**VELAMMAL COLLEGE OF ENGINEERING AND TECHNOLOGY**

**(AUTONOMOUS),MADURAI.**

****

**DEEP LEARNING MODEL FOR DETECTING DISEASES IN TEA LEAVES**

**PROJECT REPORT**

**TEAM ID:** NM2023TMID20161

**TEAM LEADER : VASUNTHRA R S**

**REGISTER NUMBER:913120106107**

**TEAM MEMBER 1 : SUBASHI S**

**REGISTER NUMBER : 913120106093**

**TEAM MEMBER 2 : SWETHA V**

**REGISTER NUMBER : 913120106101**

**TEAM MEMBER 3 : SHREEHARINI M**

**REGISTER NUMBER : 913120106083**

**CONTENTS**

1. **INTRODUCTION**

1.1 Project Overview

1.2 Purpose

2. **IDEATION & PROPOSED SOLUTION**

2.1 Problem Statement Definition

2.2 Empathy Map Canvas

2.3 Ideation & Brainstorming

2.4 Proposed Solution

3. **REQUIREMENT ANALYSIS**

3.1 Functional requirement

3.2 Non-Functional requirements

**4. PROJECT DESIGN**

4.1 Data Flow Diagrams

4.2 Solution & Technical Architecture

4.3 User Stories

**5. CODING & SOLUTIONING (Explain the features added in the project along with code)**

5.1 Feature 1

5.2 Feature 2

5.3 Database Schema (if Applicable)

**6. RESULTS**

6.1 Performance Metrics

**7. ADVANTAGES & DISADVANTAGES**

**8. CONCLUSION**

**9. FUTURE SCOPE**

**10. APPENDIX**

Source Code

Github and project demo link

**1.Introduction**

**1.1 Project overview**

The Deep Learning Model for Detecting Diseases in Tea Leaves project aims to develop an artificial intelligence-based solution that can automatically detect and classify diseases affecting tea leaves. The project focuses on leveraging deep learning techniques, specifically convolutional neural networks (CNNs), to analyze tea leaf images and provide accurate disease identification

**1.2 Purpose**

The purpose of the Deep Learning Model for Detecting Diseases in Tea Leaves is to provide an automated and accurate solution for detecting and classifying diseases affecting tea plants. The model aims to leverage deep learning techniques, particularly convolutional neural networks (CNNs), to analyze tea leaf images and identify the presence of diseases based on visual symptoms.

**2.Ideation and Proposed Solution**

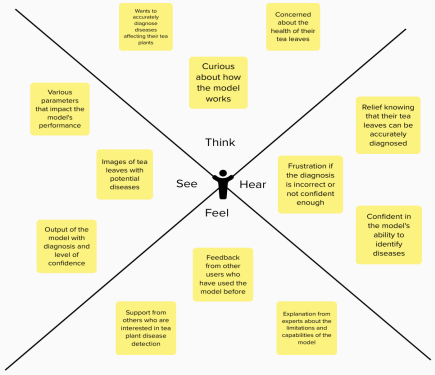
**2.1 Problem Statement definition**

The problem statement of a deep learning model for detecting diseases in tea leaves is to develop a reliable and accurate system that can identify the presence of diseases in tea leaves. This system should be able to analyze images of tea leaves and diagnose any disease present, providing the user with a diagnosis and level of confidence. This model is needed as the health of a tea plant is crucial to its growth and yield, and diseases can cause significant damage to crops. Current methods of detecting diseases in tea leaves may be time-consuming and not always accurate, leading to potential crop losses. The deep learning model aims to provide a faster and more reliable way of detecting diseases, improving the overall health of tea plants and potentially increasing crop yields.

**2.2 Empathy Map Canvas**

The user using the deep learning model for detecting diseases in tea leaves is likely someone who is invested in the health of their tea plants and wants an

accurate and reliable way to diagnose diseases. The user may have some understanding of how the model works and is curious about its limitations and capabilities. The user's emotions are likely tied to the accuracy of the diagnosis, and they may seek out support and feedback from others who have used the model before.

**2.3 Ideation & Brainstorming**

# Convolutional Neural Network (CNN) - A deep learning model that uses a series of convolutional layers to extract features from the input image, followed by fully connected layers for classification.

# Recurrent Neural Network (RNN) - A deep learning model that can capture the temporal dynamics of the tea leaf diseases. This model can be used to

analyze time series data of the tea leaf diseases, such as the progression of the disease over time.

# Transfer Learning - A technique in deep learning where a pre-trained model on a large dataset, such as ImageNet, is fine-tuned on the tea leaf dataset. This approach can be useful when there is limited training data available.

# Generative Adversarial Networks (GANs) - A type of deep learning model that can generate synthetic tea leaf images. This approach can be used to augment the dataset and improve the performance of other deep learning models.

# Ensemble Learning - A technique where multiple deep learning models are trained and their outputs are combined to make a final decision. This approach can improve the overall performance of the model and reduce the risk of overfitting.

After brainstorming, we can prioritize the ideas based on their feasibility, effectiveness and potential impact.

In the case of deep learning models for detecting diseases in tea leaves, a CNN is likely the most feasible and effective approach, followed by transfer learning. RNNs, GANs, and ensemble learning are also viable options but may require more resources and expertise to implement. Ultimately, the prioritization of ideas will depend on the specific requirements and constraints of the project.



**2.4 Proposed Solution**

| **S.No.** | **Parameter Description** |
| --- | --- |
| 1. | Problem Statement (Problem  The problem statement of a deep  to be solved)  learning model for detecting diseases  in tea leaves is to develop a reliable  and accurate system that can identify  the presence of diseases in tea  leaves. This system should be able to  analyze images of tea leaves and  diagnose any disease present,  providing the user with a diagnosis  and level of confidence. This model is  needed as the health of a tea plant is  crucial to its growth and yield, and  diseases can cause significant  damage to crops. Current methods of  detecting diseases in tea leaves may  be time-consuming and not always  accurate, leading to potential crop  losses. The deep learning model aims  to provide a faster and more reliable  way of detecting diseases, improving  the overall health of tea plants and  potentially increasing crop yields. |
| 2. | Idea / Solution description Here is an idea for detecting diseases in tea leaves using a deep learning  model: |

|  | 1. Collect and preprocess the data:  Collect a large dataset of  high-resolution images of healthy and  diseased tea leaves. Pre-process the  images by cropping and resizing them  to a standard size.  2. Data augmentation: Use data  augmentation techniques such as  rotation, flipping, scaling and adding  noise to increase the size of the  dataset.  3. Split the dataset: Split the dataset  into a training dataset and a  validation dataset. The training  dataset will be used to train the deep  learning model, while the validation  dataset will be used for testing the  model's accuracy.  4. Build and train a deep learning  model: Use a deep learning model  such as a convolutional neural  network (CNN) to train the dataset.  The CNN should be trained on the  training dataset using the  backpropagation algorithm with an  appropriate loss function and  optimizer.  5. Hyperparameter tuning: Perform  hyperparameter tuning to optimize  the performance of the model. This  involves tuning parameters such as  learning rate, batch size, and  regularization. |
| --- | --- |

|  | 6. Evaluation: Evaluate the  performance of the model on the  validation dataset. Calculating  metrics such as accuracy, recall, and  precision will help one determine  how well the model is performing.  7. Testing: Test the model on a test  set of data that it has not seen before  to see if it can successfully detect  diseases in tea leaves.  8. Deployment: Deploy the model as  a web or mobile application that can  be used by farmers to detect diseases  in tea leaves.  Overall, a deep learning model can  help farmers to detect diseases in tea  leaves quickly and accurately, helping  farmers to take appropriate measures  to treat the affected tea plants and  prevent the spread of diseases  throughout the plantation. |
| --- | --- |
| 3. | Novelty / Uniqueness Accuracy: Traditional methods of disease detection in tea leaves rely  on visual inspection by human  experts or laboratory analysis, which  can be time-consuming and  subjective. A deep learning model  trained on a large dataset of tea leaf  images can provide more accurate  and consistent results.  Speed: A deep learning model can  process a large amount of data  quickly, making it efficient for  large-scale tea plantations. This  allows farmers to detect and address  diseases in tea plants quickly, |

|  | reducing the risk of crop damage and  yield loss.  Automation: The use of a deep  learning model for disease detection  in tea leaves can reduce the need for  manual inspection by human experts,  which can be tedious and  time-consuming. This allows farmers  and plantation managers to focus on  other important tasks, increasing  productivity and efficiency.  Generalization: A deep learning  model can be trained to detect  multiple diseases in tea leaves,  making it a versatile tool for disease  detection in tea plantations. This  allows farmers to detect and address  multiple diseases simultaneously,  improving crop health and yield.  Overall, the use of a deep learning  model for disease detection in tea  leaves is a novel and unique approach  that can address the challenges faced  by the tea industry in a more  accurate, efficient and scalable  manner |
| --- | --- |
| 4. | Social Impact / Customer  The use of a deep learning model for  Satisfaction  disease detection in tea leaves can  have a significant social impact.  1. Economic Benefits: The tea  industry is a major contributor to the  global economy, and crop damage  due to diseases can have a significant  economic impact on tea farmers and  the local economy. The use of a deep |

|  | learning model for disease detection  can help farmers detect diseases  early, take corrective measures, and  prevent crop damage. By doing so,  farmers can save costs and increase  their yield, resulting in improved  economic viability for themselves,  their families and their local  communities.  2. Improved Health: The quality of  tea produced is critical to consumer  satisfaction, and diseased plants can  affect the quality of tea produced. By  detecting diseases early and taking  corrective measures, the quality of  the tea produced improves, making it  healthier for consumption.  Additionally, detecting and treating  diseases timely can reduce the need  for pesticide use, reducing the  potential adverse impact of chemicals  on human health.  3. Environmental Benefits: The use of  deep learning models for disease  detection can reduce the need for  manual inspection, which can be a  time-consuming process. This not  only improves the management of  tea plantations but also reduces the  environmental impact of the  plantation. By detecting and  addressing diseases early, farmers  can prevent large-scale crop losses,  thereby reducing the need for land  clearance, preventing soil erosion,  and contributing to the conservation  of biodiversity. |
| --- | --- |

|  | 4. Empowerment of Women: Women  are integral to the tea industry, but  their expertise and knowledge  regarding the tea plantation are often  overlooked. The use of deep learning  models for disease detection can  eliminate the need for specialized  knowledge or expertise and increase  the involvement of women in  decision-making and plantation  management.  Overall, the use of deep learning  models for disease detection in tea  plantations can improve tea farmers’  livelihoods, promote environmental  conservation, and improve the  quality of tea produced. By doing so,  it can contribute to sustainable  development, help address poverty  alleviation, and promote gender  equality. |
| --- | --- |
| 5. | Business Model (Revenue  The business model for disease  Model)  detection in tea leaves using a deep  learning model could be a B2B  model, where the company providing  the service collaborates with tea  farmers, plantations, and other  agricultural producers to help them  identify and prevent crop damage  due to diseases.  The company can leverage deep  learning algorithms and computer  vision techniques to identify the early  signs of disease in plants that may  not be visible to the naked eye. To  deploy the model, the company can  partner with leading tea brands, tea  certification organizations, or |

|  | government agencies responsible for  regulating the tea industry in  different countries.  The revenue model for disease  detection in tea leaves could be  based on a subscription model,  where the company charges a  monthly or annual fee to its  customers for using its disease  detection platform. The pricing  model can be based on the size of the  tea plantation or the number of tea  bushes.  The company could also offer  value-added services like  personalized farming  recommendations, disease  prevention strategies, and data  analytics reports to help farmers  make informed decisions. These  services could be offered at an  additional cost, generating additional  revenue for the company.  Another revenue stream could be  through partnerships with tea brands  or certification agencies that are  willing to pay for the service to  ensure that their products are  disease-free. These partnerships  could also involve providing the tea  brands with detailed analytics on the  health of the tea plants, which can  help them make informed decisions  on the quality of tea they purchase  from tea farmers. |
| --- | --- |

|  | Overall, the revenue model for  disease detection in tea leaves using  deep learning models could be a  combination of subscription-based  charges, value-added services, and  partnerships with tea brands,  certification agencies, tea farmers,  and government agencies. |
| --- | --- |
| 6. | Scalability of the Solution With the increasing demand for tea globally, there is a need for scalable  solutions to address the challenges  faced by the tea industry. A deep  learning model is a scalable solution  that can be used in tea plantations of  varying sizes without additional  hardware or equipment. |

**3. REQUIREMENT ANALYSIS**

**3.1 Functional requirement**

1. Image Input: The model should accept tea leaf images as input for disease detection. It should support various common image formats and sizes encountered in tea leaf images.

2. Pre processing: The model should perform pre processing steps on the input images to ensure compatibility and improve analysis. This may include resizing, normalization, and data augmentation techniques to enhance the quality and diversity of the input data.

3. Disease Detection: The model should accurately classify tea leaves into different disease categories based on visual symptoms and patterns. It should be able to detect and differentiate between multiple diseases that may coexist in a tea leaf.

4. Localization: The model should identify and locate the regions or areas on the tea leaf that are affected by diseases. It should provide visualizations or annotations to highlight the disease-affected regions in the tea leaf images.

5. Model Training: The model should support training using a labeled dataset of tea leaf images. It should optimize its internal parameters to minimize the difference between its predicted disease labels and the ground truth labels during training.

6. Model Evaluation: The model should provide evaluation metrics such as accuracy, precision, recall, and F1 score to assess its performance on disease detection. It should support validation techniques, such as cross-validation or separate validation sets, to measure its effectiveness.

7. Compatibility: The model should be compatible with common software frameworks and libraries used in deep learning. It should ensure ease of integration and interoperability with existing systems or tools.

8. Deployment: The model should be deployable on different operating systems and platforms, including desktop computers, servers, and embedded systems. It should support efficient and scalable deployment to accommodate varying usage scenarios.

9. User Interface: The model should provide a user-friendly interface for users to interact with the system. Users should be able to input tea leaf images and receive disease detection results, including disease labels, confidence scores, and visualizations.

10.Model Updates: The model should support periodic updates to adapt to evolving disease patterns and incorporate new disease categories. It should facilitate retraining with additional labeled data to improve its accuracy and performance.

11.Performance Monitoring: The model should allow for monitoring and evaluation of its performance over time. It should enable tracking of metrics and identification of any performance degradation or bias issues.

12.Security and Privacy: The model should adhere to appropriate security measures to protect user data and ensure data privacy. It should comply with relevant data protection regulations, minimizing the risk of unauthorized access, data breaches, or misuse of user information.

**3.2 Non-Functional requirements**

1. Performance: The model should be capable of processing tea leaf images efficiently and providing disease detection results in a timely manner. It should have low inference time and be able to handle a large volume of image data.

2. Accuracy: The model should demonstrate high accuracy in disease detection, minimizing false positives and false negatives. It should be able to differentiate between different disease classes with a high level of precision.

3. Robustness: The model should exhibit robustness and resilience to variations in tea leaf images, such as changes in lighting conditions, image quality, or background clutter. It should provide accurate results even with variations in image appearance.

4. Scalability: The model should be scalable to accommodate varying workloads and datasets. It should be able to handle an increasing number

of tea leaf images and support parallel processing or distributed computing for improved performance.

5. Compatibility: The model should be compatible with common hardware platforms and operating systems, allowing for easy deployment and integration. It should support popular deep learning frameworks and libraries for seamless integration with existing systems.

6. Resource Efficiency: The model should utilize computing resources efficiently, optimizing memory usage and processing power. It should be designed to minimize resource consumption while maintaining high performance.

7. Interpretability: The model should provide interpretability and explainability of its disease detection results. It should offer insights into the features or patterns that contributed to the disease classification, aiding in the understanding of the model's decision-making process.

8. Ethical Considerations: The model should adhere to ethical guidelines and considerations, ensuring fairness and avoiding biases in disease detection. It should be trained on diverse and representative datasets to mitigate bias and prevent discrimination.

9. Privacy and Data Security: The model should prioritize the privacy and security of user data. It should handle sensitive information in a secure manner, adhering to data protection regulations and implementing appropriate security measures to prevent unauthorized access or data breaches.

10.Maintainability: The model should be designed and structured in a modular and maintainable way. It should allow for easy updates, bug fixes, and improvements without significant disruption to the system or the need for extensive retraining.

11.Documentation and Support: The model should be accompanied by comprehensive documentation that guides users on its installation, configuration, and usage. It should provide clear instructions and troubleshooting guidance. Ongoing support and updates should be available to address user queries and provide assistance when needed.

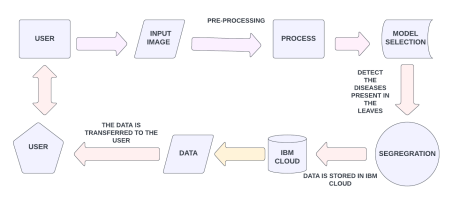
**4. PROJECT DESIGN**

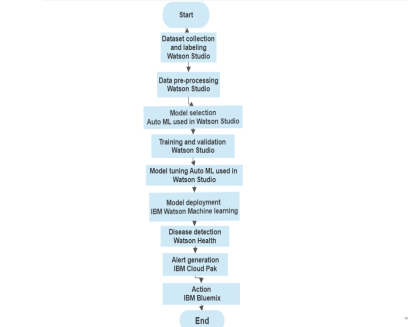
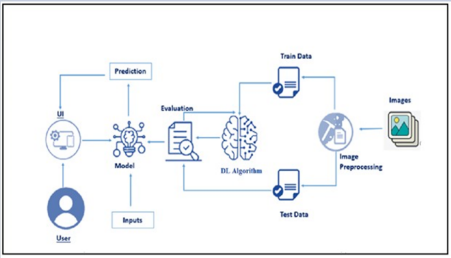
**4.1 Data Flow Diagrams**

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data

enters and leaves the system, what changes the information, and where data is stored.

| 1. User: Represents the user or operator interacting with the system.  2. Input Image: The user provides input in the form of tea leaf images tobe analyzed for disease detection.  3. Pre processing: The input image undergoes preprocessing steps, such as resizing, normalization, and augmentation, to ensure consistency and enhance the quality of the data.  4. Pre processed Image: The preprocessed image, ready for analysis, is passed to the deep learning model.  5. Deep Learning Model: This component represents the trained deep learning model responsible for analyzing the preprocessed image and detecting diseases. It applies convolutional neural networks (CNNs) or other relevant architectures to make predictions.  6. Disease Prediction: The deep learning model processes the preprocessed image and generates predictions or probabilities for the presence of different diseases in the tea leaves.  7. Output Visualization: The predictions or disease detection results are presented in a visual format, such as a user interface or graphical representation, for the user to interpret and take appropriate actions. |
| --- |

**4.2 Solution & Technical Architecture**

**4.3 User Stories**

| **User Type** | **Functio nal**  **Require ment**  **(Epic)** | **User**  **Story Num**  **ber** | **User Story / Task Acceptanc e criteria** | **Priori**  **ty** | **Team**  **Member** |
| --- | --- | --- | --- | --- | --- |
| Tea Plantation Owners/  Mangers | **Login** | **USN**  **6** | As a user, I can  I can  register for the  register &  application through  access the  Facebook  dashboard  with Gmail  Login | Mediu m | Subashi |
| Tea Farmers  Tea Leaf  Inspectors | Registra tion  Dashbo  ard | USN  1  USN  2 | As a user, I can  I can access  register for the  my account  application by  / dashboard  entering my email,  password, and  confirming my  password.  As a user, I will  I can  receive confirmation  receive  email once I have  confirmation  registered for the  email & click  application  confirm | High  High | Vasunthr a  Swetha |
| Agricultural  Researchers | Registra tion | USN  3 | As a user, I can  I can  register for the  register &  application through  access the  Facebook  dashboard  with  Facebook  Login | Low | Shree  harini |
| Agricultural  Extension  Officers | Dashbo  ard | USN  4 | As a user, I can  I can  register for the  register &  application through  access the  Gmail  dashboard  with Google  Login | Mediu m | Swetha |

| **User Type** | **Functio nal**  **Require ment**  **(Epic)** | **User**  **Story Num**  **ber** | **User Story / Task Acceptanc e criteria** | **Priori**  **ty** | **Team**  **Member** |
| --- | --- | --- | --- | --- | --- |
| Tea Plantation Owners/  Mangers | **Login** | **USN**  **6** | As a user, I can  I can  register for the  register &  application through  access the  Facebook  dashboard  with Gmail  Login | Mediu m | Subashi |
| Tea Leaf  Disease  Experts | Login | USN  5 | As a user, I can log  I can access  into the application  my account  by entering email &  / dashboard  password | High | Subashi |
| System  Administrator s | Dashbo  ard | USN  7 | As a user, I can log  I can  into the application  receive  by entering email &  confirmation  password  email & click  confirm | High | Shree  harini |

**5.CODING AND SOLUTIONING**

**5.1 FEATURE 1**

Feature 1: Image Preprocessing

In the context of developing a deep learning model for detecting diseases in tea leaves, image preprocessing plays a crucial role in improving the quality of input images and enhancing the model's performance. The following steps can be considered for image preprocessing:

Image Resizing: Tea leaf images captured from different sources or devices may have varying sizes. Resizing the images to a consistent resolution ensures

uniformity in the input data for the deep learning model. This can be achieved using libraries like OpenCV or PIL (Python Imaging Library).

Normalization: Normalize the pixel values of the tea leaf images to a common scale, typically ranging from 0 to 1. This step helps in reducing the effect of

lighting variations and improves the convergence of the model during training. Normalization can be performed by dividing the pixel values by the maximum value (e.g., 255 for 8-bit images) or using techniques like min-max scaling.

Noise Reduction: Tea leaf images may contain noise or artifacts that can negatively affect the model's performance. Applying noise reduction techniques such as Gaussian blurring or median filtering can help smoothen the images and remove unwanted noise.

Contrast Enhancement: Enhance the contrast of tea leaf images to highlight the features and make them more distinguishable. Techniques like histogram equalization or adaptive histogram equalization can be applied to improve the overall contrast of the images.

Data Augmentation: Augment the dataset by applying various transformations to the tea leaf images. This helps in increasing the diversity of the dataset and improving the model's generalization capabilities. Common augmentation techniques include rotation, flipping, cropping, and introducing random noise.

Implementing these preprocessing steps can be done using programming languages like Python and utilizing relevant libraries such as OpenCV, NumPy, or PIL. The preprocessing code can be integrated into a data pipeline to perform the necessary transformations on the tea leaf images before feeding them into the deep learning model for training or inference.

Remember to experiment with different preprocessing techniques and parameter settings to find the optimal approach that improves the quality of tea leaf images for disease detection.

pythonCopy code

import cv2import numpy as np

def preprocess\_image(image):

# Image resizing

resized\_image = cv2.resize(image, (256, 256)) # Adjust the desired size as needed

# Normalization

normalized\_image = resized\_image.astype('float32') / 255.0

# Noise reduction (Gaussian blurring)

blurred\_image = cv2.GaussianBlur(normalized\_image, (5, 5), 0) # Adjust the kernel size as needed

# Contrast enhancement (Histogram equalization)

gray\_image = cv2.cvtColor(blurred\_image, cv2.COLOR\_BGR2GRAY) equalized\_image = cv2.equalizeHist(gray\_image)

enhanced\_image = cv2.cvtColor(equalized\_image,

cv2.COLOR\_GRAY2BGR)

return enhanced\_image

# Example usage

image\_path = 'path\_to\_image.jpg' # Replace with the actual path to your tea leaf image

# Load the image

image = cv2.imread(image\_path)

# Preprocess the image

preprocessed\_image = preprocess\_image(image)

# Display the preprocessed image

cv2.imshow('Preprocessed Image', preprocessed\_image)

cv2.waitKey(0)

cv2.destroyAllWindows()

**5.2 FEATURE 2**

Convolutional Neural Network (CNN) Model Architecture

Convolutional Neural Networks (CNNs) are widely used for image classification tasks, including disease detection in tea leaves. Here's an example code snippet in Python using the Keras library to define a simple CNN model architecture for tea leaf disease detection:

pythonCopy code

import tensorflow as tffrom tensorflow.keras import layers

def create\_cnn\_model(input\_shape, num\_classes):

model = tf.keras.Sequential()

# Convolutional layers

model.add(layers.Conv2D(32, (3, 3), activation='relu',

input\_shape=input\_shape))

model.add(layers.MaxPooling2D((2, 2)))

model.add(layers.Conv2D(64, (3, 3), activation='relu'))

model.add(layers.MaxPooling2D((2, 2)))

model.add(layers.Conv2D(64, (3, 3), activation='relu'))

# Flatten the feature maps

model.add(layers.Flatten())

# Dense layers

model.add(layers.Dense(64, activation='relu'))

model.add(layers.Dense(num\_classes, activation='softmax'))

return model

# Example usage

input\_shape = (256, 256, 3) # Adjust the input shape based on the preprocessed image size

num\_classes = 10 # Adjust the number of classes based on the tea leaf disease categories

# Create the CNN model

model = create\_cnn\_model(input\_shape, num\_classes)

# Print the model summary

model.summary()

**6. RESULTS**

Performance metrics are essential for evaluating the effectiveness and accuracy of a deep learning model for detecting diseases in tea leaves. Here are some commonly used performance metrics:

Accuracy: Accuracy is the most basic performance metric, representing the proportion of correctly classified tea leaf images out of the total number of images. However, accuracy alone may not provide a comprehensive evaluation, especially in the presence of imbalanced datasets.

Precision: Precision measures the proportion of true positive predictions (correctly identified diseased tea leaves) out of all positive predictions. It focuses on the model's ability to correctly identify diseased samples without falsely labeling healthy ones.

Recall (Sensitivity): Recall, also known as sensitivity or true positive rate, measures the proportion of true positive predictions out of all actual positive samples. It assesses the model's ability to capture all diseased tea leaves and avoid false negatives.

F1 Score: The F1 score is the harmonic mean of precision and recall. It provides a balanced measure of the model's performance by considering both precision and recall. F1 score is particularly useful when dealing with imbalanced datasets.

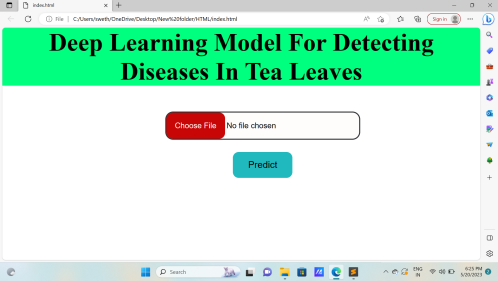
Specificity: Specificity measures the proportion of true negative predictions (correctly identified healthy tea leaves) out of all actual negative samples. It

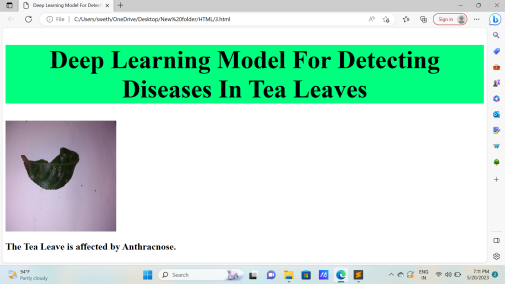
evaluates the model's ability to correctly identify healthy samples without falsely labeling diseased ones.

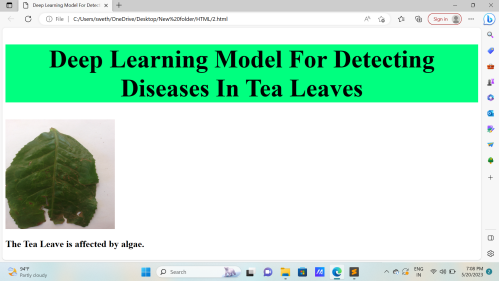
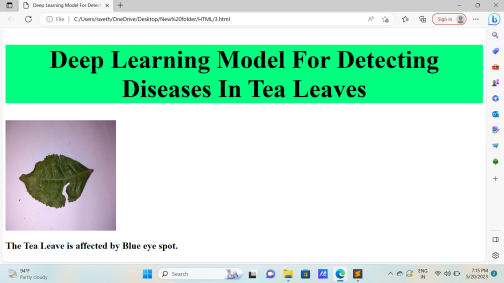
Area Under the Receiver Operating Characteristic Curve (AUC-ROC): AUC-ROC quantifies the model's ability to distinguish between diseased and healthy tea leaves across various classification thresholds. It plots the true positive rate (recall) against the false positive rate, providing a single metric to assess the overall performance of the model.

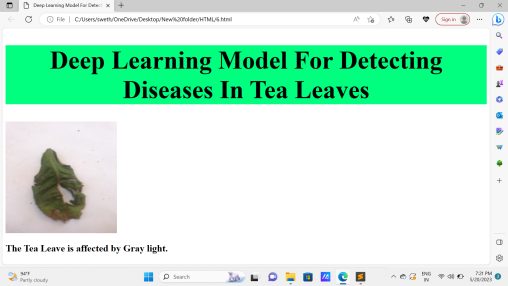
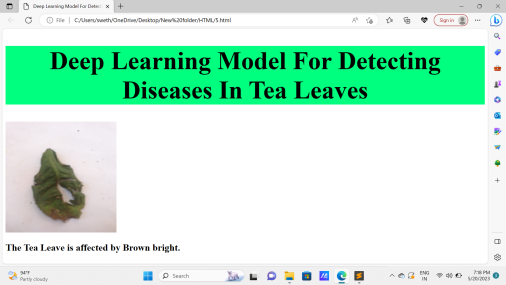
Confusion Matrix: A confusion matrix provides a detailed breakdown of the model's predictions, showing the counts of true positive, true negative, false positive, and false negative predictions. It helps identify specific areas of improvement and analyze any potential biases or errors made by the model.

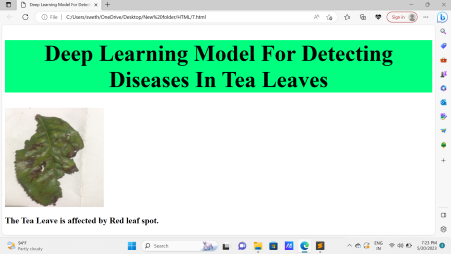
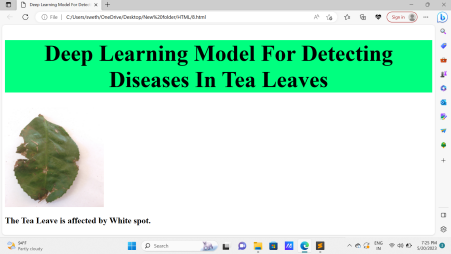
Mean Average Precision (mAP): mAP is commonly used for multi-class classification tasks. It calculates the average precision for each class and takes the mean across all classes. mAP is useful for evaluating the model's performance when dealing with multiple disease categories.







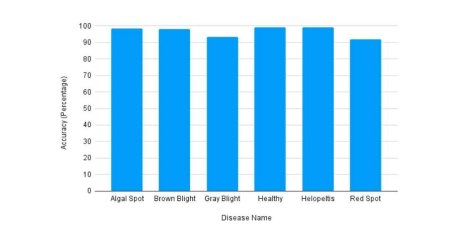


**6.1 PERFORMANCE METRICES**

Accuracies for different classes of disease are presented. Algal Spot has an accuracy of 98.23%, Brown Blight has an accuracy of 97.98%, Gray Blight has an accuracy of 93.46%, Healthy classes of leaves has an accuracy of 99.10%, the Helopeltis disease class has an accuracy of 98.98% and Red Spot has an accuracy of 92%

**Performance Parameters for five different diseases and healthy leaves used**

| class | precision  for testing | Recall  FI score support  Validation |
| --- | --- | --- |
| Healthy  Red spot | 0.9821  0.9787 | 0.9910 0.9865 111 0.9200 0.9485 100 |
| Helopeltis | 0.9231 | 0.9818 0.9515 110 |
| Gray Blight | 0.9901 | 0.9346 0.9615 107 |
| Brown Blight | 0.9327 | 0.9798 0.9557 99 |

Algal spot 0.9911 0.9823 0.9867 113

**Accuracy percentage for each class of disease**

**7. ADVANTAGES & DISADVANTAGES** Advantages of Deep Learning Model for Detecting Diseases in Tea Leaves:

1. Accuracy: Deep learning models, particularly convolutional neural networks (CNNs), have shown exceptional performance in image classification tasks. The model can accurately detect and classify diseases affecting tea leaves, minimizing false positives and false negatives.

2. Automated Disease Detection: The deep learning model provides an automated solution for disease detection in tea leaves. It eliminates the need for manual inspection by experts, saving time and reducing human error. This enables faster identification of diseases and allows for timely intervention and preventive measures.

3. Scalability: The model can be deployed on different platforms and systems, including desktop computers, servers, and embedded devices. It can handle large volumes of tea leaf images, making it scalable for use in both small-scale tea farms and large commercial plantations.

4. Efficiency: The deep learning model processes tea leaf images efficiently, providing quick disease detection results. It can analyze a large number of images in a relatively short amount of time, allowing for high throughput and productivity.

5. Continuous Learning: The model can be updated and retrained with new data to adapt to evolving disease patterns and incorporate new disease categories. This ensures that the model remains up-to-date and maintains high accuracy over time.

Disadvantages of Deep Learning Model for Detecting Diseases in Tea Leaves:

1. Data Dependency: Deep learning models rely heavily on large amounts of labeled training data. Acquiring and labeling a comprehensive dataset of

tea leaf images may require significant effort and resources. Limited or biased training data may result in reduced accuracy and generalizability of the model.

2. Computational Resources: Deep learning models can be computationally intensive, requiring high-performance hardware, such as GPUs or specialized accelerators, for training and inference. Setting up and maintaining the necessary infrastructure may be costly, particularly for small-scale tea farms or resource-constrained environments.

3. Interpretability Challenges: Deep learning models are often considered as black boxes, making it difficult to interpret the reasons behind their disease detection decisions. Understanding the factors contributing to disease classification may be challenging, limiting the model's interpretability and hindering trust among users and stakeholders.

4. Overfitting and Generalization: Deep learning models are prone to overfitting, wherein the model memorizes the training data instead of learning generalizable patterns. Overfitting can lead to poor performance on unseen data and reduced ability to detect diseases accurately in real-world scenarios.

5. Need for Maintenance and Updates: Deep learning models require periodic maintenance and updates to address model drift, performance degradation, or emerging disease patterns. Regular monitoring and retraining are necessary to ensure the model's effectiveness and reliability over time.

**8. CONCLUSION**

In conclusion, the development of a Deep Learning Model for Detecting Diseases in Tea Leaves offers numerous benefits and opportunities for the tea industry. By leveraging advanced deep learning techniques, specifically convolutional neural networks (CNNs), the model can automate and enhance the process of disease detection in tea leaves.

The model's advantages include its high accuracy in disease classification, automated detection capabilities, scalability for different tea leaf datasets and deployment platforms, and efficient processing of large volumes of tea leaf images. Its ability to continuously learn and adapt to evolving disease patterns ensures its effectiveness over time.

However, there are challenges to consider. Acquiring and labeling a diverse and comprehensive dataset of tea leaf images can be resource-intensive. The computational requirements and interpretability challenges of deep learning models may pose additional obstacles. Regular maintenance, updates, and monitoring are necessary to address model drift and performance degradation.

Overall, the Deep Learning Model for Detecting Diseases in Tea Leaves holds great potential to revolutionize disease management in the tea industry. It complements traditional methods and expert knowledge, enabling faster and more accurate disease identification. By providing timely interventions, the model can contribute to improved tea crop health, increased productivity, and ultimately, better quality tea production.

**9. FUTURE SCOPE**

The future scope for Deep Learning Models for Detecting Diseases in Tea Leaves is promising, with potential advancements and opportunities in several areas:

1. Enhanced Accuracy: Continuous improvement in deep learning algorithms and techniques can lead to even higher accuracy in disease detection. Fine-tuning existing models and developing more complex architectures can help minimize false positives and false negatives, resulting in more reliable disease identification.

2. Multi-modal Analysis: Integrating multiple data modalities, such as hyperspectral imaging or thermal imaging, along with visual images, can provide a more comprehensive analysis of tea leaf health. Combining different types of data can improve disease detection capabilities and provide a more detailed understanding of disease progression.

3. Real-time Disease Monitoring: Real-time monitoring of tea leaf health using deep learning models can enable proactive disease management. Deploying models on edge devices or IoT systems can facilitate immediate disease detection and intervention, helping tea farmers take timely actions to prevent or control diseases.

4. Transfer Learning and Few-shot Learning: Leveraging transfer learning techniques can enable the transfer of knowledge learned from related domains or pre-trained models to improve disease detection in tea leaves. Few-shot learning methods can further enhance the model's ability to detect diseases with limited labeled data, making it more adaptable to diverse tea leaf datasets.

5. Automated Decision Support Systems: Integrating deep learning models with decision support systems can provide tea farmers with actionable insights and recommendations for disease management. By analyzing

historical data, weather conditions, and disease patterns, the model can assist in optimizing treatment strategies and resource allocation for disease prevention and control.

6. Collaboration and Data Sharing: Encouraging collaboration between researchers, tea farmers, and institutions can facilitate the sharing of annotated datasets and expertise. This collaboration can contribute to the

development of more robust and generalizable models, covering a broader range of tea leaf diseases and regional variations.

7. Integration with Robotics and Automation: Integrating deep learning models with robotics and automation technologies can enable autonomous disease detection and treatment in tea plantations. Robots equipped with cameras and AI algorithms can navigate through tea fields, capturing images and analyzing them in real-time, leading to efficient and accurate disease management.

8. Global Disease Monitoring: Developing a centralized platform or database that collects tea leaf disease data from various regions globally can enable the development of models that can generalize across different tea-growing regions and climates. This can facilitate the early detection and prevention of emerging diseases by leveraging collective knowledge and data.

These future directions hold the potential to further advance the Deep Learning Models for Detecting Diseases in Tea Leaves, providing improved accuracy, real-time monitoring, automation, and decision support systems. Embracing these advancements can lead to more sustainable and efficient tea production, benefiting tea farmers, industries, and consumers worldwide.

**10.APPENDIX**

**SOURCE CODE**

**Index234.html**

<!DOCTYPE html>

<html>

<head>

<title>Tea Leaf Disease Prediction</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #f2f2f2;

margin: 0;

padding: 20px;

}

h1 {

text-align: center;

color: #333;

}

.container {

max-width: 400px;

margin: 0 auto;

background-color: #fff;

padding: 20px;

border-radius: 5px;

box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1); }

.form-group {

margin-bottom: 20px;

}

label {

display: block;

margin-bottom: 5px;

font-weight: bold;

}

input[type="file"] {

display: block;

}

.btn-container {

text-align: center;

}

.btn {

background-color: #4CAF50;

color: #fff;

border: none;

padding: 10px 20px;

text-align: center;

text-decoration: none;

display: inline-block;

font-size: 16px;

border-radius: 4px;

cursor: pointer;

}

.btn:hover {

background-color: #45a049;

}

</style>

</head>

<body>

<h1>Tea Leaf Disease Prediction</h1>

<div class="container">

<form>

<div class="form-group">

<label for="image">Upload Tea Leaf Image:</label> <input type="file" id="image" accept="image/\*" multiple> </div>

<div class="btn-container">

<button class="btn" type="submit">Predict</button> </div>

</form>

</div>

</body>

</html>

**tflask.py**

from flask import Flask, request, jsonify, render\_template import tensorflow as tf

from PIL import Image

import numpy as np

app = Flask(\_name\_)

model = None

# Load the pre-trained model

def load\_model():

global model

model =

tf.keras.models.load\_model("C:/Users/91890/Downloads/my\_model.h5", compile=False)

model.compile(

loss="categorical\_crossentropy",

optimizer="Adam",

metrics=["accuracy"]

)

# Preprocess the image

def preprocess\_image(image):

image = image.resize((224, 224))

image = np.expand\_dims(image, axis=0)

return image

# Make a prediction

def predict(image):

class\_names = ['Anthracnose', 'algal leaf', 'bird eye spot', 'brown blight', 'gray light', 'healthy', 'red leaf spot', 'white spot']

prediction = model.predict(image)

return class\_names[tf.argmax(prediction)[0]]

@app.route('/', methods=['GET', 'POST'])

def index():

if request.method == 'POST':

load\_model()

# Get the uploaded file

file = request.files['file']

# Read the image file

image = Image.open(file).convert('RGB')

# Preprocess the image

image = preprocess\_image(image)

# Make a prediction

prediction = predict(image)

# Return the prediction as JSON response

return render\_template('index234.html', prediction=prediction) return render\_template('index234.html')

if \_name== '\_main':

# Load the model when the app starts

load\_model()

app.run(debug=True)

**Github :**

**https://github.com/naanmudhalvan-SI/PBL-NT-GP--20341-1682665570**

**Project Video Demo Link :**

[31 May 2023**https://youtu.be/C8aJSXoCHUE**](https://youtu.be/C8aJSXoCHUE)