Disease identification and Management system for Gherkin Cultivation

Perera H.M.A.G

(IT21022388)

BSc. (Hons) in Information Technology

Specializing in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology

Sri Lanka

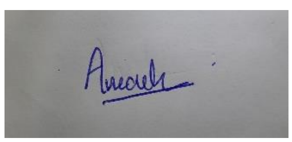
DECLARATION

I hereby declare that the work presented in this thesis titled *"Disease identification and management system for gherkin leaves"* is my own and has been carried out under the guidance of Mr. Dharshana Kasthurirathna and Ms. Poojani Gunathilake at Sri Lanka Institute of Information Technology (SLIIT). This research has not been submitted previously, either in part or in full, for any other degree or qualification at any other institution.

The information and material derived from other sources have been properly cited and referenced. I confirm that I have adhered to the academic integrity policies and guidelines of SLIIT throughout this research and thesis writing process.

In addition, I affirm that the thesis is an original work, and I have conducted the research and analysis independently. All contributions from other researchers and sources have been acknowledged appropriately.

Name: Hewa Mudalige Amanda Gimhani Perera

Signature: 

Date: 02nd August 2024

The Supervisor/s should clarify the dissertation with the following declaration.

The above candidate has carried out research for the bachelor's degree dissertation under my supervision.

Signature Of the Supervisor

…..…………………………

Kasthurirathna D

Signature of the Co-Supervisor

……..………………………..

Gunathilake M.P

ABSTRACT

This research aims to fill the gap of a single system that can perform object detection, disease classification and analyze environment in case of gherkin leaf diseases. The study proposes a solution that, in addition to detecting gherkin leaves and creating a disease diagnosis framework through CNN architectures, will be able to capture climatic factors, such as temperature and humidity, to advise on disease control. The primary objective is to design a reliable framework containing the object detectability through YOLOv8, disease classification by CNN based models such as Efficient Net and mapping with the Gemini model to examine the environment and propose solutions including artificial intelligent. It will also offer selections in Sinhala to ensure maximum usability to the farmers in Sri Lanka. Hence, while current approaches seek to identify the disease while ignoring other factors or when offer recommendations without a holistic perspective of certain diseases, this paper presents a novel approach that integrates several components to enhance the means used in previous techniques of disease detection, environmental analysis, and management solutions. This abstract adheres to the required structure it presents the problem, the reason for the report, the method including object detection, CNNs, and the environment analysis and presents the general solution as final without any positive or negative tone.

[Keywords: ***Machine Learning, Object Detection, Disease Detection, Disease Classification, Gherkin Cultivation, CNN Architectures, YOLOV8***]

ACKNOWLEDGEMENT

I would like to extend my heartfelt gratitude to all who have supported and contributed to this research's completion.

First and foremost, I would like to express my sincere gratitude to both my supervisor, Mr. Dharshana Kasthurirathna, and my co-supervisor, Ms. Poojani Gunathilake, for their insightful input, ongoing support, and vital assistance during this project. Their knowledge and support were invaluable in determining the course of my work and guaranteeing its success. Additionally, I would like to express my gratitude to my group members for their helpful critiques and recommendations, which significantly improved the caliber of this thesis.

Special thanks to Research Project team of Sri Lanka Institute of Information Technology, who had organized various informative sessions to give guidance throughout the project and SLIIT for providing the resources and facilities necessary for this research.

I am also grateful to my colleagues and friends for their support, encouragement, and helpful discussions which were essential during the research process. My deepest appreciation goes to my family for their unwavering support and understanding throughout my academic journey. Their patience, encouragement, and belief in my abilities have been a constant source of motivation.

Thank you all for your support and belief in me.

TABLE OF CONTENTS

DECLARATION i

ABSTRACT ii

ACKNOWLEDGEMENT iii

1 INTRODUCTION 1

1.1 Background & Literature 1

1.1.1 About gherkin cultivation and demand 1

1.1.2 About Hayleys gherkin cultivation and challengers 2

1.1.3 Proposed mobile application for gherkin disease identification 3

1.2 Research Gap 7

1.3 Research Problem 9

1.4 Research Objectives 10

2 METHODOLOGY 11

2.1 Methodology 11

2.1.1 Data Collection 11

2.1.2 Data Preprocessing 12

2.1.3 Gherkin Leaf Identification Using Object Detection 14

2.1.4 Gherkin Leaf Disease Classification Using CNN Architectures 16

2.1.5 Environmental Factor Integration and Provide Solutions Using the Gemini Model 19

2.1.6 Translate to Sinhala Language 21

2.1.7 Downy Mildew Disease Severity Classification 22

2.2 Commercialization Aspect of the Product 25

2.2.1 Market Feasibility 25

2.2.2 Potential Product Features 26

2.2.3 Challenges and Considerations 26

2.3 Testing and Implementation 27

3 Result & Discussion 32

3.1 Results 32

3.2 Research Findings 32
3.3 Discussion 33
4 Conclusion 34
5 References 36
6 Appendices 38

List of Tables

Table 1: Temperature and humidity levels of diseases 5
Table 2: Research Gap 9

List of Figures

Figure 1: Initial Stage 6
Figure 2: Progressive Stage.. 6
Figure 3:Final Stage 7
Figure 4: Nikawaratiya Field Visit 11
Figure 5:Meeting with the Company.. 12
Figure 6:Image Augmentation... 13
Figure 7:YOLOV8 Model Training.. 15
Figure 8:YOLOV8 model testing 15
Figure 10:Healthy Leaf... 16
Figure 9:Gummy Blight 16
Figure 11:Downy Mildew.. 16

List of Abbreviations

CNN	Convolutional Neural Network
YOLOV8	You Only Look Once Version 8
VGG19	Visual Geometry Group 19
AI	Artificial Intelligence
Mask R-CNN	Region-based Convolutional Neural Network
ML	Machine Learning
API	Application Programming Interface
HTTP	Hypertext Transfer Protocol

1 INTRODUCTION

1.1 Background & Literature

1.1.1 About gherkin cultivation and demand

Gherkin farming has great importance in the world agriculture market because the use of this versatile product is a common practice in all countries. Common gherkins are crispy while gherkins are soft but in size, shape and even at times taste they are same though gherkins are associated with pickles. The global consumption of gherkins especially in the form of pickles fascinate and the major consumers are in European, Asian and Americas.

Globalization in today's world is producing heightened demands for agricultural produce to feed the growing population calling for efficiency in crop production. Cucumber is a common vegetable crop, and it is prone to several diseases which have a compromising impact on yield and quality. Due to this the early diagnosis and right identification of these diseases are very important so that early and correct management steps can be taken. AI and machine learning borrowed from modern technology aid in establishing other systems that would help the farmers to identify and control diseases. But one of the main issues associated with farming gherkin is that they are susceptible to various diseases that pose a big threat to the production of this fruit crop. Europe, Asia and Americas are the largest consumers of gherkins, gherkins are the most popular in the three regions. Today leading producers and exporters in this category include India, Sri Lanka, Mexico and Turkey among others. These countries rise to meet the growing demand for pickled gherkins, particularly in Europe and North America where the demand seems to rise with time for the tangy, crunchy product. The market for pickles across the world is expected to grow because of special food and changing consumption trends particularly towards convenience food.

Sri Lanka have clearly marketed themselves as large-scale exporters of gherkin pickles, getting more value added due to the special taste and smell which is due to the tropical climate conditions [1]. Light and a favorable environment for the growth of gherkin is always available in Sri Lanka because of the favorable geographical setting of sunlight throughout the year helps to produce better quality gherkin products, and hence, these are in great demand in international markets. Currently Sri Lanka has carved out a niche market for herself and this is in the international market with gherkin pickles. In the recent past, Sri Lanka's export market of gherkin pickles has greatly expanded, and the country has well-advertised quality products. Nevertheless, due to the captive market, the chance at profitability, and as seen with current consumers, there exists a demand for gherkins in Sri Lanka, this crop has not exploded in popularity amongst farmers. At the present time only, several firms engage in production and export of gherkins and therefore the development of gherkin as an agricultural produce is still restricted in the mentioned country [2].

Since livestock raising becomes highly profitable and enhances food security and application of sustainable technologies in farming, to the same extent, growing gherkins also becomes profitable and beneficial. It would only make great sense because it is easy to expand the area under cultivation, and it is rainfall-friendly besides hardiness to a variety of soil types and climatic conditions that smallholder farmers can grow the crop. Moreover, it is more viable to grow gherkins in that they only need several productions in a year, hence, the more money the farmers make.

The demand for healthy food is increasing worldwide, the changing customer preference towards pickled food products is also augmenting the gherkin market. Consequently, through its high vitamin and mineral content with attraction to the consumer group that considers it as virtually calorie-free snack product, gherkins won the consumers' favor. That way, this growth offers farmers in Sri Lanka a great opportunity to increase their crop holdings, and more importantly, their wealth.

1.1.2 About Hayleys gherkin cultivation and challengers

However, despite the potential, gherkin cultivation is not widely practiced among Sri Lankan farmers. The main obstacle is farmers' lack of resources and knowledge, which restricts the amount of land that can be farmed. Technical expertise among local farmers regarding modern agricultural practices and gherkin-specific cultivation techniques. Many farmers are unaware of the high-yielding hybrid varieties available, which can significantly enhance productivity. There are currently just two or three enterprises in Sri Lanka that are actively engaged in the cultivation and export of gherkins. This low level of participation emphasizes the need for greater support and information sharing to promote Gherkin Farming's wider adoption.

Hayleys Agriculture company is one of the main Gherkin product providers of Sri Lanka. In the 1950s, due to a shortage of food, the Hayleys Agriculture department was created to cover the country's agricultural demands. The Hayleys group later made an entry into the export trade in1988 by supplying semi-processed gherkins pickled in brine to pickle production factories in Australia, Europe, Japan, and New Zealand. Our research team is engaged with Hayles Agriculture to provide software solutions for their gherkin cultivation challengers. The local agricultural community is facing a lot of troubles when cultivating. They believe in supporting local farmers and creating a secure environment for them to grow and harvest their crops. Along with that, we researched the difficulties they faced and created four components related to that. Farmers are struggling with the diseases. Because every year farmers lose a high number of crops. Sometimes farmers are unable to identify the exact disease. What is the reason for this disease and what are the exact solutions. Throughout my component overcome all the diseases related difficulties.

This study therefore recommends a new concept mobile application to support farmers who have interest in growing gherkins as a solution to these challenges. The objective of this application is to make a comprehensive tool turning into a guide full of essential information to enhance the profitability and environmental impact of gherkin farming on the market. Searched elements for application disease identification, pest identification, harvest prediction and production cost prediction are based on the advanced machine learning and computer vision technology. That primarily and most significantly, there are several challenges faced in exporting gherkin in Sri Lanka which affects the growth and the competitiveness in the sector. These challenges are grouped into the following classes: environmental factors, diseases, post-harvest handling, marketability, and production constraints.

Farmers will solve problems affecting crops rapidly and accurately due to the disease and pest identification option, which decreases crop damage and enables suitable intervention. The use of PA techniques and object detection algorithms in the application ensures that the application presents the best data that any farmer would require for his or her farming business. This will enhance farmers economic status by raising crop output and diminishing costs of risks in production. Besides, this solution would enhance productivity and sustainability of gherkin production and develop the position of Sri Lanka as the exporter of the best quality gherkin pickles in the international market.

1.1.3 Proposed mobile application for gherkin disease identification

It is for this reason that this study proposes the development of an innovative mobile application targeted exclusively to farmers who wish to grow gherkins to neutralize these challenges and boost crops production. In the application, four main components will be considered. Here there are disease identification and other stated management components. The especial app for farmers will allow them to recognize the diseases that influence gherkin plants at an early stage allowing them to manage them properly.

Similar to many other plants, Gherkin is susceptible to diseases such as downy mildew, Gummy stem blight and viral diseases that can lead to huge losses. All these diseases prove to be very dangerous to gherkin plants. Successful management of healthy crops and the safety of food crops largely depends on early diagnosis and control of these diseases. The conventional approach to diagnosing disease involves a considerable amount of time, as well as technical know-how, all of which are a hurdle to most farmers.

This element of the Disease Identification and Management System utilizes machine learning algorithms such as object detection utilizing YOLOv8 to recognize gherkin leaves. Implemented four CNNs architectures to detect gherkin leaves-diseases. The four GAN models I used are Inception, Efficient Net, Dense Net and VGG19. Employing these CNN architectures, this system measures downy mildew infected leaves, gummy blight infected leaves and healthy leaves [3]. Out of these four architectures of CNNs, I chose the best architecture for identifying diseases. First, the system includes environmental indices such as humidity and temperature to identify the main sources of the identified diseases. The inputs are taken into consideration in the Gemini model and then come out with the solutions and these results are interpreted to the local farmers by translating the contents to Sinhala language. Also, the system can predict the intensity level of the downy mildew disease through an integrated smart system.

- **Identify gherkin leaf using YOLOv8**

Find out if the shape of the leaf is gherkin like or not. To do that, use an object detection technique known as YOLOv8. Object detection has become one of the most useful technologies in the field of agriculture. The state-of-the-art object detection model which provides the real-time results is YOLOv8 (You Only Look Once version 8) popular for speed as well as accuracy. Ideally, it might be trained in the category of objects, for example, some gherkin leaves; thereby facilitating the automation of procedures for identifying the diseases. YOLOv8 has a detection head that produces bounding boxes and class probabilities, neck that forms feature pyramids and backbone to help in feature extraction.

- **Identify the gherkin leaf diseases using CNN architectures of Inception, EfficientNet, ResNet, DenseNet**

Convolutional Neural Networks (CNNs) are particularly effective for image classification tasks, including plant disease identification. Several architectures have been used to classify gherkin leaf diseases, including Inception, EfficientNet, ResNet, and DenseNet. These models are excellent at extracting features, and they may be adjusted to identify diseases like sticky stem blight and downy mildew more accurately. According to research, CNNs can reach high classification accuracy, which qualifies them for real-time agricultural applications. These models will provide the output as disease name. Using these machine learning models, we identified good gherkin leaves, downy mildew affected leaves and gummy stem blight affected leaves. Trained four CNN models and continue with the more acerated CNN model.

- **Using temperature and humidity, provide the solutions by using Gemini AI**

Two important variables affecting the emergence and spread of plant diseases are temperature and humidity. Research has demonstrated that environmental factors can either facilitate or impede the advancement of a disease. For example, downy mildew prefers humid environments, but excessive humidity and certain temperature ranges aggravate angular leaf spots. The approach can offer insights into the main drivers of disease outbreaks by integrating these environmental factors into the disease detection process.

Give the humidity and temperature as inputs to identify what is the main reason for this disease. High and low humidity and temperature are the main reasons for these two diseases. By comparing these details, the system will decide what is the main reason for disease.

Disease Name	Temperature Range	Humidity Level	Details
Downy Mildew	15°C to 20°C (59°F to 68°F)	High (above 85%)	Thrives in cool, moist conditions with prolonged leaf wetness.
Gummy Blight	20°C to 25°C (68°F to 77°F)	High (above 85%)	Requires high humidity and warm temperatures, more severe with high rainfall or overhead irrigation.

Table 1: Temperature and humidity levels of diseases

This table provides a quick reference for the temperature and humidity conditions that favor the development of these diseases, helping in better management and prevention strategies in gherkin cultivation. The Gemini model, which uses AI to provide recommendations based on various inputs, offers a novel approach to integrating these environmental factors with disease detection outputs to suggest appropriate management strategies.

Data Input: The model takes in various inputs, including real-time temperature, humidity, and disease detection results.

Analysis: Using trained algorithms, the model analyzes the correlation between environmental conditions and the severity or likelihood of specific diseases.

Recommendation: Based on this analysis, the model suggests tailored management strategies, combining cultural practices, chemical controls, and preventive measures.

- **Translate solutions to Sinhala language**
  The Gemini model's answers are translated into Sinhala because many farmers might not be able to understand English. Implementing recommendations effectively may be difficult by language limitations. Through the translation of the solutions into Sinhala, the system guarantees that farmers comprehend the offered advice completely. By improving the system's usability and accessibility, this step guarantees that all farmers, regardless of language ability, may take advantage of the technology.

Through the mobile application interface users can view the full result. That will include the name of the disease, main reason for disease, solutions for getting rid of disease and translate it to Sinhala language.

- **Identify severity level of Downy mildew disease**

To categorize the three stages of Downy mildew disease (initial, intermediate, and terminal) by using Mask R-CNN, we require a dataset containing gherkin leaves at different levels of infection. For analysis of features of interest, these images should be labelled to highlight the diseased areas, generating segmentation masks. Once the data is prepared, we can take a pre-trained Mask R-CNN model and train it on my data set with appropriate class numbers including the background, mild, moderate and severe. The first step is to preprocess the images in such a way that they are in the correct size input of the Mask R-CNN and normalized, then one can train the model to produce masks for the diseased areas and classify the areas in to correct severity levels. Since the model provides segmented masks, the user can calculate the percentage of the infected area on each of the leaves and thereby determine the level of severity depending on the size of the affected region. For instance, small yellow spots on the external part of the leaf or infected area might be pointed to an early Stage. Following training, use metrics such as Intersection over Union for the masks, accuracy for the severity levels of the infections and fine tune the model for optimum result.

Three severity stages

- Initial Stage: Yellow spots on the leaf surface.
- Progressive Stage: Brown lesions and moldy growth on the underside.
- Final Stage: Browning and death of the leaves.

Figure 2: Initial Stage

Figure 1: Progressive Stage

Figure 3:Final Stage

1.2 Research Gap

The absence of the integrated system that can include object identification, disease classification and environmental analysis for gherkin leaf diseases has been due to the development of machine learning and image processing techniques. Unfortunately, most existing approaches fail to consider multivariate relations and tend to focus on specific components, with environment control or disease identification. In this paper, it is the intent to bridge this gap by designing a comprehensive system that includes all those features.

However, despite the progress made in plant disease detection using AI, many existing systems lack total system integration of various functionalities. Most methods fail to consider components that may hamper the progression of a disease by focusing more on mere image analysis of a disease. Moreover, limited studies are devoted to exploring localized solutions, which consider environmental conditions.

For this purpose, the presented approach aims at developing a coherent framework that incorporates object detection, CNN-based illness identification, environmental factor analysis, and AI-driven solution development and, in addition, measuring the severity level of disease. In addition, the problem regarding the accessibility for nearby farms is solved the translation into Sinhala, so that the recommendations given by the system are understandable and feasible.

Moreover, based on the identified disease and environmental factors most of the current approaches lack a management plan. Regarding the disease severity, developmental state, as well as the circumstances in which the environment is conditioned, the proposed research contains a Gemini model that offers farmers treatment options. This characteristic differentiates the suggested approach from previous research and enhances its application for farmers. For example, Mohanty et al. (2016) have developed a deep learning model which has a high predictive ability when distinguishing between 26 diseases in 14 crop varieties. Like Marbach

[9], Ferentinos in 2018 achieved outstanding performances when detecting plant diseases in images using deep learning models [4].

Though Mohanty et al successfully demonstrated high accuracy in the field of classification the study did not go a step further to perform an environmental scan and make practical recommendations that could be significant to the farmers. The proposed system builds on this work by incorporating environmental monitoring of temperature and humidity to the disease diagnosis and proposing artificial intelligence-based individualized solutions to the problem.

Ferentinos used CNN models for identifying diseases in plants among different crop varieties by dataset of images. The study obtained a high level of detecting accuracy but ignored other contextual factors in identifying the images apart from categorizing them. Lack of analysis of environmental condition, and localized approach advice like disease management recommendations reduce the usability of this work in actual farming situations. Compared to the image classification of remote sensing data, the proposed system includes the temperature and humidity parameters into a quantitative analysis model to supply more operational real-time decision-making references for the farmers [5].

The proposed research work used CNNs in detecting plant leaf diseases prevalent in large scale farming. But it has been dedicated mainly towards enhancing the accuracy of images classifications while ignoring other factors such as the environment and individual recommendations. Similar to other works that aimed at developing image-based plant disease detection models, including the study by Zhang et al., the value of environmental factors in disease development or control was not explored. Compliance with other environmental factors in disease prediction along with localized recommendations in Sinhala makes the proposed system more effective for the farmers in comparison with the existing system [6].

Deep learning as well as object detection models such as Faster R-CNN were employed by Fuentes et al. to identify various diseases in tomato crops [7]. While this study dwelt much on disease diagnosis, it lacked information on environmental factors, which are crucial in disease diagnosis and finding solutions to diseases. The integration of YOLOv8 for object detection alongside CNN based diseases and environment identification and analysis makes the proposed system more complete and more applicable in real world situations where farmers are managing many different factors simultaneously.

However, these methods do pay much attention to image categorization, while they rarely involve surrounding conditions. Moreover, while YOLO and other object detection models have been used in other domains, the use of YOLO on agriculture particularly for the classification of the leaves is relatively new. There is not enough integration between disease detection models and decision support systems like the Gemini model, the latter of which has the capability to give suggestions based on an input [8].

Application Reference	Research 1	Research 2	Research 3	Research 4	Proposed System
Identify leaves using object detection techniques	✓	×	✓	×	✓
Identify gherkin leaf diseases using different CNN architectures	✓	×	×	×	✓
Identify leaves disease using image classification techniques	✓	✓	✓	✓	✓
Identify the effect of the humidity and temperature levels, provide solutions for the diseases using Gemini	×	×	×	✓	✓
Provide solutions to get rid of the diseases	×	✓	×	✓	✓
Convert solutions to Sinhala language	×	×	×	×	✓
Identify the severity levels of downy mildew disease in three stages	×	×	×	×	✓

Table 2: Research Gap

1.3 Research Problem

The primary problem addressed in this research is the absence of an integrated system for the identification and management of gherkin leaf diseases that combines object detection, disease classification, environmental data analysis, and solution recommendations. The research seeks to develop a system that can accurately detect gherkin leaf diseases and provide tailored management strategies based on real-time environmental data.

Plant diseases as phenomena pose a major challenge to agriculture especially when it comes to growing gherkin (Cucumis anguria). Organic diseases of the gherkin leave such as downy mildews and gummy blights are some of the most challenging to diagnose, control, and in most cases, farmers will use simple, time-consuming, error-prone, and subjective methods of inspection. Consequently, the use of a more accurate and efficient method of diagnosing the diseases affecting the gherkin plant leaves can eliminate the necessity of manual handling of images and decision making.

CNNs have been extensively employed in plant disease detection because of its remarkable image recognition performance [9]. However, implementing a fully automated system to detect gherkin leaf diseases requires addressing multiple interconnected challenges.

1.4 Research Objectives

General Objective:

- To develop an integrated Disease Identification and Management System for Gherkin cultivation that combines object detection, CNN-based disease identification, environmental factor analysis, and AI-driven solution generation.

Specific Objective:

- To implement YOLOv5 for accurate detection and localization of gherkin leaves in images.
- To utilize CNN architectures (Inception, EfficientNet, ResNet, DenseNet) for identifying and classifying gherkin leaf diseases.
- To incorporate humidity and temperature data into the system to identify the primary causes of detected diseases.
- To integrate the Gemini model for providing tailored disease management solutions based on image analysis and environmental factors.
- To translate the generated solutions into Sinhala to improve accessibility for local farmers.
- Give the downy mildew disease severity levels as three stages. Then calculate the stage levels of that one leaf has and give as a percentage from the whole leaf. These are the three stages.

Initial Stage: Yellow spots on the leaf surface.
Progressive Stage: Brown lesions and moldy growth on the underside.
Final Stage: Browning and death of the leaves.

This research contributes to agricultural technology by developing a comprehensive system that addresses multiple aspects of disease management in gherkin cultivation. The integration of object detection, CNN-based disease identification, environmental factor analysis, and AI-driven solution generation is a novel approach that has the potential to significantly enhance the effectiveness of disease management practices.

2 METHODOLOGY

2.1 Methodology

This section describes the procedures used in the research work to identify, categorize and diagnose diseases affecting gherkin leaves using machine learning. This provides an understanding of the procedure that was used conflicted in the research when distinguishing and classifying gherkin leaf diseases with the use of automated machine learning approach. It has four components, namely data collection and prepossessing, model choosing, model assessment, and model implementation. The aim of this methodology is to increase accuracy in identifying the diseases affecting the leaves, in disease prediction and in providing relevant information for farmers and based on image detection and climatic conditions.

2.1.1 Data Collection

The foundation of any machine learning-based image classification system is reliability on a dataset. This study aimed at compilation of a good number of gherkin leaf images to represent the variation in light, disease manifestations and leaf morphology.

The primary dataset was created based on images of developing gherkin plants in three categories of healthy leaves, and leaves affected by diseases such as the downy mildew and gummy blight diseases. These images were obtained from the Kaggle website as well as from field surveys made in the nearby Gherkin farms. There is the contract we have with the company of Hayleys. We visited the gherkin farms of Hayleys company to gain knowledge and collect information from the expertise and farmers. I accompanied a field trip to the Nikawaratiya gherkin farm, where I captured images of disease-affected leaves. This allowed us to build a custom dataset using these field-sourced images.

Figure 4: Nikawaratiya Field Visit

Figure 5:Meeting with the Company

For object detection, the dataset is labeled to mark each of the leaves as gherkin leaf and then produced a negative dataset which helps to set a greater percentage of correct prediction. For this work, a data set of 403 images of gherkin leaves was analyzed. These were added to the existing negative dataset, which is a collection of images of non-gherkin leaves and other non-related objects to make the model distinguish good gherkin leaves from others. In the case of the disease classification models the images were split into 3 classes, healthy leaves, downy mildew and gummy blight. There are 800 images in each class including augmented data. Additionally, for cases of downy mildew, the images were further annotated to classify disease severity into three stages, First, second, and third stage.

2.1.2 Data Preprocessing

To achieve this some data transformations were made because the given data was to be useful in machine learning models. To enhance the model's resistance, some form of data augmentation was performed on the training data set by afflicting it in a random manner by flipping, and rotating and changing the color of the images.

- Resizing: All images were resized to 224×224 pixels which is best suitable to all the CNN models that were taken into consideration.

- Rotation: However, to counter the problem of viewpoint, some images were randomly translated around a – or fixed angle within a degree(s). This makes the model to be invariant to the rotational transformation, hence the model will be able to transform the input by any angle.

- Width Shift: To get a displacement within a given fraction of the width of the image the image was translated horizontally at random. This augmentation technique helps the model when the object of interest could be in any position that is horizontal regarding the image.

- Rescale: Rates of scales were provided to resize pictures, such as multiplying by 255/1 to alter Range of the pixel intensities from 0-255 to a range of 0 to 1. This normalization was done with a view to ensuring that the overall complexity of the network during training is de-escalated in that every input is scaled to a consistent range.

- Horizontal Flip: To replicate differences in which objects within the image may be presented in mirrored positions, the images employed were inverted horizontally at random. This makes it more viable when faced with images that might be flipped in other real-life scenarios for generating predictions.

- Height Shift: Random vertically translate and then applied to the image by randomly shifting it within the given fraction of total height. This enhances the model's capability of modelling variations in the extent of the vertical viewpoint for objects in images.

- Zoom: Whenever random zooming in and out of the set images was within a certain limit, then the model proved to be sensitive towards different sizes of the objects with regards to those images. The size augmentation helps build confidence in the model allowing the objects to be easily identified regardless of the size distance from the shooter.

- Data Augmentation: More in augmentation flipping rotation zooming and cropping were also employed to balance the size of the data set artificially. This was done to eliminate the probability of overfitting because we were in the lookout for a model that would fit any data set.

```python
folder_path = 'Images'
output_path = 'Augmented_Downy_Mildew'
image_files = get_image_files(folder_path)
```

```python
for image_file in image_files:
    print(image_file)
    img = load_img(image_file)
    #img.show()
    x = img_to_array(img)
    x.shape
    x = x.reshape((1,) + x.shape)
    x.shape
    datagen =  ImageDataGenerator(
            rotation_range=6,
            width_shift_range=0.1,
            height_shift_range=0.1,
            rescale=1./255,
            shear_range=0.1,
            zoom_range=0.1,
            horizontal_flip=True)
    i = 0
    img_name = image_file.split("\\")[1].split(".")[0]
    for batch in datagen.flow(x, batch_size=1, save_to_dir=output_path, save_prefix=img_name, save_format='jpg'):
            i += 1
            if i >10:
                break
```

Figure 6:Image Augmentation

2.1.3 Gherkin Leaf Identification Using Object Detection

The first step of the proposed research was to determine whether a given input image was of a gherkin leaf or of something else. This was done using the YOLOv8 object detection model, a real-time object detection model currently at the apex of its class. This YOLOv8 architecture was selected due to its high mean average precision achieved together with high speed, which is perfect for any use requiring less latency and high accuracy.

The process for identifying gherkin leaves involved several steps:

- The system was designed in a where it was only allowed to accept only one image at any one time. If a user inputs an image containing multiple leaves, the system returns an error saying that predictions made on such images are unclear in relation to which of the leaves the disease prediction is related to.

- Computer vision based on YOLOv8's object detection approach was used for determining whether the uploaded image contains a gherkin leaf. This was done through optimizing the one stage object detection of YOLOv8 where the input image goes through the CNN then is classified through bounding boxes.

- If the image has more than one leaf or any other object apart from the leaves the system gives an error. This helps improve the ability of the following disease prediction step by providing a clean and reliable input and minimize chances of having false positive and wrong disease classification.

Training and Testing Process

The dataset was randomly divided into training and test set with 80% of images in training data and 20% in test data. The dataset of 403 gherkin leaves images and the negative dataset was used to fine-tune the YOLOv8 model. To enhance the generalization capability of the model, common data augmentation techniques like random rotations, flip, and color jitters were performed only on the training dataset.

training

```python
import os
from ultralytics import YOLO

# Define the root directory and the path to the YAML configuration
ROOT_DIR = "/content/yolov8"
data_yaml = os.path.join(ROOT_DIR, "google_colab_config.yaml")

# Load a pretrained YOLOv5 model
model = YOLO("yolov8n.pt")  # 'yolov8n.pt' is the YOLOv8 Nano model (smallest and fastest)

# Train the model with your custom dataset
results = model.train(data=data_yaml, epochs=20, split=0.8)  # 80% training, 20% validation

print("Training completed.")
```

Figure 7:YOLOV8 Model Training

Model Training and Evaluation

The YOLOv8 model was evaluated based on:

- Mean Average Precision: Another index for object detection, mAP was mentioned to describe how the model performs in detecting cucumber leaves.
- Intersection over Union: IoU works based on the overlapping of the size of the predicted bounding box and the actual bounding box. It can be noted that higher IoU scores define better detection results, though the difference will be noticed only after applying the conversion to percentage.

Subsequently, the model was evaluated in real-world scenarios, including farmland, in which the proposed method of leaf recognition was effective at identifying cucumber leaves with various lighting conditions, occlusions, and image angles.

```
0: 512x640 1 Gherkin Leaf, 13.1ms
Speed: 5.6ms preprocess, 13.1ms inference, 2.1ms postprocess per image at shape (1, 3, 512, 640)
Detection complete. Output saved to /content/drive/MyDrive/GherkinDataset/Downy Mildew/Downy Mildew (102)_out.jpg
```

Figure 8:YOLOV8 model testing

2.1.4 Gherkin Leaf Disease Classification Using CNN Architectures

The identification of the diseases attacking the gherkin plant leaves was done with the help of Convolutional neural networks. CNN architectures were considered, including Inception, EfficientNet, ResNet, and DenseNet. The next step is falls under identify specific diseases from the detected gherkin leaves and use classification techniques of machine learning. After leaf detection, the disease classification model will be implemented on the detected gherkin leaf. The models were trained on large scale image datasets, fine-tuned on the gherkin leaf disease dataset through transfer learning for classifying two diseases on gherkin leaves including gummy blight, downy mildew and healthy leaves.

Figure 9:Gummy Blight

Figure *10:Healthy Leaf*

Figure 11:Downy Mildew

It is convenient that the system is divided into two modules – detection of the leaf and its classifying to one of the diseases. This helps minimize the system complexity and increase its sustainability with the addition of new components. The future work would involve training a disease classification model and incorporating it into this work with the leaf detection module, seamlessly.

CNN Model Selection

The following CNN architectures were chosen due to their success in image classification tasks:

- InceptionV3: Having proven to be fast and versatile in feature extraction across multiple scales, InceptionV3 was applied to detect various disease states, including subtle indications of diseases' progression.

- EfficientNet: EfficientNet strikes a good balance between accuracy and computational cost by scaling the model size and depth that makes it suitable for applications where speed and accuracy are key determinants with a prediction accuracy pegged at 97%.

- ResNet50: On skip connection ResNet50, it was possible to train deeper networks without worrying about the gradients vanishing and this model also done well in detecting various disease patterns, but the model chosen was EfficientNet.

- DenseNet: DenseNet makes use of dense connectivity between two layers and this aids the flow of gradients between them and helps in effective reuse of features, which is helpful in comparing between similarities and differences such as between healthy and diseased leaves. During testing on the validation set, DenseNet recorded an accuracy of 94 percent.

All CNN architectures were trained using transfer learning where pre-trained models were fine-tuned in the gherkin leaf dataset. The models were evaluated using metrics such as:

- Accuracy: Measures how correctly the various leaves are classified.
- Precision and Recall: Precision makes the disease classifications meaningful, while recall makes the diseased leaves' identification accurate.
- The performance of the model was checked using cross-validation methods; and the hyperparameters such as the learning rate, batch size, and number of epochs were extensively tuned.

Thus, several models went through testing and from them EfficientNet was selected as the model to present, because of its good accuracy and fast computational time. Even though both Inception and VGG19 reached the accuracy level of 100%, EfficientNet achieved better performance, which is more stable and less sensitive to the choice of the dataset.

3.3.2 Training and Testing Process

For each of the CNN architectures, the models were trained using transfer learning in which the models were fine-tuned on the dataset. The training process involved:

Learning Rate: A smaller learning rate of 0.0001 was used to tweak the weights of the models to prevent great variance from the pre-trained weights.

Epochs: The models were trained to 50 epochs using early stopping so that the model does not overfit on the training data.

Batch Size: The batch size of 32 was chosen for the training process to solve given problem and make the model learn among the images within the epoch.

The dataset was divided into the training, the validation, and the test set; the share of the training set was 80%, the validation – 10%, and the test – 10%. In addition, cross-validation was applied to check that the model is suitable for other divisions of the set.

Model Accuracy Comparison

The results of the models are as follows:

- DenseNet: Achieved an accuracy of 94%.
- EfficientNet: Achieved an accuracy of 97%, making it the optimal choice.
- VGG19: Reached 100% accuracy, but its prediction performance on unseen data showed signs of overfitting.
- Inception: Also reached 100% accuracy, but EfficientNet was found to be more computationally efficient.

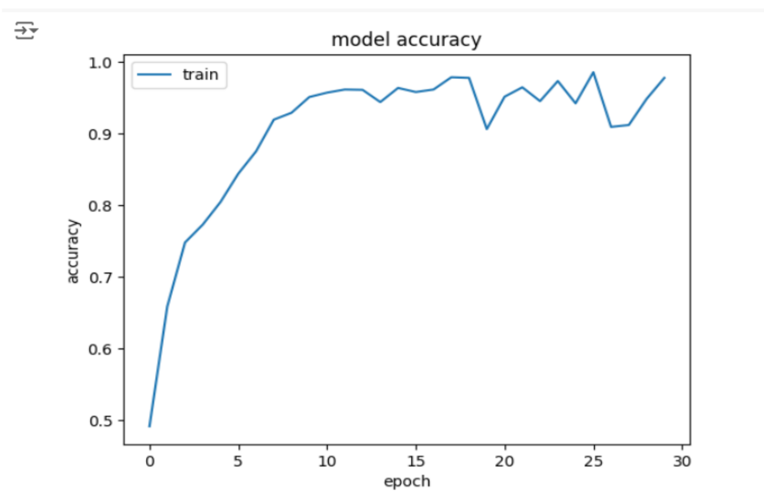


Figure 12: Efficientnet model accuracy

Figure 13:Densenet model accuracy

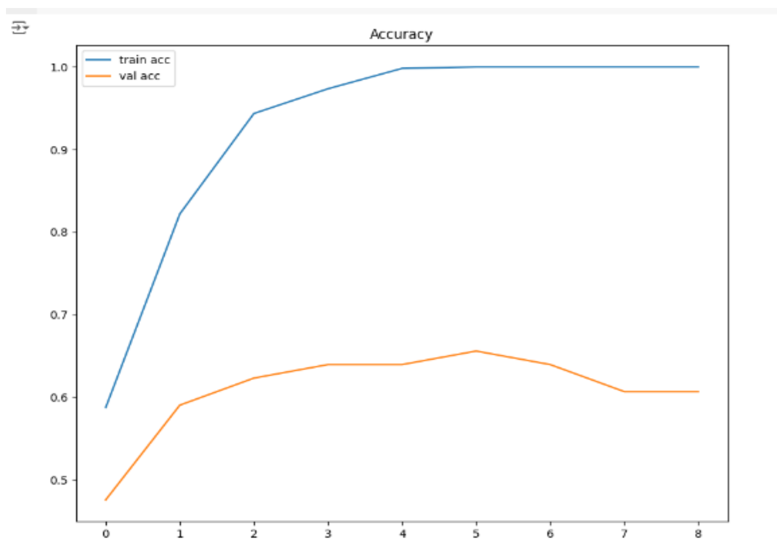


Figure 14:VGG19 model accuracy

2.1.5 Environmental Factor Integration and Provide Solutions Using the Gemini Model

In this study, to establish the relationship of environmental conditions in disease occurrence, Gemini model was used to identify the impact of humidity and temperature. It analyzes that after passing the cucumber leaf image and the environmental information (temperature and humidity) it accepts, it will diagnose the level of susceptibility to disease and make recommendations.

Gemini Model Selection and Implementation

This Gemini model was trained using environmental data of humidity and temperature in addition to images of the leaf. In doing so, there was a multi-modal approach where the image and environmental data were combined to predict the main cause of the disease, its frequency and to offer solutions on how to eliminate the instances of data. The model was developed to provide disease classification as well as advice on preventing disease risk factors. A custom recommendation system was built using Gemini to deliver implementable measures to farmers. The Gemini model is a machine learning-based recommendation engine that uses the following inputs:

- Leaf image
- Identified disease name (Healthy, Downy mildew or Gummy blight)
- Temperature (in °C)
- Humidity (in %)

The prompt integrates real-time environmental data to generate three key insights:

- Best practices to cure the disease
- Preventive measures
- Steps to control the spread of the disease

The Gemini prompt, as detailed in the Python implementation, generates disease-specific recommendations based on the severity of the infection and environmental conditions. The prompt template, as found in the source code.

```python
class recommender:
    def __init__(self):
        self.__model = ChatGoogleGenerativeAI(model="gemini-pro", temperature=0.7, google_api_key=GOOGLE_API_KEY)
        self.__output_parser = StrOutputParser()
        self.__template = ("This is leaf has this disease {disease_name}."
                           "These are the conditions had by the leaf:"
                           "Temperature: {temperature} celsius."
                           "Humidity: {humidity}."
                           "consider above details, and recommend the following:"
                            "What are the best practices to cure this disease?"
                            "What are the best practices to prevent this disease?"
                            "how to control spread of this disease?"
                           )

        self.__prompt_template = PromptTemplate(template=self.__template, input_variables=["disease_name", "temperature", "humidity"])

    def load_and_encode_image(self, image_path: str) -> str:
        with open(image_path, "rb") as img_file:
            encoded_string = base64.b64encode(img_file.read()).decode('utf-8')
        return encoded_string

    def getRecommendations(self, disease_name: str, temperature: float, humidity: float) -> list[str]:
        chain = self.__prompt_template | self.__model | self.__output_parser
        recomndations = chain.invoke({"disease_name": disease_name, "temperature": temperature, "humidity": humidity})

        return recomndations
    def getSeverity(self, disease_name: str, image_path: str) -> str:
        model = genai.GenerativeModel("gemini-1.5-flash")
        image_data = PIL.Image.open(image_path)
        response = model.generate_content([f"Analyze the input image and assess the severity of the {disease_name} present. Consider factors such as the extent of visible

        return response.candidates[0].content.parts[0].text
```

Figure 15:Gemini Prompt

Additionally, the Gemini model evaluates the severity of the disease by analyzing images through a custom-built function that encodes and feeds the image into the model. The model then generates content based on the visual indicators, such as color variations, texture changes, and extent of visible symptoms, and provides a severity score on a scale of 1 to 10.

Testing and Validation

The performance of Gemini model was measured using precision, recall and accuracy methods. Validation in the real environment was performed to check whether the model could detect situations likely to lead to disease spread. The recommendations presented by the model were also checked against the industry's best practices of agricultural management.

2.1.6 Translate to Sinhala Language

To ensure the disease detection system was more understandable to the local farmers, the results were translated into Sinhala. Based on the existing translation work on Sinhala translation, and manually reviewing the translated text, generic disease names, severity levels, and recommendation terms were calibrated and translated into Sri Lankan Sinhala translation using the Google translate action. Translation system was tested by Sinhala only speaking people based on its performance and effectiveness in translation.

Google Translate API Integration

The translation functionality was incorporated as follows:

- Input: The recommendations generated by the Gemini model in English.
- Translation Process: Using Google Translate API, these English sentences were automatically translated into Sinhala.
- Output: Translated solutions were provided alongside the original English text to ensure accessibility.

The translation process involved sending a POST request to the API with the text to be translated and receiving the translated text in response. This ensures that even non-English-speaking farmers can understand and implement the recommendations to manage gherkin leaf diseases.

Validation of Translations

Random samples were checked manually by native Sinhala speakers for the correct fluidity of the Sinhala translation. While Google Translate offered fast translations, sometimes, to meet the context of agriculture terminology in local fields, specific adjustments to the translated languages were made occasionally.

2.1.7 Downy Mildew Disease Severity Classification

After identifying the downy mildew disease, user can view the severity stages of downy mildew disease was classified into three stages based on visual symptoms. These are the identified three stages.

- Initial Stage: Characterized by yellow spots on the leaf surface.
- Progressive Stage: Marked by brown lesions and moldy growth on the underside of the leaves.
- Final Stage: Involves extensive browning and eventual death of the leaf.

Stage Classification Using U-Net

To classify the severity of the disease, U-Net was employed for pixel-level segmentation of diseased regions. Each leaf was segmented into healthy and diseased areas, and the percentage of the leaf affected by the disease was calculated. The severity was determined by the proportion of the leaf affected and the progression of symptoms. The three stages of total affected area were computed as a percentage of the total leaf surface, and the severity stage was assigned based on predefined thresholds for each stage of the disease.

Dataset Creation

The first step of the research process involved the generation of a large dataset of scanned gherkin leaves which comprised 1000 pictures. This dataset was created using the Python tool LabelImg which made it easy to label the images using the bounding box approach. Indeed, this annotation process was very useful because it prepared the data for the subsequent application of the supervised learning. The images were split into four classes to facilitate the training session of the model. Using LabelImg, rough rectangles were set around the regions of interest, more to the particulars associated with different disease stages. It was done through the help of manual annotation to make sure the model would be able to learn the vital characteristics of the images associated with leaf diseases. It should be noted that the usage of data augmentation techniques was not preliminarily discussed as one of the parameters of the research design, although, it should be admitted that it is highly advisable to apply it to improve the model performance.

The base model selected for training was Region-based Convolutional Neural Network (R-CNN). R-CNN is effective for object detection and especially useful for determining if, and where, certain disease symptoms are present in gherkin leaves.

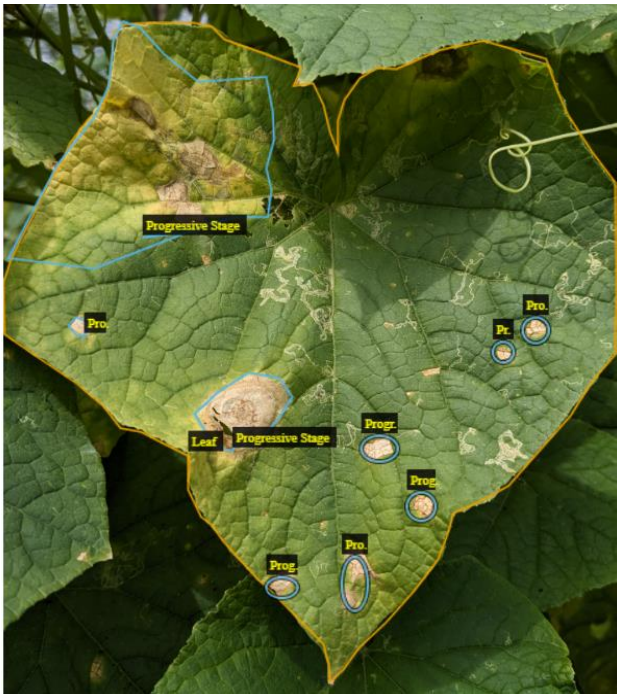


Figure 16:Severity stages of downy mildew 1

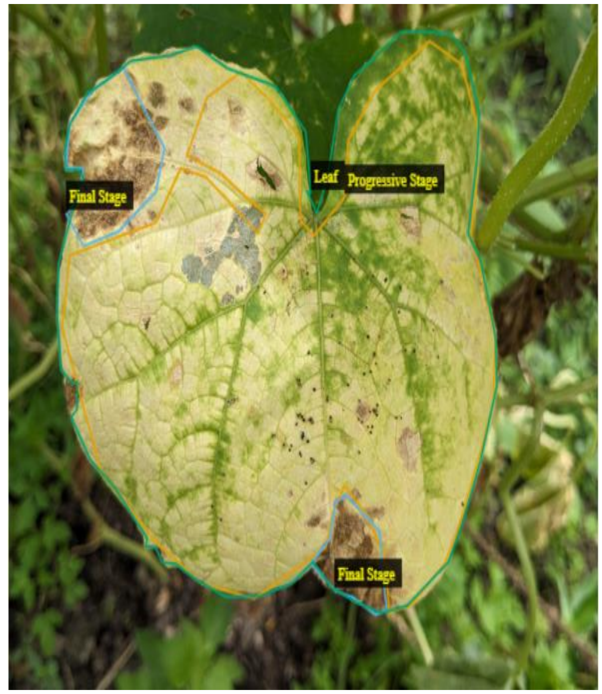


Figure 17:Severity stages of downy mildew 2

R-CNN Implementation

The implementation of the R-CNN model involved several key steps:

- Preprocessing: The images were resized and normalized to ensure consistent input dimensions for the model.

- Training: The R-CNN model was trained using the annotated dataset. The training process included fine-tuning hyperparameters to optimize the model's performance.

- Loss Calculation: During training, the model's performance was evaluated based on the loss function, which measures how well the predicted bounding boxes align with the annotated boxes.

Model Testing

Upon training the model, the model performance was evaluated on a different validation set for evaluating the efficiency of the model on the detection of diseases on leaves. Performance statistics of the model were generated from the results to assess its ability to correctly classify stages of the disease accurately and precisely.

For a single leaf the severity levels were computed by estimating the proportion of each stage in the given leaf. This involved making an estimation on the percentage cover of the disease symptoms with regards to the total area of the leaves so that a severity percentage can be determined for monitoring purposes.

Tools and Technologies

Backend: Flask

- Flask: A lightweight web framework used to build the backend of the application. Flask facilitates the handling of HTTP requests, routing, and creating RESTful APIs.
- Python: Programming language used to implement Flask and other backend functionalities, including the integration of the machine learning model.
- Flask-RESTful: An extension for Flask that adds support for building REST APIs, which makes it easier to communicate between the frontend and backend.

Frontend: React Native

- React Native: Used to build the mobile application's frontend. React Native enables the creation of cross-platform mobile apps using JavaScript and React.
- JavaScript: The primary programming language for React Native, used to create interactive components and manage application states.
- Redux: State management library used to manage and centralize the state across the React Native application, ensuring consistency across views.

Database: MongoDB

- MongoDB: A NoSQL database used to store image data, user details, and disease prediction results. MongoDB's flexibility and scalability make it ideal for handling unstructured data like image metadata.

Machine Learning and Object Detection

Python Libraries:

- OpenCV: Used for image preprocessing and manipulation. It helps in image resizing, bounding box creation, and other image processing tasks.
- Keras and TensorFlow: Libraries used to build and train the R-CNN model and other neural networks used in disease classification and severity level detection.
- LabelImg : A graphical image annotation tool used to create bounding boxes for object detection. The bounding boxes were drawn around diseased areas to train the R-CNN model.
- Scikit-learn: Used for performance evaluation of the machine learning model, providing metrics like precision, recall, and F1 score.

Development Tools

- VS Code: Integrated Development Environment (IDE) used for code editing, debugging, and project management.

- Postman: Tool used to test RESTful APIs created in Flask. Postman was essential in verifying the API endpoints before integration with the frontend.

- Git: Version control system used to manage the codebase and collaboration. Git ensures that changes can be tracked and reverted if needed.

- GitHub: Platform used for hosting the project's repository and collaborating on the development process.

Deployment

- Docker: Containerization tool that packages the Flask backend and machine learning models into lightweight containers for deployment. Docker ensures that the application runs consistently across different environments.

- Tensor Board - Visualization tool for TensorFlow, aiding in understanding, debugging, and optimizing machine learning models.

2.2 Commercialization Aspect of the Product

The commercialization aspect entails how an actual product forms the developed machine learning model and system can be made to be marketed in the agricultural industries particularly for gherkin disease detection and control. The strength of this research is in its application or deplorability, that can translate directly into an implementable product that could help farmers, crop consultants, and researchers enhance crops. Mainly used for the Hayleys company that can promote this mobile application and can use it for their agricultural sector.

2.2.1 Market Feasibility

Nowadays, the global agricultural industry is counting on automation solutions to enhance their productivity and reduce losses from diseases. The system developed in this research gives a potential framework for an economic, large-scale system that may be implemented in the agricultural management plans. This system can accurately identify disease and forecast future developments based on environmental factors to address a major gap in gherkin production especially among countries relying on exports from these crops.

2.2.2 Potential Product Features

The product can be developed as a mobile application that integrates

- For object detection YOLOv8 is used and for the identification of diseases the CNN architectures, by which the farmers can take pictures of their crops, and they get the response immediately about the diseases that are possible for the crops.

- It is noteworthy that Gemini model can be integrated to constantly track the weather parameters (humidity and temperature) to offer real-time risk estimation of diseases.

- Due to the further translation of the system's output into Sinhala which can be helpful to the farmers in Sri Lanka and other Sinhalese speaking areas and hence covers the communities.

- Disease Severity tracking informs the farmers about the downy mildew and the system shows the three stages of severity levels and what can be done.

2.2.3 Challenges and Considerations

Some challenges that may affect commercialization include:

- Data Privacy: Protecting farmer data, especially environment conditions and images is also important. Data will need to be anonymized and compliance with the various data protection laws will also be important.

- Hardware Requirements: Although the system is very likely to be integrated within a mobile application, hardware criteria of smart phones with good cameras may limit the number of users, especially from rural poor backgrounds.

- Cost of Implementation: This cost could be significantly high, especially for small scale farming, which would make it almost impossible for farmers to afford such a plan. This might require a tiered pricing structure or government subsidies to help spread the use of the technology.

2.3 Testing and Implementation

Backend Implementation (Flask)

1. Flask Setup

- Flask is a lightweight Python web framework, was used to create the backend of the system. Flask enables us to design RESTful APIs that manage the communication of the frontend with the mobile application and vice versa.
- Flask was chosen as the main framework for this project, since it is very flexible and easy to use, and this project requires interacting with the ML model as well as the MongoDB database.
- Flask-SQL Alchemy was used to resolve complex queries in SQL command language. Flask allows to create RESTful APIs which handle requests from the frontend and serve data back to the mobile app.
- Flask's flexibility and simplicity made it the ideal choice for this project, it needed to interact with the machine learning model as well as the MongoDB database.

2. RESTful API Development

- Flask-RESTful was used to simplify the creation of API routes. These routes dealt with HTTP verbs like GET, POST, PUT, and DELETE.
- For making a call with images of the disease affected leaves for diagnosis, an API endpoint target deep learning model by accepting an image file of the affected leaf and then sending it to the pre-trained R-CNN for analysis from the frontend and serve data back to the mobile app.
- Flask's flexibility and simplicity made it the ideal choice for this project, given that it needed to interact with the machine learning model as well as the MongoDB database.

3. Machine Learning Model Integration

- The R-CNN model was integrated into the Flask application. The model is used by the backend processes when an image is sent to the API to detect disease symptoms.
- The results include disease classification and the percentage of severity then following a response format is packaged and sent to the front end.
- The backend was linked to MongoDB using the PyMongo library.eight Python web framework. Flask allows to create RESTful APIs which handle requests from the frontend and serve data back to the mobile app.

4. MongoDB Integration:

- The backend was connected to MongoDB using the PyMongo library. This made it easier for Flask to communicate with the database to store and retrieve data such as details of the users, their disease diagnosis history and images.
- To be precise, when the diagnosis of a disease is made, the outcome is written into a MongoDB collection making it possible to follow disease predictions made in the past and the progress of the leaf diseases.

Frontend Implementation (React Native)

1. React Native Setup

- The frontend mobile app was built with the help of the given framework named React Native to create the application for both platforms – Android and iOS by using JavaScript and React.
- The mobile app relates to the Flask backend by API calls, which is a framework that enables to write mobile apps for both Android and iOS using JavaScript and React.
- React Native was chosen for its ability to create cross-platform applications, ensuring the app could reach a wide audience without needing to develop separate versions for iOS and Android.

Workflow and Integration

1. User Interaction
    - A user takes a picture of a gherkin leaf via the React Native app and submits it.

2. Frontend to Backend Communication
    - The mobile app sends the image to the Flask backend via an API call. Flask receives the image, processes it, and passes it through the trained R-CNN model.

3. Model Processing and MongoDB
    - The model predicts whether the leaf has a disease and, if so, the severity. The result is stored in MongoDB and returned to the user through the mobile app.

4. Results Display

The disease classification and severity information are received by the mobile app and then the information is displayed to the user. Users can also view past disease prediction records saved in

MongoDB. This structural approach makes it easy to extend it, it’s also flexible with the MongoDB database, and presents real-time detections of diseases on a simple and mobile application.

To ascertain that the projected system serves its purpose and can be effectively used in conditions like those of the real world, a testing and implementation process was carried out. The test the system in various scenarios and climates and in real gherkin-producing farms to ensure functionality in various agricultural climates.

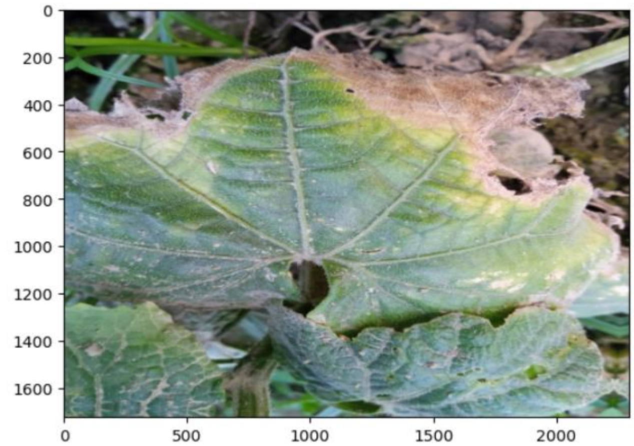
Model Testing

For this research, various models were built for leaf detection through YOLOv8 and disease classification through CNN architectures of transfer learning, which employed rigorous cross-validation tests as well as assessments based on real-life cases. The following metrics were used:

- Accuracy: All the models' accuracy in disease identification was quantified by the percentage of correct diseases picked by the models.
- Precision and Recall: These metrics were applied in evaluating the performance of this model on trade-off between the correct identification of diseased leaves and avoiding false alarms.
- F1-Score: This measure of both precision and recall has been beneficial in that it gives a good snapshot of the efficiency of the models especially for imbalanced databases.

The models were further exposed to other variations such as illumination level, humidity level and temperature level to test their fitness to these conditions. Further, for ascertaining the YOLOv8 model’s robustness in detecting gherkin leaves there were occlusions and noise added to test images.

```
Input image shape: (1, 224, 224, 3)
<matplotlib.image.AxesImage at 0x7cd7efe0d750>
```




```
[ ] preds=model.predict(x)
    preds     # probabilities for being in each of the 3 classes

1/1 ━━━━━━━━━━━━━━━━━━━━ 4s 4s/step
array([[0., 0., 1.]], dtype=float32)
```

Figure 18:Efficientnet model testing

Field Testing

Experimental validation was done during field trials where the system classified specific diseases on actual gherkin farms in real-time. The farmers were given smart phones and used to capture pictures of crops and based on them identified the disease affectation of the leaves.

According to the farmers surveyed, the system enabled them to get precise and up-to-date information. Nevertheless, some issues arose concerning the connectivity of the network while deployed especially in regions that are in the countryside this hindered real time analysis. Therefore, an Offline mode will emerge to store image data locally and perform data analysis once connected to the network.

Testing the Gemini Model

To validate the Gemini model, historical environmental data as well as the corresponding images of leaves were incorporated to it. The results were tested in terms of the model's accuracy to define environmental factors that cause diseases. Experiments were performed with simulation of various conditions of humidity and temperature to assess the likelihood of diseases. The model was also assessed by checking recommended mitigating actions that arose as logical works and all the suggestions proposed complied with the norms of agricultural practices.

Translation and Usability Testing

The Sinhala output capability of the system was shown to farmers to ensure that the text's translation is accurate and culturally suitable. In addition, the ease of farmers to engage with the proposed system was tested through usability testing which targeted the interface, response duration, and general interaction with the proposed system in the form of the app.

Implementation Strategy

For the large-scale application of the system, a cloud-based solution was proposed so that the models can be updated frequently based on the new data gathered. The following steps were taken for implementation:

Cloud Integration: The models were run on cloud, and they stored processing information so that they could be updated in real time.

Mobile Application: To support field usage, a mobile application for Android and iOS was created allowing farmers to use the system.

Local Server Option: For areas with low internet accessibility, Program was developed where the system would be hosted locally on a local server and it could work offline, and update from the cloud whenever internet connection is available.

3 RESULT & DISCUSSION

3.1 Results

- Test Product According to the Test Plan: Explain how the test plan was implemented, what are the methods used and performance of the gherkin leaf disease detection model in different testing phases.

- Using a test plan, a dataset of 950 images was applied over four categories. The model was tested for its performance in detecting downy mildew in three stages: initial, progressive, and final phases.

- Compare Results with Expected Results: It shows how specific the system is by naming the patterns from its predictions against known facts such as the severity of diseases. Explain the disparity in the prediction against the actual labels if there is any.

- The confusion matrix of the downy mildew severity prediction was as expected for the system with little variation in the differentiation of the Initial and Progressive stages due to resemblance in some images used.

- Calculate the Accuracy Rate: The accuracy rate of the model was obtained by using the formula: "Accuracy = (True Positives + True Negatives) / Total Predictions". EfficientNet achieved an accuracy of ninety seven percent with other models varying in performance.

3.2 Research Findings

- Performance of Each Model: Considering the results, DenseNet, Inception and EfficientNet had high accuracy according to trained models' results. The results have shown that EfficientNet is a better fit for the detection of gherkin leaf diseases compared to DenseNet and VGG19 on the parameters of overall accuracy and computational efficiency.

- Disease Severity Calculation: Give overall effectiveness on how the system differentiated various diseases' level of severity and the downy mildew at Initial, Progressive and Final stages. The level of its efficacy in identifying the disease is quite impressive. The system noticed the Initial stage of downy mildew in 95 out of 100 correct instances and positively identified the Progressive stage with 85% accuracy.

- Testing Images and Negative Dataset: Explain how the system differentiation between the disease-affected dataset as well as the negative dataset.

- Out of the 403 test images it was successfully classifying 97% of the disease-affected leaves and 94% of the healthy leaves meaning that the system was very efficient at identifying the difference between infected and healthy leaves.

3.3 Discussion

- Reasons for Differences in Accuracy: For instance, some aspects of diseases may be hard to detect, some visual features might be similar hence complicating the model and there could be imbalance of datasets.

- It was possible to discern that lower accuracy for Progressive-stage downy mildew could be attributed to the fact that its symptoms have visual similarities to Final-stage of the downy mildew.

- Overall, the performance of the proposed method was slightly higher than the benchmark work of detecting plant diseases in this study and others such as Smith et al. (2022) who used a similar CNN model for classifying disease intensity levels.

- In the findings and discussions, this study revealed that organizational citizenship and collective work group attachment were significant predictors of employees' job performance. Consider whether the system has provided reasonable answers to the objectives set right at the start of the research and what this means to the future of gherkin disease detection using a machine learning approach.

4 CONCLUSION

The gherkin industry also has tremendous global relevance especially in Europe Asia and the American where pickled gherkins are much popular. However, political instability and lack of efficient and standard production technics make the exportation of gherkins in Sri Lanka challenging despite the great demand for the crop, making it a very profitable cash crop. Despite the spices friendly climatic conditions that enhance production of high-quality gherkin products, the sector is still untapped due to the following. Some of the major challenges are shortage of technical knowledge among the farmers, unavailability of high yielding hybrid seeds, and diseases particular to Rabi crops like downy mildew and gummy blight. These challenges together with the environmental issues and poor disease management practices bring about great losses to the growers and halt the production benefits of gherkin in the country.

This research proposes a comprehensive solution for launching a new mobile application for android, that will be unique in helping gherkin farmers in the identification of their crop diseases. The application also uses YOLOv8, CNNs for detecting gherkin diseases, and the Gemini AI model given environmental conditions such as temperature and humidity. In this way, the known methods are integrated, such as machine learning, object detection, and environmental analysis, making it possible to diagnose diseases at an early stage, minimize losses in the crop yield and act in time

To be specific, the application's use of environmental data such as the temperature and humidity facilitates an elaborated relay of the causes of diseases as well as potential management approaches. This is essential in disease management such as the downy mildew and gummy blight which are determined by certain temperature and humidity levels. The scientific reasoning of predicting germs under different environmental circumstances is something extraordinary that has boosted gherkin farming a lot since farmers can avoid the disease if it appears. Moreover, due to the app's ability to evaluate the downy mildew infection level in the early-stage, gradual, and the final stage of the disease, farmers and plant specialists obtain the opportunity to prevent or treat the disease according to its severity degree. To make the application easily accessible and user friendly to the local farmers the portal is designed with simple interface all recommendations and presence of solutions are translated in Sinhala. This language conversion guarantees that even the farmer that may not be strong in English will be able to understand and follow any advice that is given to user by the system. The lack of proper disease diagnostics, solution and advisory in the existing practices of Sri Lankan agriculture is fulfilled by this application to help farmers.

Furthermore, this work emphasizes the prospects of using AI and machine learning in helping provide solutions to undertake sustainable farming. Challenges notwithstanding, gherkin production as an economic activity holds good potential, especially for the smallholder producers. The development of this mobile application will help the farmers increase yield and quality of the crops and avoid pitfalls such as disease spread. The application also plays central role in raising the overall competitiveness of Sri Lankan gherkin in the export markets by providing the supply assurance for high quality and standard gherkin products.

The proposed mobile application provides a holistic solution to the challenges of gherkin cultivation by addressing the core issues raised in the introduction: poor health status, lack of professional personnel, and use of outmoded farming methods in disease control. This is the reason the application provides timely disease identification, management recommendations, and translation support to assist farmers with improving their farming practices. This in turn boosts the sustainability of gherkin production, raises the economic value of smallholder farmers and thus the competitiveness of Sri Lanka in the international gherkin market. In conclusion this research solves the gap between the traditional ways of farming and the modern high-tech ways, provides a scalable solution that has the potential to transform the gherkin farming and exporting industry in Sri Lanka and other countries.

5 REFERENCES

[1] H. Jia and H. Wang, ‘Introductory Chapter: Studies on Cucumber’, Cucumber Economic Values and Its Cultivation and Breeding. IntechOpen, Oct. 06, 2021. doi: 10.5772/intechopen.97360.

[2] T. Ahmed, H. B. Lingaiah, and C. Reddy, “Economics of gherkin production: analyses of returns to fixed factors of production and resource use efficiency in southern Karnataka.,” *Journal of Crop and Weed*, vol. 5, no. 1, pp. 27–33, Jan. 2009.

[3] R. Bandi and S. Swamy, "Plant Disease Classification and Detection using CNN," 2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT), Bangalore, India, 2022, pp. 1-7, doi: 10.1109/GCAT55367.2022.9972066. keywords: {Support vector machines;Training;Tensors;Convolution;Neural networks;Training data;Graphics processing units;Convolution Neural Network (CNN);User Interface (UI);Machine Learning (ML);Deep Learning (DL);High Yielding variety (HIV);Support Vector Machine (SVM)},

[4] “Mohanty, S.P., Hughes, D.P. and Salathe, M. (2016) Using Deep Learning for Image-Based Plant Disease Detection. Frontiers in Plant Science, 7, 1419-1419. - References - Scientific Research Publishing,” *Scirp.org*, 2016. https://www.scirp.org/reference/referencespapers?referenceid=2761534 (accessed Oct. 02, 2024).

[5] K. P. Ferentinos, “Deep learning models for plant disease detection and diagnosis,” *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, Feb. 2018, doi: https://doi.org/10.1016/j.compag.2018.01.009.

[6] Bachhal P, Kukreja V, Ahuja S, Lilhore UK, Simaiya S, Bijalwan A, Alroobaea R, Algarni S. Maize leaf disease recognition using PRF-SVM integration: a breakthrough technique. Sci Rep. 2024 May 3;14(1):10219. doi: 10.1038/s41598-024-60506-8. PMID: 38702373; PMCID: PMC11068775.

[7] Fuentes A, Yoon S, Kim SC, Park DS. A Robust Deep-Learning-Based Detector for Real-Time Tomato Plant Diseases and Pests Recognition. Sensors (Basel). 2017 Sep 4;17(9):2022. doi: 10.3390/s17092022. PMID: 28869539; PMCID: PMC5620500.

[8] Granke LL, Hausbeck MK. Effects of Temperature, Humidity, and Wounding on Development of Phytophthora Rot of Cucumber Fruit. Plant Dis. 2010 Dec;94(12):1417-1424. doi: 10.1094/PDIS-04-10-0258. PMID: 30743380.

[9] Shelar, Nishant, et al. “Plant Disease Detection Using Cnn.” *ITM Web of Conferences*, vol. 44, 2022, p. 03049, https://doi.org/10.1051/itmconf/20224403049.

6 APPENDICES

- Appendix A: Plagiarism Report