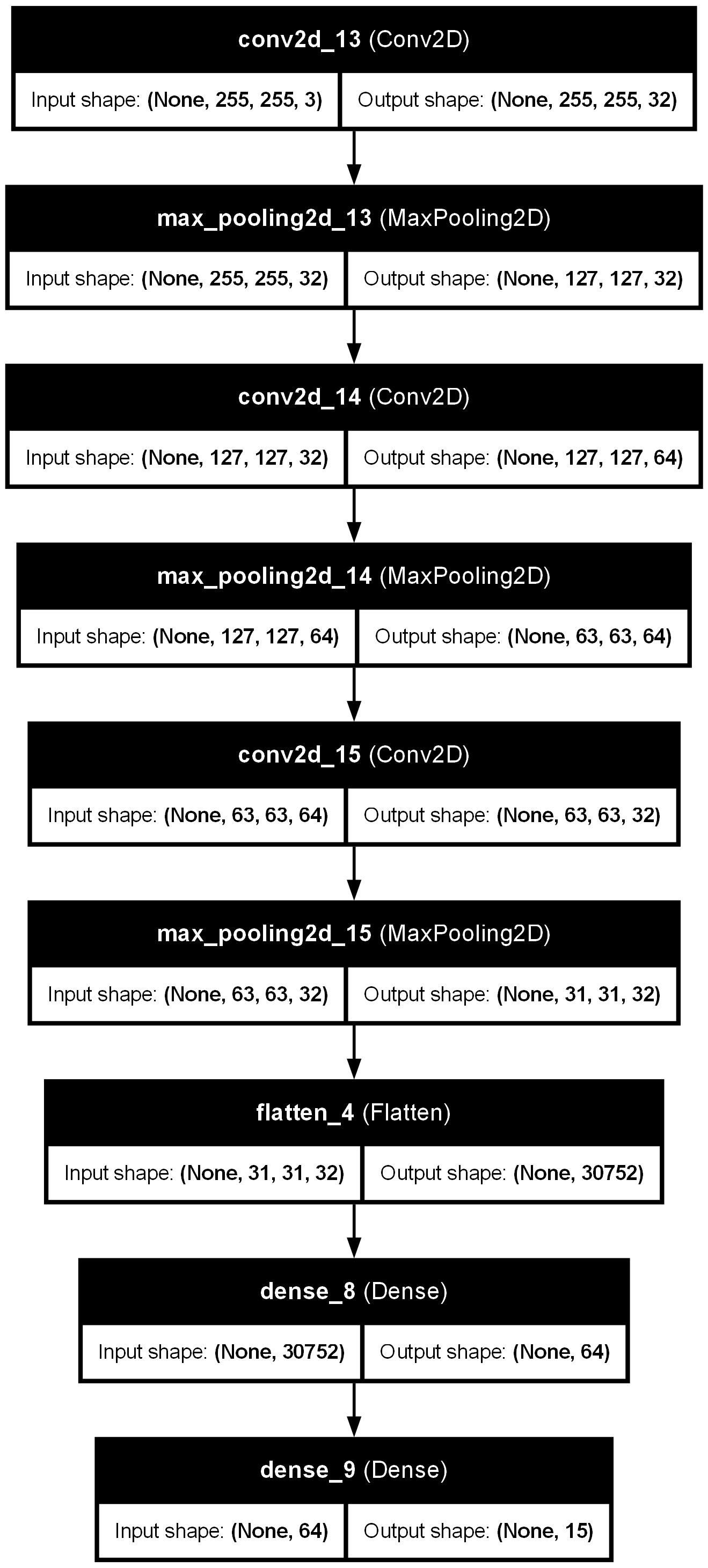
## Wheat model

Before the images are fed to the Convolutional Neural Network (CNN) they had to be pre-processed after the images was pre-processed the images were augmented, where this process synthetically increases the size of the dataset and even if an image is given rotated the model will be able to easily identify the diseases. The figure 13 shows the image pre-processes and the data augmentation that was done on the images. These augmentations ensure that whatever the setting the picture was taken it can give a proper output for the input. After that the all the images will flow through the ‘train\_datagener’ from the given image location. The batch size is adjusted due to the less CPU memory usage and larger batch sizes mean that the model will train faster. However, different batch sizes were also tested such 15,25 but did not show any significant improvements in the accuracy of the model.



*Figure 13: Data augmentation and image pre-processing done on the wheat images.*

The Convolutional Neural Network (CNN) that was used for the final model includes 4 layers with 32 filters in each layer and batch normalization has been used in the first and second layers of the model and dropout has been used in the third and the fourth layers of the model. At last, the output from the top layer will go through the dense layer with 64 neurons where the collected features are condensed and analysed, and the final dense layer will decide the disease with the most confidence. The below image shows the model architecture that was used for the final project.



*Figure 14: The final wheat disease detection model architecture.*

During the training process the above model showed some levels of over-fitting, to mitigate the over-fitting model was introduced with L2 regularization where a penalty is added that is proportional to the squared values of the weights, for the model that was used here the L2 regularization=0.0002 was used[17, p. 107]. Moreover, padding was also used in the model. The use of padding is to maintain the size of the image as the output size shrinks after each convolutional operation and to avoid losing the edge information because the convolution mainly focuses on the centre of the image and by using padding the boarder pixels will also be considered[17, p. 107].

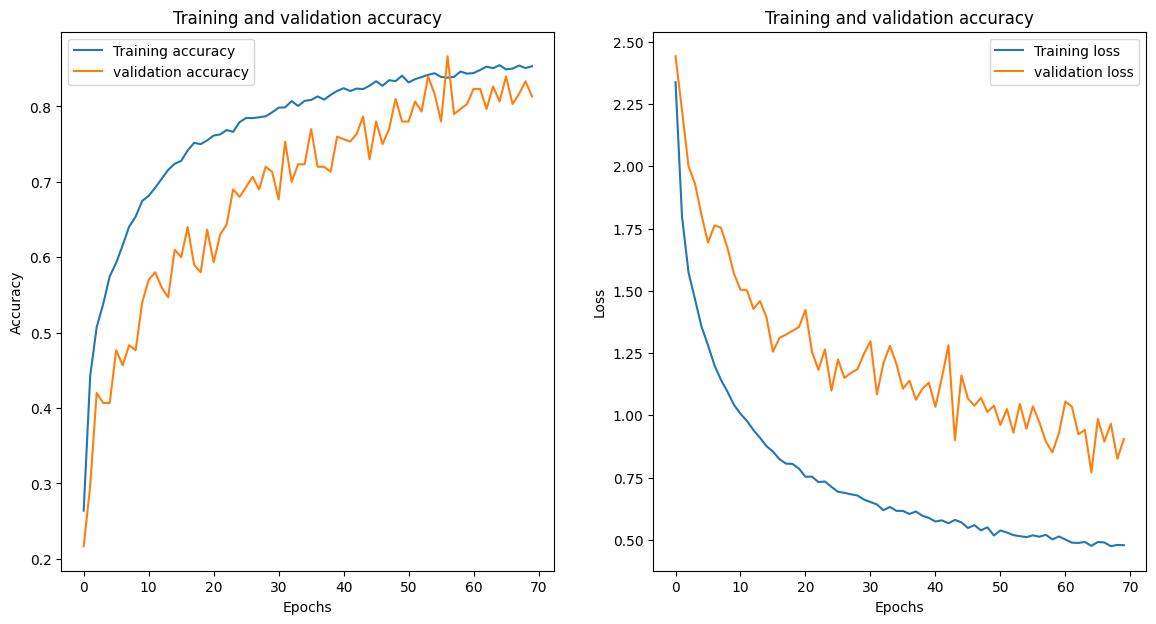
For training of the model Adam (Adaptive Moment Estimation) was used as it is one of the most used and effective optimization algorithms. To evaluate the model the accuracy of the model was used. To improve the accuracy methods such as learning rate schedular was used where it helps to improve the accuracy of the model, helps with unstable learning where the training loss is bouncing. There are many types of scheduling strategies that are available however, for this project ‘ReduceLROnPlateau’ was used where the learning rate is reduced if the validation loss does not improve in this case it is set to 4 epochs and then the learning rate will be reduced by a factor of 0.5. After setting all the necessary settings the model is trained for 70 epochs.

A collage of images of plants

AI-generated content may be incorrect.

*Figure 15: the activations of the model through each layer.*

The figure 15 shows the how the filter in each layer extracts the information from the image. The pixels that are coloured in yellow are the parts the model thinks that it thinks important, and it is extracted. Moreover, Layer 1 extracts the basic textures such as leaf veins, layer 2-3 extracts more defined patterns and structures and the layer 4-6 will extract even more abstract features from the image that has been passed from the top layer and then passed on to the dense layers in the model where all the data that is collected is used to make the prediction.



*Figure 16: validation, training accuracy and the validation, training loss for the wheat model.*

The above graph shows the accuracies and losses for the wheat model. This model has been able to achieve 0.86 validation accuracy over a 70-epoch period and the validation accuracy seems to be stabilizing around at 0.80 but the validation loss keeps fluctuating which could mean that the model is showing over-fitting

A screenshot of a computer

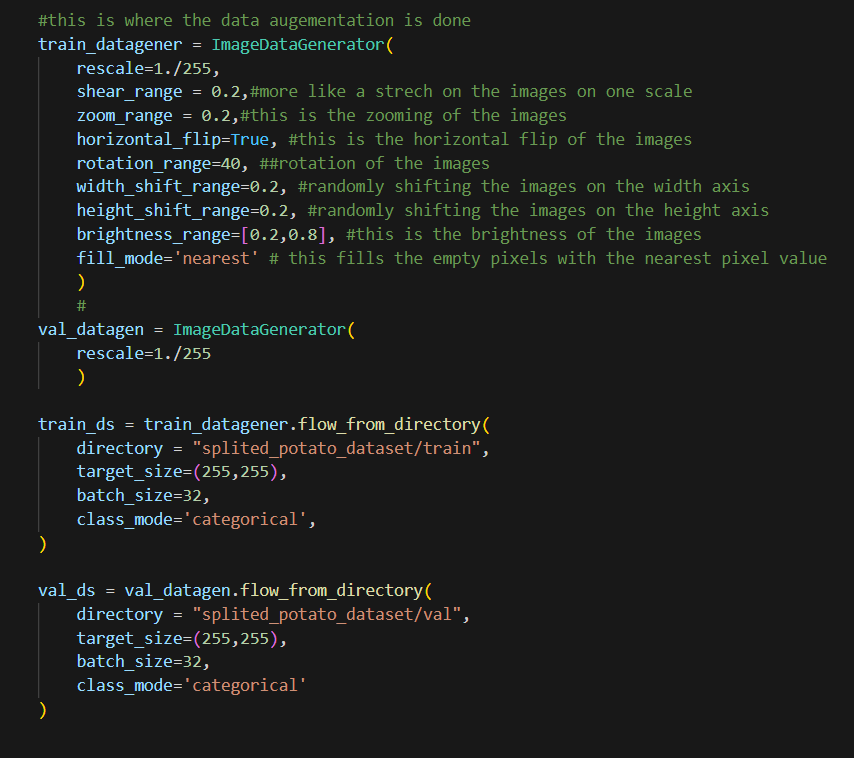
AI-generated content may be incorrect.

*Figure 17: confusion matrix for the wheat model.*

The confusion matrix above shows the true positive rate of the model where the best performers include the smut with a perfect classification and the stem fly and septoria being the other best performers. However, this model has weak performers too, highest false positive rate which are yellow rust and tan spot where 40 pictures are classified as healthy but are yellow rust.

## Potato model

Such as the previous model that was described above, we need to pre-process and perform data augmentation on the dataset that we are using. All the images are rescaled first so if they have different sizes that vary everything will become the same size. After rescaling the images, the data augmentation is applied. For this model the several data augmentation techniques are applied such as shear range, zoom range horizontal flip, rotation flip. All the types of techniques that are used is showed in the figure below.



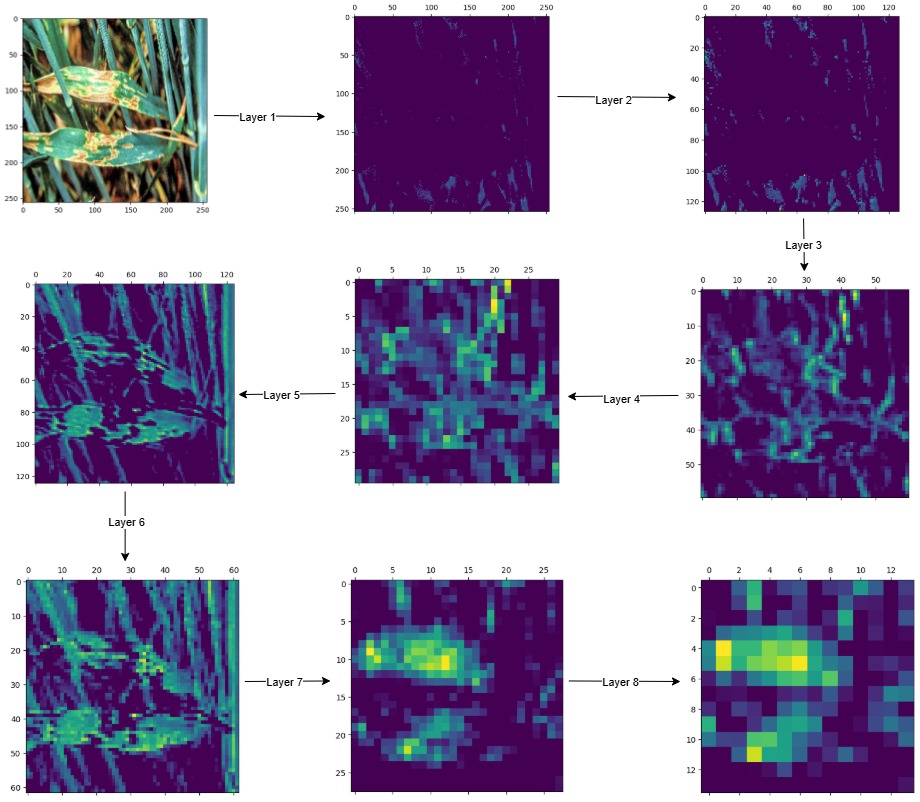
*Figure 18: Data augmentation and image pre-processing done on the potato images.*

Detection of diseases was also done using the custom-made CNN network. The network that was used for the detection of diseases in potato leaves has 6 layers comprised of layers with 128 and 64 one after the other. Batch normalization was also used in each of the layers to make the training much faster and more stable by normalizing the activations of the layers. The below image shows the model architecture that was used for the potato disease detection model.



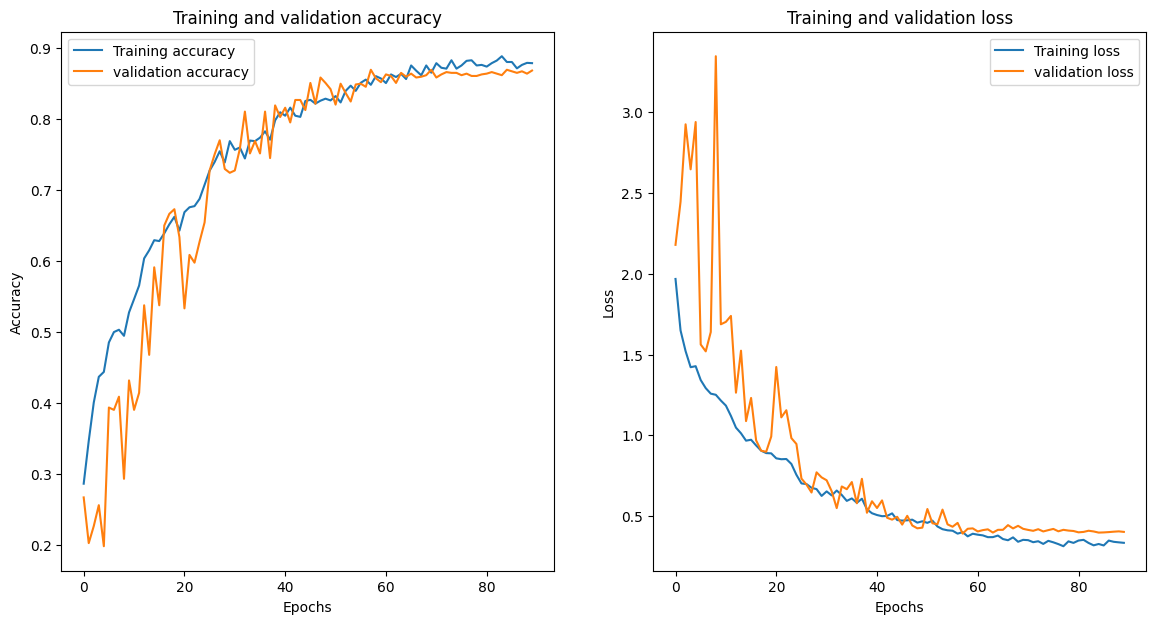
*Figure 19: The final wheat disease detection model architecture.*

This model also uses ‘adam’ as the optimizer and the metric that is used for the assessing the model is the accuracy. For this model learning rate adjuster was used with the patience = 5 which means that if the validation loss does not improve after 5 epochs the learning rate will be adjusted by a factor of 0.5. After setting all the settings the model is trained for 50 epochs.



*Figure 20: the activations of the model through each layer in the potato model.*

The figure 20 shows the how the information is extracted through each layer of the potato model. The first two layers of the potato model does not absorb any information but finds the background noise. By time it comes to the layer 5 and 6 the filters identify the diseased areas and extracts the information and at the last layers the picture is more zooned in and more abstract information is extracted from the images.



*Figure 21: validation, training accuracy and the validation, training loss for the potato model.*

The figure 21 shows the final model that was used for the final use in the webapp. This model was trained for 90 epochs with the batch size of 25 and the optimizer that was used was adam. The model initially shows some fluctuations in early periods of the learning curve however, towards the end of the training of the model it can be observed that the model is fitting well which suggests that the model is generalizing well to the unseen images and predicting properly. The validation accuracy of the model reaches to 0.86 and it plateaus. The validation loss of the model also settles at around 0.4.

A screenshot of a graph

AI-generated content may be incorrect.

*Figure 17: confusion matrix for the potato model.*

The above image shows the confusion matrix for the final potato disease detection model. The model can accurately detect diseases such as bacteria, where only 7 images were only misclassified. Fungi, where 111 are classified correctly and virus, which has 87 images that has been classified correctly. However, the model has the weakest performance in the nematode class. This may be due to low number of samples that are available in the test set. Moreover, the healthy class also shows some misclassification where 11 samples were classified as pest, 7 as virus and 2 as fungi. Pest class also shows the same behaviour with 18 classified as fungi.

## Web app

After training the models as the means of brining all the models under one place the web app was created for users to easily use the models. The users can upload the images that they have and get the recommendations for the diseases. The webapp is kept at a very minimal level as the users with no experience prior can easily engage with the models and get the output. The usability test was done with a user, and the results are shown below.

A white sheet of paper with black text

AI-generated content may be incorrect.

*Figure 18: ‘Think aloud’ usability test.*

## Robotic Car

A robotic car was also developed in conjunction to the models and the webapp. The robotic car that has been developed has the capability of driving, taking images and detecting diseases with the user’s assistance. The robotic car uses the sun founder’s PI Car X as the base and Raspberry Pi 4B with 1GB RAM was used as the brain of the car with the Robot HAT for controlling the actuators. The car can either be equipped with either of the models with minor adjustments such as converting the. keras file to Tflite to be used in edge devices. The car was made to driven by the user to the intended location and take the image when the command is given by the user, and the picture will be automatically analysed, and the output will be given to the user.

The code that was used for the car was modified from an existing code to steer the car. Several parts such as the model for identifying the disease was added newly to the program.

A collage of a robot

AI-generated content may be incorrect.

*Figure 19: The robotic car.*

Figure 19 shows the robotic car that was created to demonstrate the usage of the models in the real-world scenarios. The robotic car also has a GPS sensor which can be used to get the live location of the vehicle which is useful in many applications such as identifying the locations with the specific diseases and for measuring the lethality of a disease.

A screenshot of a computer program

AI-generated content may be incorrect.

*Figure 20: menu for controlling the car.*

The figure 20 shows the menu to control the car through the keyboard where the users can use to drive the car to the location they intend to analyse and take the photo which the users can then get the disease name and the confidence level on the prediction.

A screenshot of a computer

AI-generated content may be incorrect.

*Figure 21: output after analysing the photo.*

The figure 21 shows how the users will get the prediction from the car after they have initiated to take the photo of the diseased area.

## Related work

Over the years some of the technologies that has been used by the researchers include the use of transfer learning, where the models which are already trained but only needs to be tuned. These includes models such as ResNet50, EfficientNet as these seems the [18, pp. 566–571]. Furthermore, some researchers[19, p. 144024] have concentrated on single diseases which is good as the training and inference times can be faster and it may be efficient if the field does not necessarily contain any other diseases. Some researchers have use Long Short-Term Memory in conjunction with the CNN architecture to improve the performance of the model[20, Vol. 9]. Furthermore, transfer learning methods are also used for the feature extraction of the plant diseases by the researchers[21, pp. 45377–45392].

When it comes to the robotic car researchers have used Lego programable brick as the brain of the system to control the robot and the model for the robot is trained using the VGG pre-trained model due to the low size of the model[22, pp. 1–8].

Additionally, there are no websites that allow the users to upload the images of potato or wheat leaves for identification of diseases and the models that has been developed only has been published to the GitHub.

# Conclusion

All in all, this project aimed at providing an application for detecting diseases in the wheat and potato plants as these are most grown types of crops in the UK. When it came to the dataset selection process features such as quality of the data, number of classes that are in the dataset where higher the number the classes the more diseases that the model can identify. It was mentioned that some of the images for the dataset will be collected from the real-world however, it could not be done due to time constraints. The wheat model which took the most amount of time to develop, 9 different model iterations before training the final model. Each of the models that was trained helped in deciding what must be done in the upcoming models which also helped in the making of the potato model which only needed 3 iterations before finding out the best model that was used in the artefact. The validation accuracy of the final model of the wheat is 0.80 and the validation accuracy of the potato model was 0.86.

After training two models specifically for wheat and potato a webapp was created which integrated the two models for the use of any user and for easy usage by a non-technical person. As well as webapp a robotic car was also developed with the integration of the wheat model which the user can drive the car with the use of the keyboard and take the picture where the car can analyse the disease and provide the output.

## Future work

Both the models that are used for this project can be further improved to at least 0.90 to 0.95 but requires more time and computation power to test with higher batch numbers. Furthermore, the use of keras tuner to find the best model parameters must be tested to find out the best model parameters for the models. The figure 21 shows how the keras tuner is working to find out the best model parameters. However, as the image depicts that this program can take some time to complete compared to a normal model which means that a machine with higher computation power such as a machine that is equipped with a graphical Processing unit is which handles parallel processing is required to execute the program in a much faster pace.

It also seen that there are no websites that allow the users to find out about diseases in their wheat and potato plants however, the webapp can be made public for everyone’s use so the users can upload the image and get disease and the recommendations for the disease that their plants are having. Furthermore, the robotic car can also be further improved by adding GPS and saving all the scan records into a CSV file where machine learning principles then can be applied to unravel more information about the data that was collected. In addition, pesticides can also be added to the car to the car that can be sprayed on the weeds to protect the plants that of interest or swarm robotics can be used where one or more set of robots identify the where the diseases and weeds are and another set of robots can be used to apply the solutions that are needed to overcome the situation.

A screenshot of a computer

AI-generated content may be incorrect.

*Figure 22: running of the keras tuner for the wheat model.*

## Reflection

Throughout the project there was many fields that I discovered and learnt the lesson and applied them back at the project. One of the biggest discoveries during the whole project is during the training of the wheat model where it took more than 9 iterations of different models to find out the best model that achieves the validation accuracy this led me to the finding of the problem of overfitting and solving it was a breakthrough for the model improvement for both wheat and potato models. The robotic car was also another biggest achievement of this whole project which is one of the two artefacts that was created. All the solutions that were created can be used readily with minor improvements to the car such as elevating the camera such that it can capture images that are clearer.

During the project there were many challenges that was faced such as finding proper datasets. However, datasets were collected from Kaggle but did not contain equal amounts of images as this can create a class imbalance which can make the models bias the only way is to create the datasets by our own, but such creation takes both time and labour which the time and the resources did not allow. The other major setback was the time it took to train the models. Each wheat model took more than 6 to 7 hours to finish training and there are 10 models that was trained to observe the results which took more time than initially allocated in the project proposal and This acted as a domino effect on the whole project. The lesson learned from that was to use a machine with higher computation power to reduce the training time of the models which could have saved time and made an even better model.

During this project lifecycle many lessons were learned such as maintaining records of the progress and getting ready for supervisor meeting and achieving all the milestones that are set by the supervisor which helped to be on focus and organized which helped to deliver a good artefact. In overall, this project has shown which areas that needs more attention and knowledge.

# References

[1]

V. Singh, N. Sharma, and S. Singh, “A review of imaging techniques for plant disease detection,” vol. 4, pp. 229–242, 2020, doi: 10.1016/j.aiia.2020.10.002.

[2]

V. K. Shrivastava, C. J. Shelke, A. Shrivastava, S. N. Mohanty, and N. Sharma, “Optimized Deep Learning Model for Disease Prediction in Potato Leaves,” vol. 9, Sep. 2023, doi: 10.4108/eetpht.9.4001.

[3]

F. Chollet, *DEEP LEARNING with PYTHON*. United States of America: Manning, 2015, p. 107.

[4]

A. Xenakis, G. Papastergiou, V. C. Gerogiannis, and G. Stamoulis, “Applying a Convolutional Neural Network in an IoT Robotic System for Plant Disease Diagnosis,” presented at the 2020 11th International Conference on Information, Intelligence, Systems and Applications (IISA, 2020, pp. 1–8. doi: 10.1109/IISA50023.2020.9284356.

[5]

L. Li, S. Zhang, and B. Wang, “Plant Disease Detection and Classification by Deep Learning-A Review,” vol. 9, pp. 56683–56698, 2021, doi: 10.1109/ACCESS.2021.3069646.

[6]

A. Tabbakh and S. S. Barpanda, “A Deep Features Extraction Model Based on the Transfer Learning Model and Vision Transformer ‘TLMViT’ for Plant Disease Classification,” vol. 11, pp. 45377–45392, 2023, doi: 10.1109/ACCESS.2023.3273317.

[7]

G. Shrestha, M. Das, and N. Dey, “Plant disease detection using CNN,” presented at the 2020 IEEE applied signal processing conference (ASPCON), IEEE, 2020, pp. 109–113. doi: https://doi.org/10.1109/ASPCON49795.2020.9276722.

[8]

I. Goodfellow *et al.*, “Generative adversarial networks,” vol. 63, no. 11, pp. 139–144, 2020, doi: https://doi.org/10.1145/3422622.

[9]

M. H. Saleem, J. Potgieter, and K. M. Arif, “Plant Disease Detection and Classification by Deep Learning,” vol. 8, no. 11, pp. 468-, Oct. 2019, doi: 10.3390/plants8110468.

[10]

S. D. Khirade and A. B. Patil, “Plant Disease Detection Using Image Processing,” in 2015 International Conference on Computing Communication Control and Automation. IEEE, Feb. 2015, pp. 768–771. doi: 10.1109/ICCUBEA.2015.153.

[11]

A. W. Salehi *et al.*, “A Study of CNN and Transfer Learning in Medical Imaging: Advantages, Challenges, Future Scope,” vol. 15, no. 7, p. 5930, Mar. 2023, doi: 10.3390/su15075930.

[12]

Y. Liu, G. Gao, and Z. Zhang, “Plant disease detection based on lightweight CNN model,” in 2021 4th International Conference on Information and Computer Technologies (ICICT). Piscataway: IEEE, Mar. 2021, pp. 64–68. doi: 10.1109/ICICT52872.2021.00018.

[13]

N. Wang *et al.*, “Key Challenges in Plant Pathology in the Next Decade,” vol. 114, no. 5, pp. 837–842, May 2024, doi: 10.1094/PHYTO-04-24-0137-KC.

[14]

J. Jiang, X. Jiang, L. Xu, Y. Zhang, Y. Zheng, and D. Kong, “Noise-robustness test for ultrasound breast nodule neural network models as medical devices,” vol. 13, p. 1177225, Jun. 2023, doi: 10.3389/fonc.2023.1177225.

[15]

D. Suri, S. Saksenaa, U. Sehgal, and R. Garg, “Disease Classification in Wheat from Images Using CNN,” presented at the 2023 13th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 2023, pp. 566–571. doi: 10.1109/Confluence56041.2023.10048813.

[16]

T. Liu *et al.*, “High-throughput identification of fusarium head blight resistance in wheat varieties using field robot-assisted imaging and deep learning techniques,” vol. 480, p. 144024, 2024, doi: 10.1016/j.jclepro.2024.144024.

[17]

K.-C. Cheng, “The inexplicable rise of kidney disease in Sri Lanka’s farming communities,” 2024. Accessed: 2025. [Online]. Available: https://www.aljazeera.com/features/2024/4/28/the-inexplicable-rise-of-kidney-disease-in-sri-lankas-farming-communities

[18]

K. Nanthini, D. Sivabalaselvamani, K. Chitra, P. Gokul, S. KavinKumar, and S. Kishore, “A survey on data augmentation techniques,” presented at the 2023 7th International Conference on Computing Methodologies and Communication (ICCMC), IEEE, 2023, pp. 913–920. doi: https://doi.org/10.1109/ICCMC56507.2023.10084010.

[19]

A. Kalia and S. K. Gosal, “Effect of pesticide application on soil microorganisms,” vol. 57, no. 6, pp. 569–596, Sep. 2011, doi: 10.1080/03650341003787582.

[20]

A. Abade, P. A. Ferreira, and F. de Barros Vidal, “Plant diseases recognition on images using convolutional neural networks: A systematic review,” vol. 185, p. 106125, Jun. 2021, doi: 10.1016/j.compag.2021.106125.

[21]

J. Chen, J. Chen, D. Zhang, Y. Sun, and Y. A. Nanehkaran, “Using deep transfer learning for image-based plant disease identification,” vol. 173, p. 105393, Jun. 2020, doi: 10.1016/j.compag.2020.105393.

[22]

M. A. Noyan, “Uncovering bias in the PlantVillage dataset,” Jun. 2022, doi: 10.48550/arxiv.2206.04374.

# Appendices

## Appendix A: project proposal

# AI-BASED CROP DISEASE DETECTION AND TREATMENT RECOMMENDATION SYSTEM.

# Introduction

This project aims on designing a system to identify plant diseases based on the pictures of the leaves of the specific trees and recommend the necessary steps to avoid the disease in the future.

# Who is this system for?

This system is mainly aimed at the people who needs a system for identifying the diseases in the field which includes users like farmers for identifying the diseases that they may not know about, and the other users can include agricultural chemical suppliers to identify which crops has which disease more commonly and in which place to effectively produce, market and sell their products. The other user of the system could be the public such as when they have plants at their house and want to identify the disease and find out the disease. However, there can be even more users that are not mentioned over here.

# Objectives

* Collect and clean the necessary datasets that are used for the model through the web and real-world.
* Train a Convolutional Neural Network (CNN) based on the collected data to successfully find out the disease.
* Find the disease that the plant is having based on the leaf and the spots on the leaves.
* Recommend the necessary steps that must be taken to cure the plant or to avoid the disease in the future.

# Other types of machine learning techniques used in the crops

The other types of machine learning that are used in the crops include multiple linear regression where it uses inputs such as humidity, temperature, soil conditions to consideration to provide an output or other types may include techniques such as transfer learning using reznet-50. However, CNN is much less complex than the reznet-50 which can be easier to handle.

* 1. Pre-processing the data such as labelling and noise removal.
  2. Finalize the technologies to be used in the project.
* Phase 02: Model Development and training (1.5 month):
  + 1. Further pre-processing such as normalizing images.
    2. Model training.
    3. Fine-tuning hyperparameters.
    4. Model evaluation.
    5. Final tuning based on the model evaluation results.
* Phase 03: Recommendation system development (1 month):
  + - 1. Recommendation engine integration
      2. Testing the full functionality of the connected systems.
* Phase 04: System integration and deployment (1 month):

1. User interface design and deployment.
2. Backend integration.
3. Testing and debugging.

* Phase 05: Evaluation, documentation and final presentation (2 weeks):

1. Evaluate the full system performance on the field.
2. Document the full system architecture how it works and the system specifics.
3. Prepare the final presentation to show case the full system functionalities and how the system works.

## Appendix B: evidence of project management.

A screenshot of a computer

AI-generated content may be incorrect.

This shows evidence of the project management that was done on the GitHub projects. The tasks were categorized for 6 main parts and as each one was moved in stages the cards were also moved.

## Appendix C: Additional Technical details

The extra technical details can be found on the GitHub page with this link: <https://github.com/sachindra2000/PLANT-DISEASE-DETCTION-AND-RECOMMENDATION-APP>

All the details, notebooks and other useful information will be provided in the GitHub page of the website

## Appendix D: Supervisor meetings.

please find below the supervisor meetings throughout project below.

# Notes from supervisor meeting(1st):

Supervisor: Mohammed F. Khan

Mode: on-campus

Date: 06/11/2024

Time: 14.00

1) try to find 2 distinct types of leaves of the crop and only train the dataset for those 2 diseases.

2) try to make it more towards the UK market.

3) do more research about the topic and about the datasets preferably from Kaggle.

4) talk to Charles about the robotic car.

5) what kind of hardware to be used in the robotic car such as depth sensor.

# Notes from the supervisor meeting(2nd):

Date: 18/11.2024

Mode: on-campus

Time: 10.45

1. Start to train the model based on the identified plants and the diseases.
2. Check the accuracy of the model.
3. Then do the feature engineering of the model using the Otsu method and or histogram of oriented gradients.

# Notes from the supervisor meeting(3rd):

Date: 02/12/2024

Mode: on-campus

Time: 10.40 AM

1. Questions to ask the sir:

What must be done for the technological review.

Read the papers and gather the information for the review.

1. Apply early stopping.
2. Do feature extraction, feature engineering.
3. Try the transfer learning models too.

# Notes from the supervisor meeting(4th):

Date: 17/12/2024

Mode: on-campus

Time: 14.00

1. Apply the hsv colorspace for the model.
2. Try to visualize the output the output of the Otsu method.
3. Do the research on the data about the plant disease.

# Notes form the supervisor meeting (mid-point review 1st):

Mode: Online

Date: 13/01/2025

1. Include the images of the results that has been achieved so far for the models that have been created.

# Notes from the supervisor meeting(5th):

Date:28/01/2025

Mode: Online

Time: 10.30 AM

1. Keep working on the wheat model.
2. Start the report and add the work that you have done so far.

# Notes from the Supervisor meeting(6th):

Date 11/02/2025

Mode: Online

Time: 10.40 AM

1. Start doing the report as some work is already done.
2. Change the pictures on the website to actual from the AI generated images.
3. Optional: start working on the report in the way that it can be made it into the research paper.

# Notes from the supervisor meeting(7th):

Date 27/02/25

Mode: Online

Time: 10.30 AM

1. Work on the report.

# Notes from the supervisor meeting(8th):

Date: 03/03/2025

Mode: on-campus

Time: 13.30

1. Work on the recommendations for the outputs.

# Notes from the supervisor meeting(9th):

Date:17/03/2025

Mode: on-campus

Time: 13.30

1. Showed the RC car capable of detecting the wheat diseases.
2. Have printed images for the detection of the diseases rather than showing images in the screen.