Cotton Plant Disease Detection Using Deep Learning

Chethana H T[1], Kunal S[2], Sahil Jain[3], Skandhan M[4], Suloni Praveen[5], Swathi Meghana K R[6]

[1,2,3,4,5,6] Department of Computer Science and Engineering, Vidyavardhaka College of Engineering, Mysuru - 570002, Karnataka, India

chethanaht@vvce.ac.in
kunalsrinidhi25@gmail.com
jainsahil1116@gmail.com
skandhanmnss208@gmail.com
sulonipraveen@gmail.com
swathi.meghana9@gmail.com

Abstract. Most of the farmers cultivate cotton in large numbers but one of the biggest issues in recent decades has been the cotton leaf disease which affects crop productivity and money. The diseases named “Leaf Lesions”, “Bacterial Blight”, “Curl virus”, and “Fusarium wilt” have a significant impact on cotton leaves. A front-end application is developed that takes both the uploaded image and the live image from the camera app. ThenConvolutional Neural Network (CNN) a Deep Learning algorithm is used in which learnable weights are assigned toan input image and biases are assigned to various objects in the image which differentiates one from the other. The developed application classifies the class of leaf diseases according to the crop features and then provides a cure for the plant illness containing the name, cost, and description of the pesticide. It also displays major driving factors for cotton production and varieties of cotton in India. In this research work, crops are categorized according to the foundation of color, and experimental results prove that it provides a recognition accuracy of 95% and performs better than the current algorithms.

Keywords: Convolutional Neural Network, Deep Learning, Support Vector Machine (SVM), Image Processing, Decision Tree Classifier, K-means Clustering

1. Introduction

One of the most important cash crops in India is cotton. The main cause of cotton's decreased productivity are cotton diseases. A subset of smart farming is the identification of crop diseases. The development of crops depends heavily on the early diagnosis of crop diseases. Most Indians work in the agriculture industry because the country is well-known for its agricultural practices. Cotton is a significant economic participant. Variations in climatic conditions and fungal and bacterial diseases have contaminated many of the plants. Thus, high-quality fertilizers must be accustomed for protecting the crops. The farmers' current methods for spotting cotton leaf disease include using their unaided eyesight. It costs money and necessitates constant inspection and monitoring, both of which cause time loss. [15]. The Different types of cotton plant diseases are shown in Fig. 1.

Fig. 1. Several Kinds of Cotton Plant Diseases

From this point forward, farmers will gain immediate advantages from the automatic forecast of many agricultural illnesses, saving them time, money, and crop life. A deep-learning algorithm is being utilized to address the growing discomfort caused to the farmers [19]. A system is proposed in this study that can forecast the occurrence of cotton crop disease according to the structure as well as the standard of the leaf. The proposed technique is primarily utilized to create a program that can identify illnesses in cotton leaves. Datasets are gathered from Kaggle and include pictures of healthy and damaged plant leaves. To solve agricultural problems, the proposed model gives an accuracy of 95% whichhelps the farmers to predict the disease of the cotton crop. Moreover, it informs the farmers regarding how to cure cotton plant disease [16].

The document is structured in the manner described below. A quick rundown of earlier developed cotton plant disease detection systems is presented in section 2. Sections 3 and 4 have been devoted to the suggested approach for creating the suggested framework as well as the experimental findings and analysis. The paper is concluded in Section 5.

2. Literature Survey

The literature survey provides a summary of the sources that have been investigated during the research. Numerous authors have proposed various methods to detect cotton leaf diseases.

J Karthika et al. [17] proposed a disease detection system utilizing the image in the cotton leaf spot processing**.** The technology is linked to a digital camera, which can be used to manipulate farmers for plant leaf exploitation using image processing algorithms. The images are evaluated to determine the symptoms of sickness. The farmer receives notification of the type of disease via the GSM interface. Multi-class SVM is used to classify diseases such as Bacteria blight, black arm spot, and leaf spot.

Preetha S et al. [13] discussed about the importance of agriculture which is an important source of revenue for Indians. The authors of this study proposed a method that uses a color transformation structure in which RGB is converted to HSV space, green pixels are masked and removed with a pre-computed threshold level, segmentation, and texture parameters are considered. Remote sensing methods are fast and superior. The characteristics of the agriculture field, different crop types and differences in their characteristics such as form, and texture are considered for working in agriculture areas.

Shima Ramesh Maniyath et. al. [10] proposed a system that makes use of Random Forests in distinguishing healthy and unhealthy leaves from the data sets produced. The writers of this research have suggested a system that uses a Random forest learning strategy for tasks like regression and classification. The histogram of oriented gradients (HOG) is a feature descriptor for object detection. The suggested method calls for additional computational power and it consumes resources as it constructs several trees to integrate their outputs.

Nikita Yadav et al. [14] suggested a mechanism to detect crop disease. It is among the issues that cause a reduction in the calibre and number of crops. The proposed research work is to analyze various recognition algorithms and picture manipulation techniques to detect crop disease. In this study, a review of numerous machine learning algorithms, such as decision trees, Random Forests, and multilayer regressors, is utilized to improve the system's precision. Crop leaf photos are used as input, and the image is then processed to determine whether there is an illness. If a disease is detected, then it will tell what type of disease it is and provide solutions such as pesticides or chemicals to cure that disease.

Paramjeet Singh et al. [16] proposed a prediction of cotton plant disease using a Support Vector Machine (SVM). It is one of the most widely used supervised learning algorithms for categorization and regression issues. The SVM approach attempts to draw the best boundary or line for dividing n-dimensional space into classes, ensuring that all subsequent data points are added to the appropriate category.

Azath M et al. [18] proposed a system in which the feature extraction process begins with splitting the input image into pixel values, followed by setting the color variance and quantization value of the edge detector. Simple edge detection is performed, with the edge pixels inside being classified as structural. Color variance is then computed in the remaining blocks.

Vani Rajasekar et al. [1] proposed a decision tree learning technique used in statistics, Machine learning and data mining. Decision trees are among the most widely used algorithms for machine learning in their intelligibility and simplicity.

Shwetha Kumari et al. [16] proposed an improved particle swarm optimization approach for extracting features from sick cotton leaf photos that use the skew divergence variance features strategy. The feature extraction process begins with splitting the input image into pixel values, followed by setting the color variance and quantization value of the edge detector. A simple edge detection is done, and blocks with edge pixels inside are classified as structural. The color variance is then computed in the remaining blocks. [16].

Pravin Srinath et al. [2] examines the algorithm's suitability for detecting cotton leaf diseases in actual environments by 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. [3] proposed a system to develop a transfer system of learning for creating a learning model. To enhance the effectiveness of the model, transfer learning and KNN is utilized to categorize the diseased leaves into various diseases and have achieved an accuracy of 95% and 86% respectively [3].

Rehan Sarwar et al. [4] suggested a detection method for leaf diseases in real-world scenarios. Experiments are carried out on the CCL dataset, comprising more than 700 cotton leaf field photos utilizing four distinct classes, three among which belong to illness and one of which is a part of robust leaves. InceptionV2 is chosen with Faster object detection R-CNN. Due to the limited number of images, transfer learning was essential in the CCL dataset training process. 0.01275 and 87.1% are the computed loss and mean, average accuracy respectively.

Bhagya Patil et al. [5] exhibited a classification of healthy and unhealthy cotton leaf images using several machine learning techniques using WEKA. Almost 3000 photos of two distinct classes of healthy and unhealthy leaves were included in this collection. For picture segmentation, an active contour based on modified factorization was employed. Two separate texture and color traits were extracted from segmented photos and used for teaching and assessment by various machine learning algorithms. Other classifiers were outperformed by the multilayer perceptron, which achieved a classification performance of 92.69%.

P.R. Rothe et al. [6] proposed a method for identifying patterns in three cotton leaf diseases and classification. This dataset on cotton was gathered using different cotton fields in their natural state. To make the pictures noise-free, pre-processing and image segmentation were done by applying a Gaussian filter and active contour model utilization. An accuracy of 85% was attained in the categorization of cotton leaf diseases using a dataset that was trained using adaptive neuro fuzzy.

Bhushan Patil et al. [7] proposed a camera and sensor system based on IoT deployed on crop fields. The contaminated survey regions were used to capture all the photographs during daily surveys. Using an intelligent agriculture IoT platform with sensor mixing regulates the irrigation system by monitoring field conditions [7].

Jayraj Chopda et al. [8] suggested a framework that captures the variables like temperature and soil moisture. The suggested method may predict cotton crop illnesses using a decision tree.

Usha Kumari et al. [9] devised a method to detect leaf sickness. For segmentation, K-means clustering is used. The disease accuracy rates for bacterial leaf spot, target spot, septoria leaf spot, and leaf mold are all 80%. [9].

Shima Ramesh Maniyath et al. [10] proposed a system to discriminate between healthy and diseased leaves using the generated data sets. The proposed study's execution steps are dataset building, feature extraction, classifier training, and classification. [10].

Konstantinos P et al. [11] proposed a distinct model for deep learning based on convolutional neural network architecture to identify plant diseases using basic photos of healthy or diseased leaves. For model training, an open collection of 87,848 photographs obtained in both lab and outdoor situations was employed. The data includes 25 plant species from 58 different classes of [plant, disease] combinations, as well as some healthy plants.

Adhao Asmita Sarangdhar et al. [12] proposed a system for detecting and controlling cotton foliar diseases, as well as monitoring soil quality. Once the disease has been identified, the farmers will be given the disease's name as well as its treatment via the Android app. The Android app also displays soil parameter measurements such as moisture, humidity, and temperature, as well as the tank's water level. Farmers can use the Android app to turn on and off relays to manage the motor and sprayer as needed. The Raspberry Pi is used to connect the full foliar disease detection system and soil quality monitoring sensors, making it a self-contained and cost-effective solution. This proposed system's overall classification accuracy is 83.26%. [12].

Vani Rajasekar et al. [1] presented a work where they utilized a 152-layer ResNet, which is 8 times deeper than VGG-19 but still has less foundation on the ImageNet database. The suggested model performed admirably on both the enormous repository and the cotton disease picture dataset with an incredible accuracy of 82% [1].

Most of the works [1][2][3][4] are carried out to detect cotton plant disease with machine learning. There haven't been many studies that use deep learning to detect cotton plant disease. As a result, there is room for novel ways to diagnose illnesses in cotton plants using deep learning.

3. Proposed Methodology

The existing system makes use of several different machine learning models, including K- Means clustering, Support

Vector Machine (SVM) and Decision Tree classifier. All these models provide a very low accuracy even after providing a large dataset. The existing system is a hardware-based system that requires regular maintenance and is very expensive.

The diagnosis of leaf diseases entails several phases, including picture collecting, preprocessing, feature extraction, and classification based on image properties such as color, shape, and texture aspects. Images from the leaf dataset's pictures are uploaded during the picture capture phase. Next preprocessing is carried out via several methods. In the next phase, characteristics are taken from the infected leaf which comes from properties between pixels in the picture. The system architecture and flow diagram of the suggested framework are shown in Fig. 2. and Fig. 3.

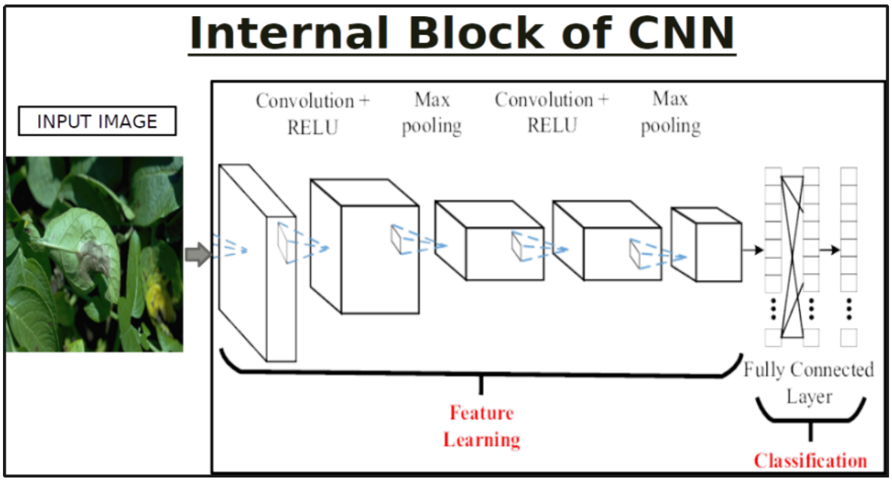

Fig. 2. System architecture of the system proposed.

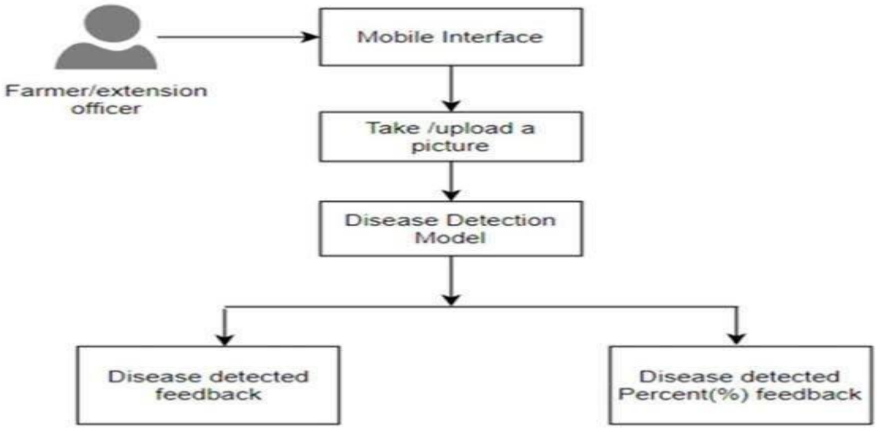

Fig. 3. Proposed System Flow Diagram

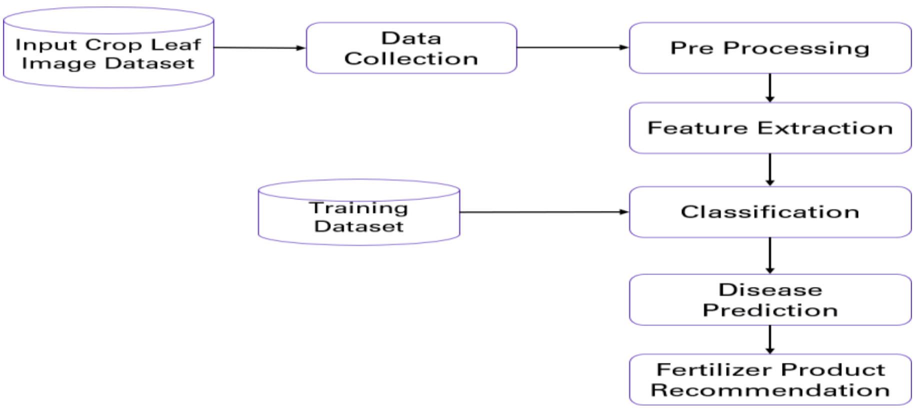

Fig. 4. Proposed Methodology

The procedures involved in the suggested approach for identifying diseases in cotton plants are shown in Fig. 4.

- **Dataset Collection**: The datasets of cotton pictures of plant leaves are gathered from the Kaggle website as well as live images from the fields with approximately 2100+ images.
- **Pre-processing:** To reduce noise from an image, pre-processing techniques are applied. The clipping photo of the leaf is used to remove the picture's region from the original image. Rescaling of the sample data is done to obtain multiple images from the single sample data.
- **Feature Extraction:** It is a dimension reduction technique that effectively portrays a piece of the image that is helpful. Here the pixels from the image are collected, and learnable weights and biases are assigned.
- **Classification and Disease Prediction:** This is achieved by applying CNN to reduce images into an easier-to-process form without compromising the characteristics that are necessary for getting a good prediction. To classify the disease in plants in a precise manner, the pictures are provided as input. The convolution layer is employed in extracting the attributes from the images. The pooling layer computes the feature values from the extracted features. Depending on the image complexity, the convolution and pooling layers can be enhanced further to extract more details. A completely linked layer turns the output of previous layers into a single vector that can be used as a source of information for the following layer. Finally, the output layer classifies the plant disease.
- **Displaying Cure:** Given the output, a plant remedy is provided by recommending suitable fertilizers to be used.

3.1 Dataset Collection

The dataset is mostly made up of leaves from the cotton plant images both of infected and healthy plants which consists of 2500 images categorized into five categories.

3.2 Model Selection

A convolutional neural network, often known as CNN or ConvNet, is particularly adept at processing input having a grid-like architecture. A digital image is a binary representation of visual data. Each neuron in the biological vision system responds to inputs only in a small area of the visual field known as the receptive field. Likewise, each neuron in a CNN only examines data in its receptive area. Because of the way the layers are configured, both simple and complexpatterns can be identified. Fig. 5. depicts the CNN model utilized in the suggested approach.

Convolutional, pooling and fully linked layers are the three layers that make up a conventional CNN[20].

- **Convolution Layer**: The convolution layer is the foundation of CNN. It carries the majority of the network's computational load. This layer performs a dot product on two matrices: one is the limited region of the receptive field, and the other is the collection of learnable parameters, also known as a kernel. The kernel is more detailed than an image but takes up less space. This means that the depth fills all three channels, but the kernel height and breadth are quite small if the image has three (RGB) channels. During the forward pass, the kernel constructs the image representation of that receptive region by sliding over the picture's height and breadth. As a result, an activation map, a two-dimensional representation of the image that depicts the kernel's response at each spatial position in the image, is created.

- **Pooling Layer:** The pooling layer replaces the network's output at locations by generating an aggregate statistic from surrounding outputs. This helps to reduce the spatial size of the representation, which reduces the amount of computation and weights required. During the pooling operation, each slice of the representation is treated independently. Pooling functions include the L2 norm of the rectangular neighborhood, the average of the rectangular neighborhood, and a weighted average calculated by the distance from the central pixel. Nonetheless, the most used option is max pooling, which offers the maximum output of the neighborhood.

- **Fully Connected Layer**: As in a traditional FCNN, all neurons in this layer are fully linked to all neurons in the layers before and following it. As a result, it can be estimated using the usual matrix multiplication and bias effect technique. The FC layer is used to map the representation between the input and the output. Because convolution is a linear process and pictures are not linear, non-linearity layers are usually included directly after the convolutional layer to add non-linearity to the activation map.

- **The Rectified Linear Unit (ReLU):** It has grown in popularity during the last few years. It is calculated by the function () =max (0,). Put otherwise, activation is just a threshold set to zero. The convergence speed of ReLU is six times faster than that of sigmoid and tanh. Regretfully, one drawback of ReLU is that it might be brittle during training. It can be updated in such a way that it never receives another update from a huge gradient passing past it. We can deal with this, though, if we establish an appropriate learning rate.

3.3 Algorithm

The algorithm used to identify illnesses and predictions is the CNN Algorithm, which is shown in Fig. 5.

Step 1: Dataset collection.
Step 2: Make feature selections with the use of information gained and ranking Step.
Step 3: Utilize the CNN algorithm for classification.
Step 4: Determine each input layer step's feature fix value.
Step 5: Determine the feature's bias class.
Step 6: After creation, the feature map is sent to the step of the forward pass input layer.
Step 7: Compute the feature pattern's convolution cores.
Step 8: Generate the feature value and subsample layer.
Step 9: The kth neuron's input deviation in the output layer is backpropagated.
Step 10: Finally, provide the categorization results with a cure.

```

#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. 5. CNN Model

Autoregressive integrated moving average is a statistical analysis method that uses time series data to forecast future events or gain a better knowledge of the data set.
The integrated moving average model with an autoregressive feature is a type of regression analysis that assesses the importance of one dependent variable in relation to other changing variables. The purpose of the model is to forecast future securities or financial market movements by studying the discrepancies between values in the series rather than actual values.
Each component of a model can be understood by outlining it as follows:

- Autoregression (AR): a model in which a changing variable regresses on its own lag, or prior, values.
- Integrated (I): depicts the differencing of raw observations for the chronology to become stationary (that is, data values are replaced by the difference between the data values and the prior values).
- Moving average (MA): reflects the relationship between a discovery and a residual error from a moving typical model applied to lagged observations.

4 Experimental Results and Analysis

This section provides the details regarding experimental results and evaluation of various approaches and methods used in the suggested framework. The complete system is tested in accordance with the specifications. It is a sort of functional testing that is based on overall requirement requirements and encompasses all system components. Table 4 mentions the comparisons of various models used for cotton identifying plant diseases.

Table 1. Comparison of Various Methods Used for Cotton Plant Disease Detection

Sl. No	Algorithm Used	Accuracy
1	CNN which uses snake segmentation	85.52%.
2	Support Vector Machine algorithm and Simple Linear Iterative Clustering algorithm	91%
3	Genetic Algorithm- based Support Vector Machine algorithm	90%
4	K-means clustering algorithm	86%
5	Data Warehousing and Mining	84.55%
6	Neural Network	83.33%
7	Image Correlation	86%
8	Proposed Methodology	95%

Fig. 6. depicts a bar graph showing the comparison of various methods and their accuracy towards the proposed system.

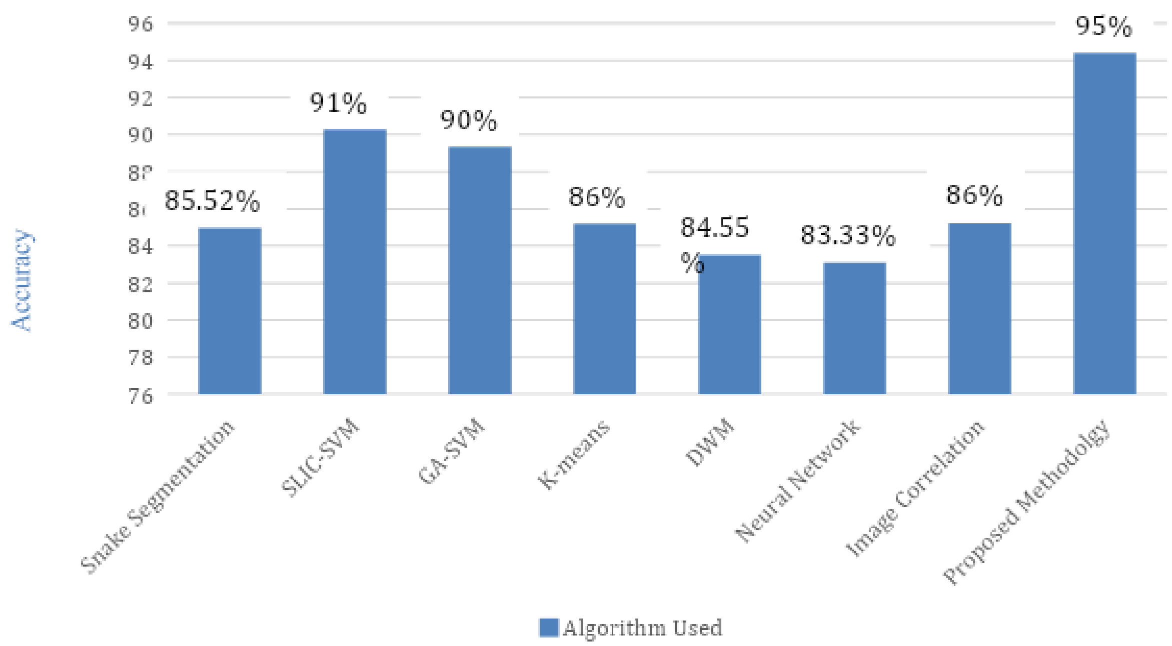

Fig. 6. Algorithm used Vs Accuracy

Fig. 7. depicts a graph that helps in visualizing how the training and validation loss, accuracy and F1 score change over the course of training.

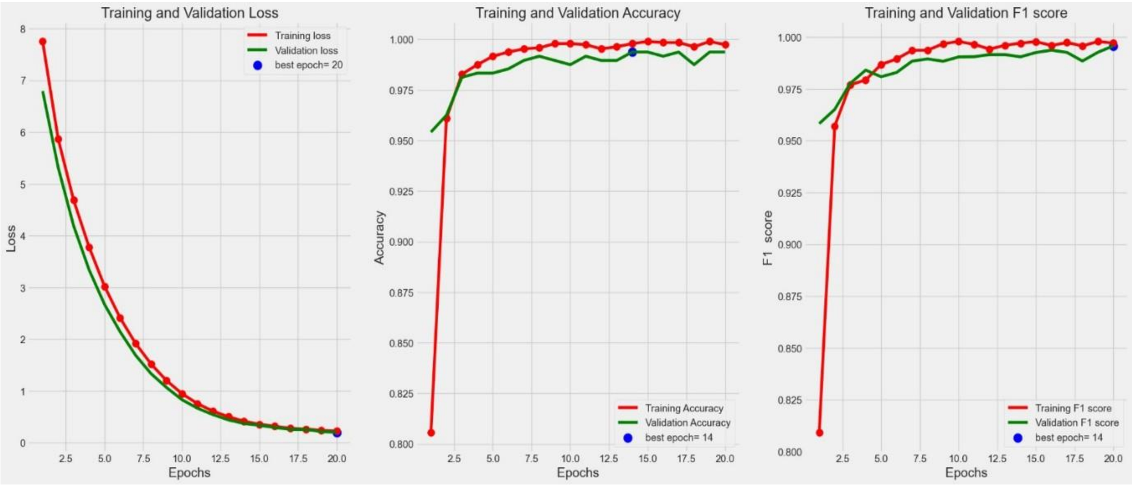

Fig. 7. Training and Validation Loss, Accuracy and F1 score

The mistake or difference between a model's anticipated output and the actual output is referred to as training loss. In the context of cotton plant disease detection, it represents to what extent the model can classify or detect diseases in the instruction dataset.

Loss, or objective function, is a numerical value that quantifies the difference between a prototype's expected output and the actual ground truth.

In the initial epochs, both the training and validation loss may be high as the prototype isn't well-trained. As the training progresses, the values of loss start to decrease. The training loss decreases as the prototype learns to fit the training data better, while the validation loss shows how effective the model can generalize to unseen data. The goal is to see a downward trend in training and validation loss. If the training loss continues to fall while the validation loss rises, this indicates overfitting, which means the model is fitting too closely to the training data and is not generalizing properly. However, if both the training and validation losses remain significant, it suggests underfitting, which occurs when the model fails to capture the patterns in the data properly.

By analyzing the graph, one can monitor the training progress, identify issues like overfitting or underfitting, and determine the optimal point where the model achieves a balance between bias and variance. It helps in making decisions such as early stopping to prevent overfitting or adjusting the model architecture or hyperparameters to improve performance.

The confusion matrix, shown in Fig. 8., summarizes the performance of a classification model by comparing the actual disease labels to the projected disease labels. It provides valuable insights into the model's accuracy and mistake rates according to various disease classes.

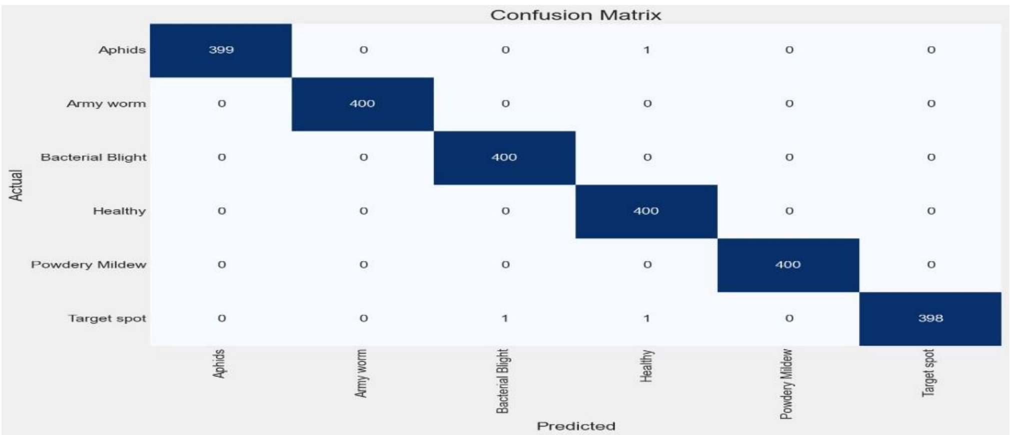

Fig. 8. Confusion Matrix

The confusion matrix is organized into rows and columns, in which every row is associated with the actual disease labels are represented by the first column, while the anticipated disease labels are represented by the second column. The matrix's cells contain the count or frequency samples falling within each category combination of the actual and predicted labels.

The confusion matrix allows for various performance indicators, such as accuracy, precision, recall (sensitivity), specificity, and F1 score, will be calculated. These metrics provide a thorough assessment of the model's performance in each illness class and overall. These metrics provide a comprehensive evaluation of the model's performance for each disease class and overall. They can help identify which illnesses are more likely to misclassification and assist in assessing the model's ability to detect and differentiate between different diseases in cotton plants.

The results of the cotton plant disease detection app are shown in Fig. 9., Fig. 10., Fig. 11. and, Fig. 12. Fig. 9. links to various options in the Apps such as select photo, start camera, detect, and get cure. Fig. 10. takes an image as input and predicts the category of disease. Fig. 11. suggests pesticides for diseases. Fig. 12. predicts organic cures for diseases like Aphids along with chemical cures.

Fig. 9. Home Page

Fig. 10. Cotton Disease Prediction Page

Fig. 11. Cure Page – Army Worm

Fig. 12. Organic Cure page - Aphids

Conclusion

Cotton leaf disease analysis is carried out to discover illnesses that are present on the leaves and can be successfully identified early on before they cause damage to the entire plant. We can say that we may achieve good production by preventing the many illnesses that are present on the leaves of the cotton plant by using meteorological datasets and image processing since the technique given here can detect the disease more precisely. The system has performed better and achieved a 95% accuracy rate thanks to the use of feature extraction and classification techniques. Future developments could see several improvements that could be implemented with the help of deep learning algorithms which could be used to detect plant diseases. These enhancements aim to improve the accuracy, efficiency, and overall performance of the disease detection system. 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.

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, Vol. 18-2, 307-324, DOI:10.13052/jmm1550-4646.1828, 29th October 2021.

2. Prashant Udawant, Pravin Srinath, "Cotton Leaf Disease Detection Using Instance Segmentation", Journal of Cases on Information Technology, 24(4):1-10, DOI: http://dx.doi.org/10.4018/JCIT.296721, October 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), DOI: https://ssrn.com/abstract=4159108, 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), Vol. 5 Issue 10, 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/v1, Vol. 1, 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", Evolutionary Computing and Mobile Sustainable Networks (pp:875-885), DOI: 10.1007/978-981-15-5258-881, January 2021.

8. Jayraj Chopda, Hiral Raveshiya, Sagar Nakum, Vivek Nakrani, "Cotton Crop Disease Detection using Decision Tree Classifier", International Conference on Smart City and Emerging Technology, 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, March 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", Computers and Electronics in Agriculture, 145:311-318, DOI: 10.1016/j.compag.2018.01.009, February 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, DOI: http://dx.doi.org/10.14445/23497157/IJRES-V7I2P106, Mar-Apr 2020.

14. Nikita Yadav, Shivani Kasar, Dnyaneshwari Abuj, Ankita Vadvale, Dr S.C. Dharmadhikar, “Crop Disease Detection and Prediction”, International Research Journal of Engineering and Technology (IRJET), Vol. 8 Issue 2, e-ISSN:2395-0056, 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, Vol. 1916, DOI: 10.1088/1742-6596/1916/1/012224, 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.

19. Chethana H.T. and Trisiladevi C Nagavi, “An Approach to Partial Occlusion using Deep Metric Learning”, International Journal of Informatics and Communication Technology, Vol.10, No.3, pp. 204-211, December 2021, ISSN:2252-8776.

20. Fig 3.1: Nidhi Prashar, and A. L. Sangal, “Plant Disease Detection Using Deep Learning (Convolutional Neural Networks)”, Second International Conference on Image Processing and Capsule Networks, Lecture Notes in Networks and Systems, Vol. 300, 10th September 2021.