VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI – 590 018

A PROJECT REPORT ON

"COTTON PLANT DISEASE DETECTION USING DEEP LEARNING"

Submitted in the partial fulfillment of the requirement of the award of degree of

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE & ENGINEERING

Submitted By:

SHRIVATSA G [4VV19CS152]

SHRISHA P [4VV19CS151]

VIKHYAT KOPPAL [4VV19CS180]

NAVEEN H N [4VV20CS412]

UNDER THE GUIDANCE OF

Dr. Chethana H T

Assistant Professor
Department of CS&E
VVCE, Mysuru

VVCE

2022-2023

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
VIDYAVARDHAKA COLLEGE OF ENGINEERING
MYSURU - 570 002

VIDYAVARDHAKA COLLEGE OF ENGINEERING

Gokulam 3rd Stage,

Mysuru-570002

Department of Computer Science and Engineering

VVCE

CERTIFICATE

This is to certify that the project report entitled **"Cotton Plant Disease Detection Using Deep Learning"** is a bona fide work carried out by **Shrivatsa G** (4VV19CS152), **Shrisha P** (4VV19CS151), **Vikhyat Koppal** (4VV19CS180), **Naveen H N** (4VV20CS412) students of 8th semester Computer Science and Engineering, **Vidyavardhaka College of Engineering, Mysuru** in partial fulfillment for the award of the degree of **Bachelor of Engineering** in **Computer Science & Engineering** of the **Visvesvaraya Technological University, Belagavi**, during the academic year **2022-2023**. It is certified that all the suggestions and corrections indicated for the internal assessment have been incorporated in the report deposited in the department library. The project report has been approved as it satisfies the requirements in respect of project work prescribed for the said degree.

Signature of the Guide	Signature of the HOD	Signature of the Principal
(Dr. Chethana H T)	(Dr. Pooja M R)	(Dr. B Sadashive Gowda)

Name of the Examiners **Signature with Date**

1)

2)

ACKNOWLEDGEMENT

If words are considered as the tokens of acknowledgements, then the words play the heralding role of expressing our gratitude.

With proud gratitude we thank God Almighty for all the blessings showered on us and for completing our project successfully.

We owe our gratitude to The Principal, **Dr. B Sadashive Gowda** for his whole hearted support and for his kind permission to undergo the project.

We wish to express our deepest gratitude to **Dr. Pooja M R,** Head of Department, Computer Science and Engineering, VVCE, for her profound alacrity in our project and his valuable suggestions.

We wish to enunciate our special thanks to our paradigmatic and relevant Project coordinators **Dr. Janhavi V** Associate Professor, **Dr. Ayesha Taranum** Associate Professor, **Prof. Usha C S** Assistant Professor, **Prof. Nithin Kumar** Assistant Professor, **Prof. Gayana J Kumar** Assistant Professor and internal guide **Dr. Chethana H T,** Assistant Professor in Department of Computer Science and Engineering, VVCE who gave us throughout our project period with their valuable suggestions and for devoting their precious time in making this project a success.

In the end, we offer our sincere thanks to our family members and friends for their valuable suggestions, encouragement and unwavering support.

Shrivatsa G (4VV19CS152)
Shrisha P (4VV19CS151)
Vikhyat Koppal (4VV19CS180)
Naveen H N (4VV20CS412)

ABSTRACT

Most of the farmers cultivate cotton in large numbers but the cotton leaf disease is the major problem in the past few decades and that results in a loss of crops, their productivity and money as well. The Cotton leaves are majorly affected by the diseases named “Leaf Lesions”, “Bacterial Blight”, “Curl virus”, “Fusarium wilt”. We will develop a frontend application that can take both the uploaded image or the live image from the camera app. Then we make use of Convolutional Neural Network (CNN) a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other.

The model is trained with large number of training datasets that is collected from both website and live from the fields. We classify the cotton plant into five different categories named “Leaf Lesions”, “Bacterial Blight”, “Curl virus”, “Fusarium wilt” and “Healthy”. A machine learning approach based on digital cotton leaf disease classification and retrieval can be achieved by extracting features from its leaf image. We classify the class of leaf diseases based on the crops features and then provide particular cure to the plant disease containing the name cost and description of the pesticide within the App. The App also displays major driving factors for cotton production and major cotton varieties in India. In this investigation crops are classified on the foundation of color. Overall, using machine learning to train the large data sets available publicly gives us a clear way to detect the disease present in plants in a colossal scale of a few million bales annually.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In India, Cotton is one of the most important cash crops. Cotton diseases are the main reason for the reduction in the productivity of the cotton. With reference to that, crop disease detection is the subset of smart farming. Crop disease detection plays an integral role in the development of the crops. India is known for its agricultural process, which means that majority of Indian citizens engaged in the agriculture sector. Cotton is a significant contributor to the economy. Various fungal and bacterial diseases have contaminated the majority of the plants. Plant disease is caused by climatic conditions, these diseases, due to evolution have evolved to such an extent where organic fertilizers have become feeble/weakened to them. Thus, High Quality fertilizers have to be used to protect the crops. Crop losses have had a huge impact on overall GDP of the country and almost 43% of crop losses occur due to these infections caused by pests and insects. In present era, the approach used for detection by farmers is through naked eye. It requires continuous observation and monitoring which results into loss of time and it is expensive [15].

In some regions, farmers need to go various places to take guidance from the experts. Henceforth, automatic prediction of various diseases on the crop will provide real time benefits to the farmers that will save time, money, and life of the crop. To tackle the increasing inconvenience caused to farmers, a machine-learning model, which can predict occurrence of cotton crop disease based on the structure and quality of leaf. Features play an important role in the classification process. Previous proposed works for detecting disease has some limitations such as low resulting accuracy and a smaller number of images used todetect disease. When they are infected by diseases, there is a change in shape, size and color. Transformation in the crop color is the imperative feature for the notification, when the health of the crop is in good stage. Then the color of the crop is dissimilar but as soon as the crop is going to be affected by some harming pathogens, the color transforms automatically. Crop diseases have turned into a dilemma because it may cause a diminution in productivity. These symptoms can be checked manually but not in the proper amount. Hence various image processing methods detect diseases on plant leaf and stems. Using image processing techniques exact disease can be identify based on color, texture or shape methods available such as k-means clustering, Canny and Sobel segmentation, and Otsu thresholding. Techniques such as Support Vector Machine (SVM), all these proposed works for detecting disease have some limitations such as low resulting accuracy. To increase plant productivity and economic process, detection and classification of plant diseases are necessary tasks. It is necessary to detect disease and spray pesticides properly on crops. We make use of CNN model for Disease prediction and provide suitable pesticide for the disease as a cure. The app also displays trading factor that contains most important drivers for cotton prices, majorvariety of cotton plants in India according to their trading name and description about them and also makes use of a prediction model to displays production of cotton in bales in future years [16].

Talking about cotton crop cultivation in India, Gujarat is the biggest cotton-growing state. Cotton leaf diseases are the main reason for the reduce the productivity of the cotton. About 80-90% of diseases such as Alternaria leaf spot and Bacterial blight mainly occur on the leaf of the cotton plant. Various image processing concepts such as image filtering, segmentation, image feature extraction has emerged to detect the leaf diseases. There are various image segmentation methods available such as k-means clustering, Canny and Sober segmentation, and Otsu thresholding. Techniques such as Support Vector Machine (SVM), Neural Network (NN), and Homogeneous Pixel Counting technique for Cotton Diseases Detection (HPCCDD)can be used for classification. Features played an important role in the classification of the process. Previous proposed works for detecting disease has some limitations such as low, resulting accuracy and a smaller number of images used to detect disease. Artificial Intelligence is helping us in all aspects. The proposed system is mainly used to develop an application which recognizes cotton leaf diseases. To solve agricultural problems using Artificial Intelligence, a cotton plant disease prediction which will help to predict the disease of the cotton crop and tell the farmers how to cure it [16].

1.1 Aim

The main aim of the project is that, we will be taking the datasets from Kaggle in which the datasets will contain the images of diseased and healthy plant leaves. We will developa frontend application that can take both the uploaded image or the live image from the camera app. Then we make use of Convolutional Neural Network (CNN) a Deep Learningalgorithm which can take in an input image. We classify the class of leaf diseases based onthe crops features into four major category of diseases and healthy and then provide particular cure to the plant disease.

1.2 Scope

- The proposed system is able to classify the given image into a diseased or a healthy cotton plant.
- It is able to give an accuracy of 95%.
- It is able to find out about the exact type of cotton leaf disease.
- It displays major cotton diseases in India along with Trading factor that explains methods for maximum production of cotton, major cotton varieties in India and also production of cotton in bales for future years.

1.3 Motivation

Cotton is the most important cash crop in India. It is also known as “White Gold” or “The King of fibers” among all cash crops in the country. About 80-90% of the diseases which occur on the leaves of cotton are Leaf Lesions, Bacterial Blight, Curl virus, Fusarium wilt. Detecting these diseases with bare eyes increased the complexity of cotton crops productivity which decreased the accuracy in identification precision. Even an expert would fail to assess and diagnose the diseases with their bare eyes, and this inadequate technique leads to more wastage of cotton crops. Due to these mistaken conclusions, most of the time, certain unnecessary pesticides which badly affect healthy cotton are applied leaving the farm.

For even a short time interval without production will affect overall nation GDP and farmers economy. Cotton farmers are usually worried about cotton prices in the market and are not aware of important drivers affecting its prices. Farmers also need knowledge of major cotton varieties grown in India that are suitable for soil in India.

1.4 Organization of the Report

The sequel of pages and their hierarchical arrangement play a pivotal role in structuring the project report properly and interlinking the vital elements of the report in the best possible format. This project report consists of 8 chapters as mentioned below:

- **Introduction:** provides the background information about the project and the basic idea of what this project is expected to do.

- **Literature Survey:** gives the detailed study of all the existing systems and its disadvantages. System requirements*:* talks about the detailed description on system requirements including both hardware and software.

- **System Analysis** provides a detailed description about the system analysis, why is it required, method of analysis of existing system, proposed system and its components.

- **System Design:** It is involved in giving a description on how the system is going to be designed, how exactly the system would be developed.

- **System Implementation:** It is all about the implementation part of the project that describes the critical coding of the project.

- **Testing:** gives information about testing the project in the real time scenarios and determines the efficiency of the system.

- **Snapshots:** It consists of snapshots of software and hardware modules.

- **Conclusion & Future Scope:** It includes the extensions that could be made to this project. References*:* consists of the papers, books, and websites we have referred to.

CHAPTER 2

LITERATURE SURVEY

Literature survey or review which combines both summary and synthesis of specific conceptual categories. Literature survey gives conclusion about how one can analyze and understand gaps exist in how a problem has been researched to date.

2.1 Survey Papers

A literature review survey, scholarly articles, or any other resources which are relevantto our area of interest in the research provides a brief description and critical evaluation of works that are related to our research problem. Literature survey provides an overview of sources that we have explored or referred to during our research.

- J. Karthika et al. proposed disease detection in cotton leaf spot using image processing**.** The common observation to discover disease is thru clean eye observation by consultants.But, to safeguard the plant from obtaining infected, a quick and early detector is needed. to observe the massive fields of crops, mechanically police work the symptoms of plant diseases becomes vital thus we'd like automatic, fast, correct, and fewer costly methodology to discover plant diseases by analyzing leaf pictures. The system is interfaced with a digital/web camera which can alter the farmers to require pictures of plant leaves. victimization image process algorithms, the captured pictures square measure analyzed to discover the symptomsof illness and to live the disease sort. The knowledge relating to the kind of the illness is goingto be notified to the farmer through the GSM interface. supported the infected illness, therelay activates the pump put in within the device to unleash drugs to the infected plant. Classification of diseases are Bacteria blight, black arm spot, leaf spot is identified by Multi- class SVM. This is for cotton disease can be identified by multi SVM, classifies disease Black spot arm. If both are not the same for their manner it sends the image and respected problemto the farmer[17].

- Preetha S et al. proposed agriculture is a significant source of income for Indian people. Experts do the manual method of detecting disease in a plant. For this, a large team was required, and continuous monitoring was required that was a complicated task when we do this with a large number of crops. In some places, farmers were unaware of the experts, and they do not have proper facilities. In this paper authors proposed a system that uses Color transformation structure RGB is converted into HSV space, Masking and removing of green pixels with pre-computed threshold level, segmentation then texture parameters are compared to texture parameters of normal leaf. Remote sensing is important to the source of agriculture information. Remote Sensing methods are fast and superior. While working with agriculture areas with remote sensing, the important parameters that should keep in mind are characteristics of the agriculture field, different crop types, and differences in their feature characteristics such as shape and texture. Computer and software play an important role in the detection and classification of leaf diseases. For leaf disease detection, there are lots of image processing and pattern recognition techniques that can be used. The key to preventing agricultural loss is leaf disease detection[13].

- Shima Ramesh Maniyath et. al. proposed crop diseases are a noteworthy risk to sustenance security, however their quick distinguishing proof stays troublesome in numerous parts of the world because of the nonattendance of the important foundation. Emergence of accurate techniques in the field of leaf-based image classification has shown impressive results. This paper makes use of Random Forest inidentifying between healthy and diseased leaf from the data sets created. In this paper authors proposed a system that uses Random forests are as a whole, learning method for classification, regression and other tasks The histogram of oriented gradients (HOG) is an element descriptorutilized as a part of PC vision and image processing for the sake of object detection. HOG is a feature descriptor used for object detection. In this feature descriptor the appearance of the object and the outline of the image is described by its intensity gradients. The disadvantage ofthis is Random Forest requires much computational power as well as resource as it builds numerous trees to combine their outputs. The histogram of oriented gradients (HOG) is an element descriptor utilized as a part of PC vision and image processing for the sake of object detection [10].

- Nikita Yadav et al. proposed detection of crop disease is very important topic for analysis. It is one of the issues that cause reduction in quality and quantity of crop. So, detection and classification of crop disease is necessary task to increase crop productivity and economic process. The proposed research work is to analyze various machine learning and image processing techniques applied to detect crop disease. In this paper we will review, different machine learning techniques, such as Random Forest, Decision tree, Multilayer Regressor, Regression algorithms, image processing techniques, Extreme Machine Learning to get better accuracy for system. Here, crop leaf images are takenas input and after processing that image, it will detect whether there is any disease or not. If disease is detected, then it will tell what type of disease it is and will provide solutions such as pesticides or chemicals to cure that disease. It will increase the productivity and economic process [14].

- Paramjeet Singh et al. proposed cotton plant disease prediction using support vector machine (SVM). Support vector machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane [16].

- Azath M et al. proposed in this classification the feature extraction is performed by splitting the input image into pixels values, then set the color variance and quantization value of edge detector. But in existing method we performed only the canny edge detector and color splitting only not check the individual pixel wise orblock wise color variance, text variance. Simple edge detection is carried out and blocks with edge pixels inside are judged into the structural category. Then, color variance is calculated in the remaining blocks. Find the variation across the edge with canny edge detector and color splitting methods. Variations in the gray level in a region in the neighborhood of a pixel is a measure of the texture [18].

- Vani Rajasekar et al. proposed decision tree learning is supervised learning approach used in statistics, data mining and machine learning. In this formalism, a classification or regression decision tree is used as a predictive model to draw conclusions about a set of observations. Tree models where the target variable can take a discrete set of values are called classification trees; in these treestructures, leaves represent class labels and branches represent conjunctions of featuresthat lead to those class labels. Decision trees where the target variable can take continuous values (typically real numbers) are called regression trees. Decision trees are among the most popular machine learning algorithms given their intelligibility and simplicity. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making [1].

- Shewtha Kumari et al. proposed enhanced particle swam optimization method adopts the skew divergence variance features method applied for feature extraction from diseased cotton leaf images. Feature extraction is performed by splitting the input image into pixels value, then set the color variance and quantization value of edge detector. Existing methods work only with Canny and Sobel edge detector and color splitting (CMYK), it will not check the individual pixel wise or block wise of Edge, color and texture variances. Initially a simple edge detectionis carried out and blocks with edge pixels inside are judged into the structural category. Then, the color variance is calculated in the remaining blocks [16].

- Pravin Srinath et al. in 2022 this study examines the algorithm's suitability for detecting cotton leaf diseases in actual environments utilizing transfer learning and an object identification method called Mask RCNN. The model's accuracy during training is found to be 94%, and as the number of optimized iterations increases, the total loss value steadily decreases [2].

- Smruti Kotian et al. in 2022 proposed to develop a transfer learning system for Python, TensorFlow, the Open CV Library, and the computer environment Jupyter are utilized to create a machine learning model. The KNN machine learning method is the most effective model for identifying various cotton illnesses. To improve the performance of the model, the epoch count is important. For classifying the diseased leaves into various diseases and for detecting the diseased leaf, Transfer Learning and KNN have achieved accuracy rates of 95% and 86%, respectively [3].

- Rehan Sarwar et al. in 2021proposed to identify diseases on leaves in real-world settings, this article performed object identification on photographs of cotton leaves taken. On the CCL Dataset, which contained 700+ cotton leaf field photos with four different classes, three of which belong to illness and one of which belongs to healthy leaves, InceptionV2 is chosen in object detection using Faster R-CNN. Due of the limited number of images, transfer learning was essential in the CCL Dataset training process. 0.01275 is the computed loss on this dataset, and 0.871, or 87.1%, is the mean average precision[4].

- Bhagya Patil et al. in 2021 gave a view of cotton healthy and unhealthy leaf image classification using different machine learning algorithms with WEKA. For training and testing, this dataset contained nearly 3000 images of two different classes of healthy and unhealthy leaves. For image segmentation, a modified factorization-based active contour was used. Two different texture and colour features were extracted from segmented images and used for training and testing by various machine learning algorithms such as RF, NB, SVM, Ada Boost, KNN, and MP. Other classifiers were outperformed by the multilayer perceptron, which achieved a classification performance of 92.69% [5].

- P.R. Rothe et al. in 2015, proposed a pattern recognition system for three cotton leaf diseases identification and classification. This cotton dataset was collected using different cotton fields in natural condition. To remove noise in the images, pre-processing was done by applying Gaussian filter and image segmentation was done by using active contour model. Different features were extracted after image pre-processing and segmentation. Adaptive neuro fuzzy was used to train dataset for the classification of cotton leaf diseases and gave an accuracy of 85% [6].

- Bhushan Patil et al. in 2021 proposed over the IoT-based system camera and sensor installed on crop fields and contaminated survey regions were used to capture all of the photographs during daily surveys. Smart agriculture using an IoT platform with sensor mixing automates the irrigation system by monitoring field conditions anywhere to make the system effective for improving crop production [7].

- Jayraj Chopda et.al.in 2018 proposed captured with the aid of variables like temperature and soil moisture, the suggested method may predict cotton crop illnesses using a decision tree. Utilize mobile applications to gather farming data from sensors and human input. Upload the gathered data to the ThingSpeak server. Use a decision tree classifier to analyse and forecast the data that was obtained [8].

- Ch. Usha Kumari et.al. in 2019 a neural network classifier is used to detect leaf illness. K-means clustering is used for the segmentation. For cotton and tomato illnesses, different characteristics are retrieved, including contrast, correlation, energy, homogeneity, mean, standard deviation, and variance. Bacterial leaf spot, target spot septoria leaf spot, and leaf mould disease are the diseases of diseased leaves taken into account in the simulation. Diseases accuracy rates are 80%, for bacterial leaf spot, target spot, septoria leaf spot, and leaf mould [9].

- Shima Ramesh Maniyath et al. in 2018 proposed in order to distinguish between healthy and diseased leaf from the generated data sets, this work uses Random Forest. The phases of implementation in our proposed study are dataset construction, feature extraction, classifier training, and classification [10].

- Konstantinos P et al. in 2017 proposed unique deep learning models were created, based on particular convolutional neural network architectures, to recognise plant diseases from simple images of healthy or diseased leaves. An open collection of 87,848 pictures that were taken in both lab settings and actual field settings was used for model training. 25 plant species from 58 different classes of [plant, illness] combinations are included in the data, along with some healthy plants [11].

- Adhao Asmita Sarangdhar et al.in 2017 proposed in a system for detection and control of cotton foliar diseases along with soil quality monitoring. The thesis proposes a support vector machine-based regression system for the identification and classification of five cotton leaf diseases, i.e. bacterial blight, alternaria, gray mold, cereospra and Fusarium wilt. Once the disease is detected, the name of the disease with its medicine will be provided to the farmers using the Android app. The Android app is also used to display values of soil parameters such as moisture, humidity and temperature along with the water level in the tank. Using the Android app, farmers can turn on/off relays to control the motor and sprayer as needed. This entire foliar disease detection system and soil quality monitoring sensors are connected using a Raspberry Pi, making it an independent and cost-effective system. The overall classification accuracy of this proposed system is 83.26 [12].

- Vani Rajasekar et al. in 2021 presented a work where they used a 152-layer ResNet, which is 8 times deeper than VGG-19 but still has less foundation, on the ImageNet database. This study describes a productive process that makes use of deep learning. The suggested model performed admirably on both the enormous repository and the cotton disease picture dataset, with an incredible accuracy of 82%, according to the findings analysis. Find the variation across the edge based on Canny and Sobel along with color splitting methods (CMYK). Variations in the gray level in a region in the neighborhood of a pixel are ameasure of the texture. Then calculate the feature level fusion to combine the Edge, color and texture feature sets, after normalization in order to yield a joint feature vector (JFV). Instead of using GA for selection of the best features in the feature vector, we used proposed EPSO to select the best parameters from both global and local features results[1].

Table 2.2 Comparison of Different Techniques of Cotton Plant Disease Detection

Sl. No	Method	Accuracy	Description
[1]	152-layer ResNet, which is 8 times deeper than VGG-19 (2021)	82%	• Focal Loss function, rather than the original Cross Entropy Loss function, is used in the system to improve the learning capacity of the smaller features • Extensive use of Layers in the range of 18 to 34 Layers, hence increasing the Computation Requirement
[2]	Mask RCNN (2022)	94%	• This paper deals with images with different backgrounds (no use of controlled environment images), having multiple cotton leaves so that it is practically feasible • In general, two-stage methods are slower, The architecture was pre-trained on the COCO dataset • Only if three requirements are met, then a detection called true positive (TP): confidence score > threshold

[3]	Transfer Learning and KNN (2022)	95% and 86%	• Used metrics collaboration is training loss + validation loss, validation loss is very low, approximately 2% which is very good. • Low accuracy and can Identify only 2 Diseases
[4]	Faster R-CNN (2021)	87.1%,	• Real-time detection of cotton leaf diseases • Faster R-CNN model for detecting and classifying diseases on leaves • Even experienced pathologists often fail to diagnose diseases which leads to wrong conclusions and treatments
[5]	WEKA (2021)	92.69%	• WEKA (Waikato Environment for Knowledge Analysis) • SVM classifier accuracy performance is more when compared to other classifiers • images of larger size are difficult to be processed healthy and diseased classification accuracy using different classifiers is compared

[6]	Pattern Recognition Techniques (2015)	85%	• Automatic detection of plant diseases is useful in monitoring large crop fields • The snake segmentation algorithm provides efficient technique to isolate the diseased spot but is a very slow process.
[7]	IOT based CNN (2021)	-	• Automated monitoring for cotton plant for diseases, If the model is trained for 10k iterations, we might see high accuracy but takes lot of time. • Through this methodology, the prediction of diseases on the cotton plant is time-consuming to make it as effective and cost of setting up IoT sensors.
[8]	Decision Tree Classifier (2018)	-	• The system can predict the cotton crop diseases using decision tree with the help of the parameters like temperature, soil moisture, etc. based on the previous year data and through sensors. • Getting faster results based on real time data of plant and surrounding. • An IoT system may be complex and require technical expertise

			to set up and use, which may not be accessible to all users.
[9]	K-means clustering and Classification with ANN (2019)	80%	• Image segmentation which is done by using k-means clustering • algorithm and classification is done by using Artificial Neural Network. • This works well for tomato plant than cotton plant and tested only for diseases and not for the healthy plants.
[10]	Random Forest (2018)	-	• In order to distinguish between healthy and diseased leaf from the generated data sets. • This work uses Random Forest. • The phases of implementation in our proposed study are dataset construction, feature extraction, classifier training, and classification.
[11]	VGG- Convolutional Neural Network (2018)	-	• In this study, unique deep learning models were created, based on particular convolutional neural network architectures, • An open collection of 87,848 pictures that were taken in both lab settings and actual field

			settings was used for model training. • 25 plant species from 58 different classes of [plant, illness] combinations are included in the data, along with some healthy plants.
[12]	Support Vector Machine-Based Regression System(2017)	83.26%	• Detection and control of cotton foliar diseases along with soil quality monitoring. • The thesis proposes a support vector machine-based regression system for the identification and classification of five cotton leaf diseases, i.e., Bacterial blight, alternaria, gray mold, cereospra and fusarium wilt. • Using a raspberry pi, making it an independent and cost-effective system. • The overall classification accuracy of this proposed system is 83.26%.

From the above table we can infer the following-

- **Accurate Object Detection**: Mask R-CNN is capable of detecting objects with high accuracy, even in cluttered and complex scenes. This is due to the use of a Region Proposal Network (RPN) that generates candidate object regions, and the use of a RoI pooling layer that allows the network to handle objects of different scales.

- **Instance Segmentation**: In addition to object detection, Mask R-CNN also performs instance segmentation, which means that it can identify and segment individual instances of each object class. This is useful for applications such as medical imaging or autonomous driving, where the ability to distinguish between individual instances of the same object class is important.

- **Multi-Class Object Detection**: Mask R-CNN is able to recognise and separate different object classes in a same image, making it ideal for realworld applications where different object classes are present.

- **High Computational Efficiency**: Mask R-CNN is designed to be computationally efficient, and is able to run in real-time on modern GPUs. This makes it appropriate for usage in a variety of applications, including those with stringent computational limitations.

- **Transfer Learning**: Pre-trained Mask R-CNN models can be fine-tuned on new data to improve their performance on specific tasks, making it easier and faster to develop new object detection and segmentation applications.

Overall, Mask R-CNN's combination of accuracy, versatility, and computational efficiency make it a highly regarded methodology for object detection and segmentation.

CHAPTER 3

SYSTEM REQUIREMENTS

System requirements are the configuration that a system must have in order fora hardware or software application to run smoothly and efficiently. Failure to satisfy these requirements may result in installation problems or performance problems. The former may prevent a tool or application from getting installed, whereas the latter may cause a product to malfunction or perform below expectation or may be to hang or crash. System requirements are also referred to as minimum system requirements. System requirements can be broadly classified as hardware requirements, software requirements, functional requirements and non-functional requirements. The hardware system requirements often specify the OS version, processor type, memory size, available disk space and extra peripherals, if any, needed. The software system requirement consistent of all necessary requirements required for project development. A functional requirement defines a function of a system or its component, where a function is described as a specification of behavior between outputs and inputs. Non-functional requirements impose constraints on the planning or implementation like performance engineering requirements, quality standards, or design constraints.

3.1 Hardware Requirements

- Processor: i5
- RAM: 8GB
- Hard Disk:1TB
- Speed:1.2GHZ+
- GPU: NVIDIA GEFORCE / AMD Radeon Graphics

3.2 Software Requirements

- Operating System: Windows
- Language: Python, Kotlin
- Framework: Jupyter Notebook, VSCode, Android Studio
- Dataset: Kaggle
- Libraries: TensorFlow, Keras, NumPy, Flask

3.3 Functional Requirements

Functional requirements are the desired operations of a program, or system as defined in software development and systems engineering. The systems in systems engineering can be either software electronic hardware or combination software-driven electronics.

- **Jupyter Notebook**: The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text. Its uses include data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

- **VS Code**: Microsoft Visual Studio Code is an integrated development environment from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps.

- **Android Studio**: Android Studio is the official integrated development environment (IDE) forAndroid application development. It is based on the IntelliJ IDEA, a Java integrated development environment for software, and incorporates its code editing and developer tools.

3.4 Non-functional Requirements

A non-functional requirement is a requirement that specifies criteria which will be used to judge the operation of a system, instead of specific behaviors.

- **Usability:** The system must be simple that people like to use it, but not so complex that people avoid using it. The user must be familiar with the user interfaces and should not have problems in migrating to a new system with a new environment. The menus, buttons and dialog boxes should be named in a manner that they provide clear understanding of the functionality. Several users are going to use the system simultaneously, so the usability of the system should not get affected with respect to individual users.

- **Reliability**: The system should be trustworthy and reliable in providing the functionalities.Once a user has made some changes, the changes must be made visible by the system. Thechanges made by the Programmer should be visible both to the Project leader and to the Testengineer as well.

- **Security:** Apart from bug tracking the system must provide necessary security and must secure the whole process from crashing. As technology began to grow in fast rate the safety became the main concern of an organization. Millions of dollars are invested in providing security. Bug tracking delivers the utmost security available at the very best performance rate possible, ensuring that unauthorized users cannot access vital issueinformation without permission. Bug tracking system issues different authenticated users their secret passwords so that there are restricted functionalities forall the users.

- **Scalability:** The system should be scalable enough to add new functionalities at a later stage. There should be a standard channel, which may accommodate the new functionalities.

- **Performance:** Ensure software applications will perform well under their expected workload. The response time of the system is good.

- **Maintainability**: The system monitoring and maintenance should be simple and objective in its approach. There should not be too many jobs and roles running on different machinessuch that it gets difficult to watch whether the jobs and roles are running without errors.

- **Portability:** The system should be easily portable to another system. This is required when the web server, which is hosting the system gets stuck due to some problems, which requires the system to be taken to another system.

- **Flexibility:** The system should be flexible enough to allow modifications at any point of time.

- **Reusability**: The system should be divided into such modules that it could be used as a part of another system without requiring much of work.

CHAPTER 4

SYSTEM ANALYSIS

System development can generally be thought of having two major components: System analysis and System design. System design is that the process of planning a replacement system or one to exchange or complement of an existing system. But before this planning can often done, we must thoroughly understand the existing systems and determine how computers can be best used to make its operation simpler and more effective. System analysis is the process of gathering and interpreting facts, diagnosing problems, and using the knowledge to recommend improvements to the system. This is the job of the system analyst.

The principal objective of the system-analysis phase is to specify what the system needs to do to meet the wants of end users. System analysis is conducted for the aim of studying a system or its parts so as to spot its objectives. It is a drag solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish their purpose.

The process of system analysis consists of various steps:

- **Collecting Information About Existing Systems:** This step involves collection of all information about the existing systems by research, observations about working operations of the systems in various scenarios and reviews about the systems by the users.
- **Identifying The Inputs, Outputs and Processes of The Existing Systems:** Every system has inputs and outputs, the system analyst needs to identify the data input to the existing system, and the data output given by the existing system. This is because any new system that is designed will need to affect similar inputs and outputs as the existing system. Any new system that is created will need to take in the same input data and will have to produce similar kind of outputs. The analyst also has got to understand how the existing system works.
- **Identifying Issues in The Existing Systems:** No system is perfect and it is job of the system analyst to try and identify where and what are the problems in the existing system. If these problems are often fixed, the system will work more efficiently and, Within the case of a business, be more profitable.

- **New System Requirements Specifications:** After the problems with existing systemare understood, the system analyst can begin to plan how the new system will fix those problems. The analyst specifies an inventory of requirements for the new system. This list is usually called the Requirements Specification.

- **Type Of Hardware and Software Required:** The system analyst will now need to decide hardware and software which will be required for the new system. The Hardware requirements would include how many computers to use, what type of network to use, how many servers required, etc. The Software requirements would include decisions such as whether to use ready- made off-the-shelf software or use a custom-written software.

4.1 Existing System

The existing system of our project makes use of various other machine learning models like K-Means Clustering, Support Vector Machine and Decision Tree Classifiers all of these models give a very less accuracy even after providing a large dataset. There are also hardware-based systems that are very expensive and requires regular maintenance.

- The system is interfaced with a digital/web camera which can alter the farmers to require pictures of plant leaves. victimization image process algorithms, the captured pictures square measure analyzed to discover the symptoms of illness. Disadvantage: Implementation was done by using Arduino and MATLAB.

- Color transformation structure RGB is converted into HSV space, Masking and removing of green pixels with pre-computed threshold level, segmentation then texture parameters are compared to texture parameters of normal leaf. Disadvantage: Process is time consuming and lengthy.

- Using ‘Decision Tree Classifier’ by taking parameters as temperature, soil moisture, etc. This would help farmers by providing better quality productions. Disadvantages: Requires real time temperature data.

4.2 Proposed System

The process of diagnosis of leaf diseases involves many tasks, such as image acquisition, preprocessing of image, extraction of features from image, classifying leaf diseases is depends on image feature that is color features, shape features and texture features and then providing the cure. The first stage is the image acquisition. In this phase, image is uploaded from the images of the leaf dataset. Then the preprocessing on image is performed using different techniques. In the third phase, extraction of features is done from the picturefor the part of the leaf which is infected. This is done on the basis of particular properties between pixels in the image or their texture Then to classify the features which are represent the given image statistical analysis tasks are performed then use deep learning-based CNN model to compare image features. Finally, classification result, once the result is predictedcure is provided within the app.

CHAPTER 5

SYSTEM ARCHITECTURE

5.1 System Architecture

The architecture of CNN and flow diagram of the proposed system flow diagram is shown in Fig. 5.1 and Fig.5.2

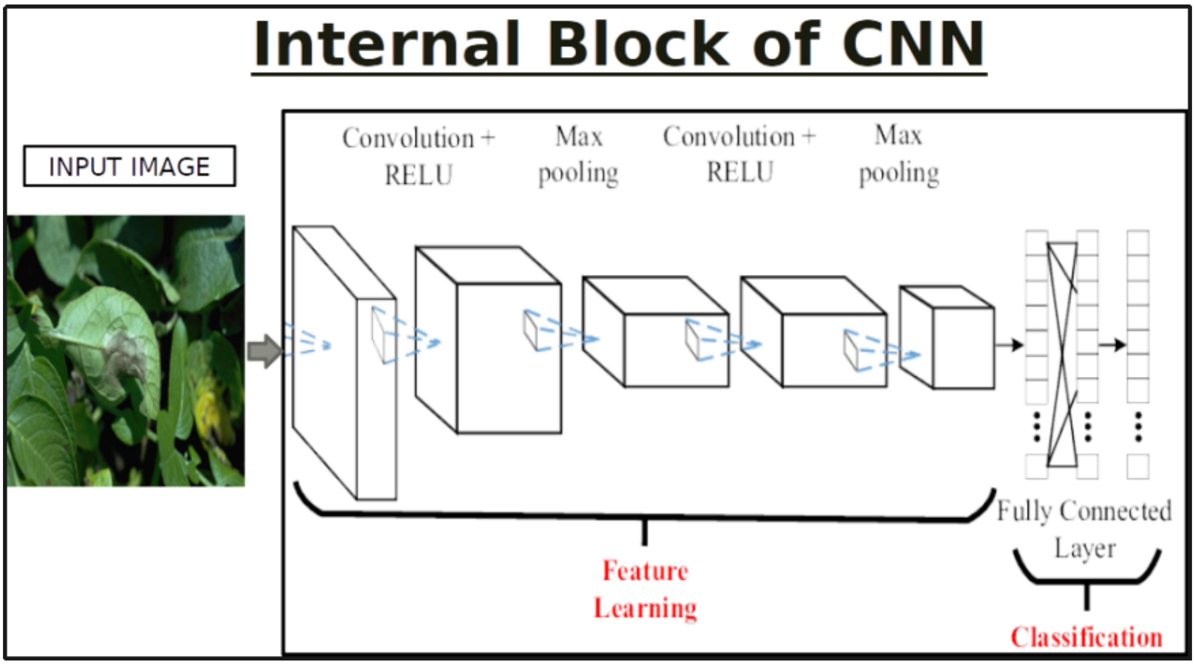

Fig 5.1 System Architecture of CNN

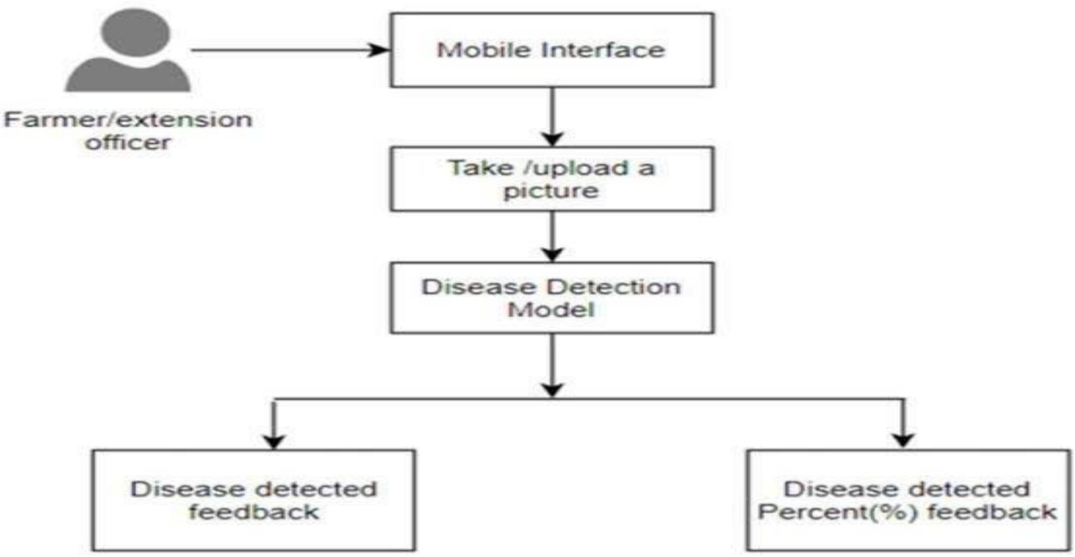

Fig 5.2 Flow diagram of proposed system

5.2 Proposed Methodology

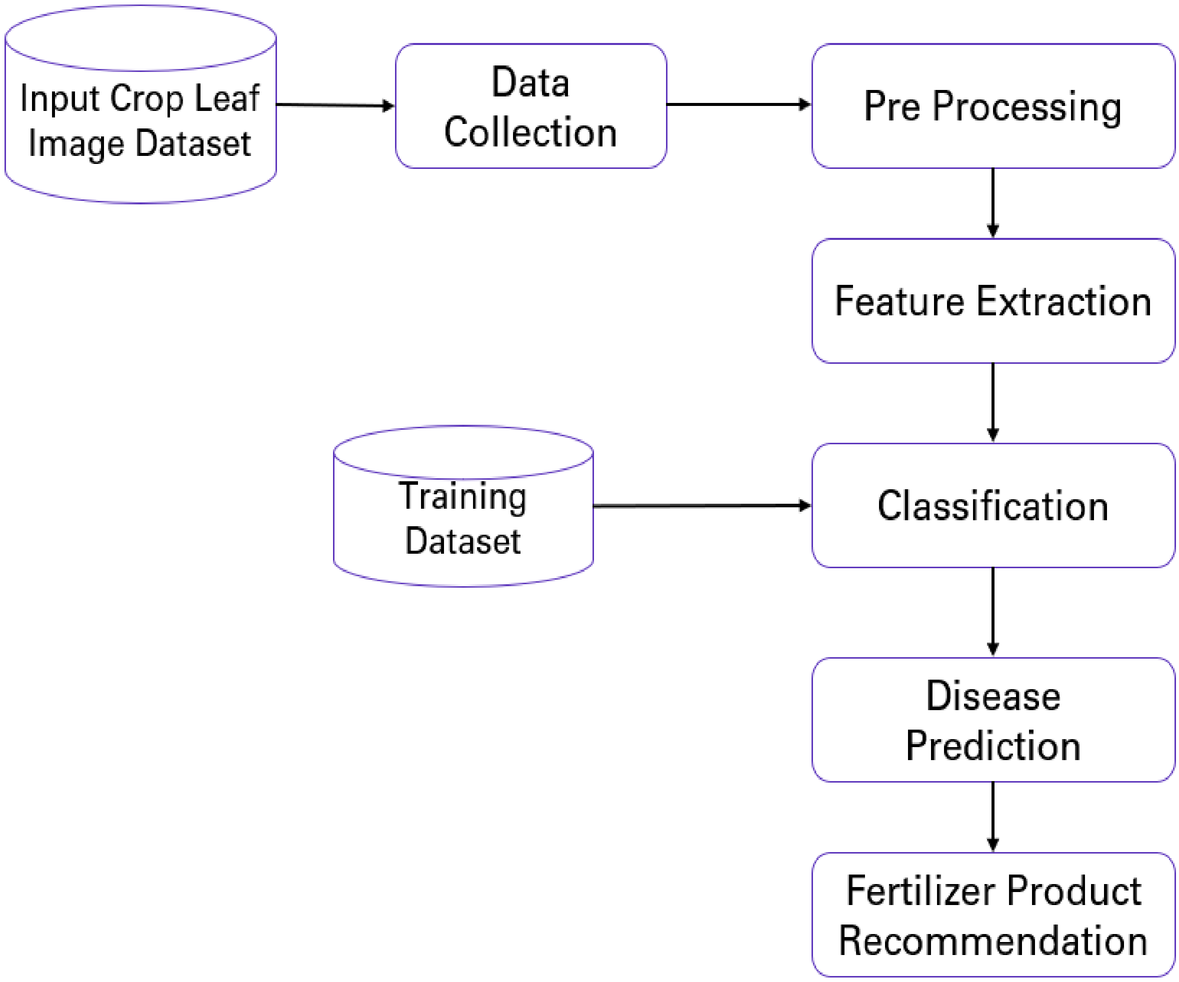

Fig 5.3 Proposed Methodology

The steps involved in proposed methodology for cotton plant disease detection according to fig. 5.3 are discussed below.

- **Dataset Collection**: The datasets of cotton plant leaf images are collected both from the Kaggle website and also live images from the fields. We have managed to collect approximately 2100+ images.

- **Pre-processing** of the sample data is done to remove noise in an image, different pre-processing techniques are used. Clipping of leaf image is applied to extract the region of the image in which we are interested. Rescaling of the sample data is done to obtain multiple images from single sample data.

- **Feature Extraction** is a type of dimension reduction technique that effectively represent the useful part of the image, here the pixels from the image are collected and learnable weights and biases are assigned.

- **Classification and disease prediction** is achieved by applying CNN is to reduce images into an easier-to-process form, without compromising the features that are essential for getting a good prediction. To classify the disease in plants in a precise manner the images are provided as input. The convolution layer is used for extracting the features from the images. The pooling layer computes the feature values from the extracted features. Depending on complexity of images, the convolution and pooling layer can be further increased to obtain more details. Fully connected layer uses the output of previous layers and transforms them into a single vector that can be used as an input for next layer. The output layer finally classifies the plant disease.

- **Displaying Cure** finally, based on the output cure to plant is provided by recommending the suitable fertilizers to be used.

CHAPTER 6

IMPLEMENTATION

6.1 Data Collection

Dataset are collected from cotton fields manually and from Kaggle, a subsidiary of Google LLC, is an online community of data scientists and machine learning practitioners. Kaggle allows users to find and publish data sets, explore and build models in a web-based data-science environment, work with other data scientists and machine learning engineers, and enter competitions to solve data science challenges. Here our dataset mainly consists of cotton plant leaf images both of infected plants and healthy plants. Our dataset consists of 2500 images which are mainly categorized into five categories.

6.2 Approaches of Models

6.2.1 Convolutional Neural Network:

A Convolutional Neural Network, also known as CNN or ConvNet, is a class of Neural Network that specializes in processing data that has a grid-like topology, such as animage. A digital image is a binary representation of visual data. Just as each neuron responds to stimuli only in the restricted region of the visual field called the receptive field in the biological vision system, each neuron in a CNN processes data only in its receptive field as well. The layers are arranged in such a way so that they detect simpler patterns first (lines, curves, etc.) and more complex patterns (faces, objects, etc.) further along. By using a CNN, one can enable sight to computers.

A CNN typically has three layers: a convolutional layer, a pooling layer, and a fully connected layer.

- **Convolution Layer**: The convolution layer is the core building block of the CNN. It carries the main portion of the network's computational load. This layer performs a dot product between two matrices, where one matrix is the set of learnable parameters otherwise known as a kernel, and the other matrix is the restricted portion of the receptive field. The kernel is spatially smaller than an image but is more in- depth. This means that, if the image is composed of three (RGB) channels, the kernel height and width will be spatially small, but the depth extends up to all three channels. During the forward pass, the kernel slides across the height and width of the image-producing the image representation of that receptive region. This produces a two-dimensional representation of theimage known as an activation map that gives the response of the kernel at each spatial positionof the image.

- **Pooling Layer:** The pooling layer replaces the output of the network at certain locations by deriving a summary statistic of the nearby outputs. This helps in reducing the spatial size of the representation, which decreases the required amount of computation and weights. The pooling operation is processed on every slice of the representation individually. There are several pooling functions such as the average of the rectangular neighborhood, L2 norm of the rectangular neighborhood, and a weighted average based on the distance from the central pixel. However, the most popular process is max pooling, which reports the maximum output from the neighborhood.

- **Fully Connected Layer:** Neurons in this layer have full connectivity with all neurons in the preceding and succeeding layer as seen in regular FCNN. This is why it can be computed as usual by a matrix multiplication followed by a bias effect. The FC layer helps to map the representation between the input and the output. Since convolution is a linear operation and images are far from linear, non-linearity layers are often placed directly after the convolutional layer to introduce non-linearity to the activation map.

- **The Rectified Linear Unit (ReLU)** has become very popular in the last few years. It computes the function $f(\kappa)=\max(0,\kappa)$. In other words, the activation is simply threshold at zero. In comparison to sigmoid and tanh, ReLU is more reliable and accelerates the convergence by six times. Unfortunately, a con is that ReLU can be fragile during training. Alarge gradient flowing through it can update it in such a way that the neuron will never get further updated. However, we can work with this by setting a proper learning rate.

6.3 Algorithm Selection

Algorithm for Disease Prediction

Convolution Neural Network (CNN) Algorithm: The structure of CNN algorithm includes two layers. First is the extraction layer of features in which each neuron's input is directly connected to its previous layer's local ready fields and local features are extracted. The spatial relationship between it and other features will be shown once those local features are extracted. The other layer is feature map layer; Every feature map in this layer is a plane, the weight of the neurons in one plane are same. The feature plan’s structure makes use of the function called sigmoid. This function known as activation function of the CNN, which makesthe feature map have shift in difference. In the CNN each convolution layer is come after a computing layer and its usage is to find the local average as well as the second extract; this extraction of two feature is unique structure which decreases the resolution.

Step 1: Dataset collection.

Step 2: Perform feature selection using information gain and ranking

Step 3: Apply Classification algorithm CNN

Step 4: Calculate each Feature fix value of input layer

Step 5: Calculate bias class of each feature

Step 6: The feature map is produced and it goes to forward pass input layer

Step 7: Calculate the convolution cores in a feature pattern

Step 8: Produce sub sample layer and feature value.

Step 9: Input deviation of the kth neuron in output layer is Back propagated.

Step 10: Finally give the selected feature and classification results along with cure if the plant is diseased.

```

#Building cnn model
cnn_model = keras.models.Sequential([
                                keras.layers.Conv2D(filters=32, kernel_size=3, input_shape=[224, 224, 3]),
                                keras.layers.MaxPooling2D(pool_size=(2,2)),
                                keras.layers.Conv2D(filters=64, kernel_size=3),
                                keras.layers.MaxPooling2D(pool_size=(2,2)),
                                keras.layers.Conv2D(filters=128, kernel_size=3),
                                keras.layers.MaxPooling2D(pool_size=(2,2)),
                                keras.layers.Conv2D(filters=256, kernel_size=3),
                                keras.layers.MaxPooling2D(pool_size=(2,2)),

                                keras.layers.Dropout(0.5),
                                keras.layers.Flatten(), # neural network beulding
                                keras.layers.Dense(units=128, activation='relu'), # input layers
                                keras.layers.Dropout(0.1),
                                keras.layers.Dense(units=256, activation='relu'),
                                keras.layers.Dropout(0.25),
                                keras.layers.Dense(units=5, activation='softmax') # output layer
])


# compile cnn model
cnn_model.compile(optimizer = Adam(learning_rate=0.0001), loss='sparse_categorical_crossentropy', metrics=['accuracy'])

```

Fig 6.3.1 CNN Model

An autoregressive integrated moving average is a statistical analysis model that uses time series data to either better understand the data set or to predict future trends.

An autoregressive integrated moving average model is a form of regression analysis that gauges the strength of one dependent variable relative to other changing variables. The model's goal is to predict future securities or financial market moves by examining the differences between values in the series instead of through actual values.

A model can be understood by outlining each of its components as follows:

- Autoregression (AR): refers to a model that shows a changing variable that regresses on its own lagged, or prior, values.
- Integrated (I): represents the differencing of raw observations to allow for the time series to become stationary (i.e., data values are replaced by the difference between the data values and the previous values).
- Moving average (MA): incorporates the dependency between an observation and a residual error from a moving average model applied to lagged observations.

CHAPTER 7

SYSTEM TESTING

System testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results. Although crucial to software quality and widely deployed by programmers and testers, software testing still remains an art, due to limited understanding of the principles of software. The difficulty in software testing stems from the complexity of software: we cannot completely test a program with moderate complexity.

Testing is more than just debugging. Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Test techniques include, but are not limited to the process of executing a program or application with the intent of finding software bugs (errors or other defects). The purpose of testing can be quality assurance, verification and validation, or reliability estimation. Softwaretesting can be stated as the process of validating and verifying that a computer program/application/product:

- Works as expected
- Can be implemented with the same characteristics
- Satisfies the needs of stakeholders.
- Meets the requirements that guided its design and development

Testing can be used as a generic metric as well. Software testing is usually a trade- off between budget, time and quality. Software testing, depending on the testing method employed, can be implemented at any time in the software development process. Traditionally most of the test effort occurs after the requirements have been defined and the coding process has been completed, but in the agile approaches most of the test effort is on- going. As such, the methodology of the test is governed by the chosen software development methodology.

7.1 Testing Methodologies

System Testing: The entire system is tested as per the requirements. It is a type of functional testing that is based on overall requirements specifications and covers all combined parts of a system.

Table 7.1: Testing Methodologies Used

Title	Authors	Algorithm Used	Accuracy
"Cotton leaf disease identification using pattern recognition techniques"	P.R. Ruth and R.V. Kshirsagar	CNN which uses snake segmentation	85.52%.
"Super pixel-based roughness measure for cotton leaf diseases detection and classification"	Yogita K Dubey, Milind M. Mushrif and Sonam Tiple	SLIC algorithmand SVM	91%
"Cotton leaf disease detection and recovery using genetic algorithm"	Sivasangari, K. Priya and K. Indra	GA-SVM classifier	90%
"Leaf disease detection using Image Processing"	Sujatha R, Y Sarvan Kumar and Garine Uma Akhil	K-means algorithm	86%

Table 7.2: Comparison of various models

Sl. No	Algorithms Used	Results
1.	Neural Network	83.33%
2.	DWM	84.55%
3.	Image Correlation	86%
4.	K Means clustering	87.21%

CHAPTER 8

SNAPSHOTS

The snapshots of cotton plant disease detection app are shown in fig 8.1, fig 8.2, fig 8.3, fig 8.4.

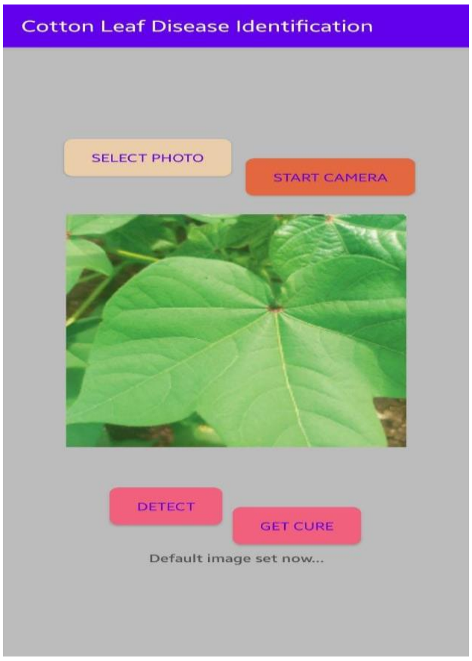

Fig 8.1: Home Page

This screen in fig 8.1 links to various options in the Apps such as select photo, start camara, detect and get cure.

Fig 8.2: Cotton Disease Prediction Page

This screen in fig 8.2 takes Image as Input and predicts the category of Disease.

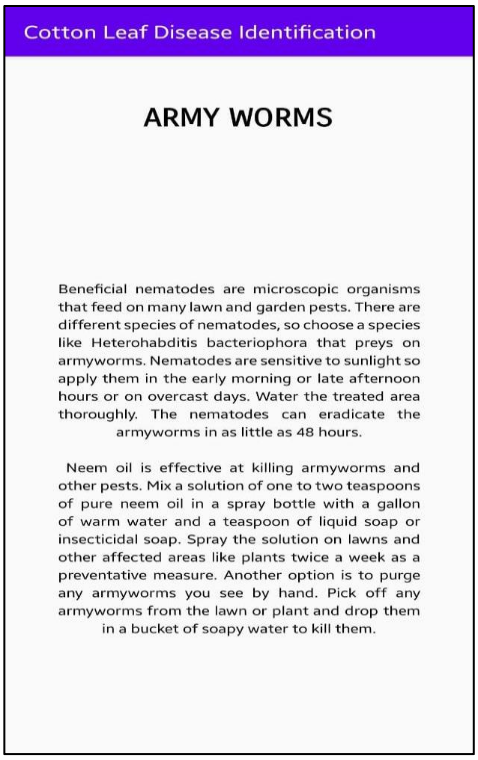

Fig 8.3: Cure Page – Army Worm

This screen in fig 8.3 suggests pesticides to particular disease

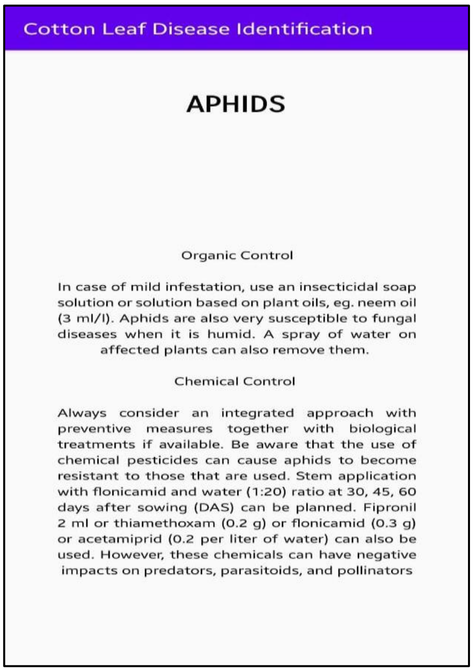

Fig 8.4 : Organic Cure page - Aphids

This screens fig 8.4 predict Organic Cure to diseases like Aphids along with chemical cure.

CONCLUSION

The disease analysis is done for the cotton leaf diseases detection is addressed, the analysis of the diseases that are present on the leaves can be effectively detected in the early stage before it will damage the whole plant. Here the technique presented can able to detect the disease more accurately, we can say that, we can archive good productivity by preventing the different diseases which are present on the leaves of cotton plant using weather dataset and image processing. The usage of classification and feature extraction processes has enhanced the performance of the system which provides better results with an accuracy of above 95% along with cure within the App. The diseases detected by this app are Bacterial Blight, Army Worm, Target Spot, Powdery Mildew, Aphids, and also can detect healthy leaf.

FUTURE ENHANCEMENT

In the future, there are several potential enhancements that could be implemented in cotton plant disease detection using deep learning techniques. These enhancements aim to improve the accuracy, efficiency, and overall performance of the disease detection system. Here are some possibilities, Larger and more diverse datasets: Deep learning models benefit from large and diverse datasets for training. In the future, efforts can be made to collect and annotate more cotton plant images depicting various disease symptoms, growth stages, and environmental conditions. This would help in training more robust models capable of accurately detecting a wide range of diseases. Pre-training models on general image recognition tasks, such as ImageNet, can provide a starting point for cotton plant disease detection. Fine-tuning these pre-trained models on cotton plant disease datasets can significantly improve their performance. Apart from visual information, other data modalities can be integrated into the disease detection system. For example, sensor data capturing environmental factors like temperature, humidity, and soil moisture could be combined with visual data to enhance disease detection accuracy. Multi-modal deep learning approaches can be developed to effectively fuse and process these different data sources. Future systems can be designed for real-time monitoring of cotton plants using deep learning models. This would involve deploying cameras and sensors in the field to continuously capture and analyze plant images and environmental data. Automated disease detection algorithms can provide immediate alerts to farmers, enabling timely interventions and reducing crop losses. To make disease detection more accessible to farmers in remote areas, efforts can be made to develop lightweight deep learning models that can run efficiently on mobile devices or edge computing devices. This would eliminate the need for a constant internet connection and allow farmers to perform on-device disease detection using their smartphones or dedicated hardware.

It's important to note that while deep learning shows promise in cotton plant disease detection, the success of these enhancements will depend on factors such as data quality, model architecture, training techniques, and the integration of domain knowledge.

REFERENCES

[1] Vani Rajasekar, Kola Venu, Soumya Ranjan Jena, R. Janani Varthini, “Detection of Cotton Plant Diseases Using Deep Transfer Learning”, Journal of Mobile Multimedia, DOI:10.13052/jmm1550-4646.1828, November 2021.

[2] Prashant Udawant, Pravin Srinath, “Cotton Leaf Disease Detection Using Instance Segmentation”, Journal of Cases on Information Technology, April 2022.

[3] Smruti Kotian, Pravalika Ettam, Shubhangi Kharche, Karuna Saravanan, Kavitha Ashokkumar, “Cotton Leaf Disease Detection Using Machine Learning”, 2nd International Conference on Advancement in Electronics & Communication Engineering (AECE 2022), July 2022.

[4] Rehan Sarwar, Muhammad Aslam, Khaldoon S Khurshid, Tauqir Ahmed, Ana Maria Martinez-Enriquez and Talha Waheed, “Detection and Classification of Cotton Leaf Diseases Using Faster R-CNN on Field Condition Images”, Acta Scientific AGRICULTURE (ISSN: 2581-365X) , September 2021.

[5] Patil B and Barkpalli V. “A Perspective View of Cotton Leaf Image Classification Using Machine Learning Algorithms Using WEKA”, DOI:10.21203/rs.3.rs-502091, May 2021.

[6] Rothe PR and Kshirsagar RV. “Cotton leaf disease identification using pattern recognition techniques”, International conference on pervasive computing (ICPC), DOI: 10.1109/PERVASIVE.2015.7086983, April 2015.

[7] Bhushan V. Patil and Pravin S. Patil, “Computational Method for Cotton Plant Disease Detection of Crop Management Using Deep Learning and Internet of Things Platforms”, DOI: 10.1007/978-981-15-5258-881, January 2021.

[8] Jayraj Chopda, Hiral Raveshiya, Sagar Nakum, Vivek Nakrani, International Journal of Research Publication and Reviews, DOI: 10.1109/ICSCET.2018.8537336, November 2018.

[9] Ch. Usha Kumari, S. Jeevan Prasad, G. Mounika, “Leaf Disease Detection: Feature

Extraction with K-means clustering and Classification with ANN”, 3rd International Conference on Computing Methodologies and Communication (ICCMC), DOI: 10.1109/ICCMC.2019.8819750, August 2019.

[10] Shima Ramesh Maniyath, Vinod P V, Niveditha M, Pooja R, Prasad Bhat N, Shashank N, Hebbar Ram, “Plant Disease Detection Using Machine Learning”, International Conference on Design Innovations for 3Cs Compute Communicate Control, DOI: 10.1109/ICDI3C.2018.00017, April 2018.

[11] Konstantinos P. Ferentinos, “Deep learning models for plant disease detection and diagnosis”, DOI: 10.1016/j.compag.2018.01.009, January 2018.

[12] Adhao Asmita Sarangdhar, Prof. Dr. V. R. Pawar, “Machine Learning Regression Technique for Cotton Leaf Disease Detection and Controlling using IoT”, International Conference on Electronics Communication, DOI: 10.1109/ICECA.2017.8212855, December 2017,

[13] Preetha S, Musqan Arshad, “Survey On Plants Disease Detection Using Machine Learning”, International Journal of Recent Engineering Science (IJRES) Volume 7 Issue 2, 27-29, Mar-Apr 2020.

[14] Nikita Yadav, Shivani Kasar, Dnyaneshwari Abuj, Ankita Vadvale, Dr. S.C. Dharmadhikar, “An Improved Cotton Leaf Spot Disease Detection using Proposed Classifiers”, International Research Journal of Engineering and Technology (IRJET), Feb 2021.

[15] Wikipedia - https://www.wikipedia.org/.

[16] Google Scholar - https://scholar.google.com.

[17] J. Karthika, K. Mathan Kumar, M. Santhose, T. Sharan and S Sri Hariharan, “Disease Detection in Cotton Leaf Spot Using Image Processing”, ICCCEBS 2021 IOP Publishing, 2021.

[18] Azath M, Melese Zekiwos, and Abey Bruck, “Deep Learning-Based Image Processing for Cotton Leaf Disease and Pest Diagnosis”, Hindawi, DOI: https://doi.org/10.1155/2021/9981437 , 2021.