Smart Crop AI

Himanshu Sharma

Abstract

“This report introduces SmartCrop AI, a machine learning application tailored to aid small and medium-sized farms in diagnosing plant diseases. Addressing the critical need for advanced yet accessible agricultural tools, SmartCrop AI combines user-friendly design with powerful image analysis technology. Developed through a Flask framework, the system uses a convolutional neural network to provide swift and accurate disease detection from plant images. Early testing showcases the model's effectiveness, promising a valuable resource for enhancing crop management and supporting sustainable farming practices. The report captures the development lifecycle, from conception to prototype validation, and reflects on the tool's potential impact on agricultural productivity.”

Problem Statement

- **Challenge**: Small to medium-sized farms struggle with resource optimization and crop management due to limited access to advanced agricultural technologies. This leads to inefficient farming practices and lower yields.

- **Objective:** SmartCrop AI aims to provide an affordable, easy-to-use AI solution for these farms. It will leverage machine learning to offer actionable insights for improving crop health, resource management, and adapting to market changes, thereby enhancing yield and profitability.

Need Assessment

- **Customer Needs:**
    - Efficient Crop Management: Tool for informed decision-making in crop cultivation and pest control.
    - Risk Mitigation: Capabilities to anticipate and respond to environmental risks. Resource Optimization: Assistance in efficient use of resources like water and fertilizers.

- **Business Opportunity:**
    - Customization: Demand for AI solutions tailored to small-scale farming operations.
    - Affordability: A gap in the market for cost-effective AI agricultural tools.
    - Sustainability: Growing global focus on sustainable farming practices and food security.

Target Specifications and Characterization

- **Target Customers:** Target Customers: Small to medium-sized farm owners, primarily in rural or semi-urban areas with varying levels of technical proficiency and limited technology budgets.

- **Characterization of Needs:**
  - Usability: An intuitive, user-friendly interface for effortless navigation.
  - Functionality: Delivering valuable insights for efficient crop management, optimal resource utilization, and adapting to dynamic market trends.
  - Compatibility: Seamless integration with existing agricultural systems and equipment.
  - Scalability: Flexible and adaptable to accommodate a variety of farm sizes and agricultural practices.

- **Customer Expectations:**
  - Reliability: Accurate, actionable insights.
  - Support: Offering robust, easily reachable customer support and service.
  - Affordability: Cost-effective solution.

- **Success Metrics:**
  - Adoption Rate: Tracking the uptake and usage of SmartCrop AI among targeted farms.
  - Yield Improvement: Measuring the tangible impact on crop yields and resource management efficiencies.
  - Customer Satisfaction: Gaining positive feedback and high satisfaction levels from users.

External Search

- **Online Information Sources:**
    - Academic and Research Platforms: Accessing journals like "Journal of Agricultural Informatics" and "Computers and Electronics in Agriculture" for AI-based farming research.
    - Industry Analysis Portals: Exploring market reports on AI in agriculture from McKinsey & Company, Gartner, IBISWorld, and Statista.

- **References:**
    - Case Studies: Analyses of AI implementation in small to medium-sized farms.
    - Government and Educational Publications: Latest research from USDA and agricultural universities.
    - Community Discussions: Insights from farming forums and social media groups.

- **Links:**
    - 'Potato Disease Classification' Playlist by 'codebasics': https://youtube.com/playlist?list=PLeo1K3hjS3ut49PskOfLnE6WUoOp_2lsD&si=QqsP1GsmKXs5bHia
    - GitHub Repository By 'codebasics': https://github.com/codebasics/potato-disease-classification
    - Crop leaf disease detection and classification using machine learning and deep learning algorithms by visual symptoms: https://www.degruyter.com/document/doi/10.1515/comp-2020-0122/html

- **<u>Plantix</u>**

Strengths Relative to SmartCrop AI:

- <u>Specialized Disease Identification</u>: Higher accuracy for plant diseases, beneficial for disease management in crops.
- <u>Treatment Recommendations</u>: Valuable for farmers needing actionable advice on disease treatment.
- <u>Community Support</u>: A platform for farmers to exchange insights could be an added advantage for SmartCrop AI.

Weaknesses as Opportunities for SmartCrop AI:

- <u>Focused Scope</u>: SmartCrop AI could offer broader functionalities beyond disease identification, like crop yield optimization and resource management. <u>Misdiagnoses Risk</u>: SmartCrop AI can integrate multiple data points (weather, soil conditions) to reduce misdiagnosis chances.
- <u>Premium Features</u>: SmartCrop AI can aim for a more inclusive range of free features to attract budget-conscious farmers.

Continue…

- **PlantSnap:**

Strengths Relative to SmartCrop AI:

- Broad Plant Database: SmartCrop AI can incorporate a similar extensive database for diverse crop identification.
- User-Friendly Design: Emphasizing an intuitive interface in SmartCrop AI to match PlantSnap's ease of use.
- Plant Information: Offering detailed crop care and growth information could be a unique selling point for SmartCrop AI.

Weaknesses as Opportunities for SmartCrop AI:

- Disease Identification: SmartCrop AI can focus more on disease identification and solutions, an area where PlantSnap is less robust.
- Accuracy in Disease Diagnosis: Leveraging high-accuracy algorithms for disease diagnosis in SmartCrop AI.
- Free vs. Premium Features: Providing more comprehensive features in the free version of SmartCrop AI to stand out.

Continue…

- **Conclusion**: SmartCrop AI can carve a niche in the agricultural AI market by combining Plantix's disease identification strengths with PlantSnap's extensive plant database and user-friendly design, while also addressing their respective weaknesses. This includes offering a more comprehensive range of free features, integrating broader functionalities like yield optimization, and providing detailed crop care information.

Applicable Patents

- **Focus**: Identifying patents related to AI in agriculture, particularly for crop monitoring and disease diagnosis, to guide SmartCrop AI's development and ensure legal compliance.

- **Method**: Conducting searches in patent databases and consulting legal experts for insights and validation.

- **Goal**: To ensure SmartCrop AI aligns with existing patents and identifies areas for potential innovation and new patent filings.

Applicable Regulations

- **Focus**: Ensuring SmartCrop AI complies with data protection laws, agricultural standards, environmental regulations, and AI ethics.

- **Method**: Conduct legal research and consult experts to align SmartCrop AI with all relevant regulations and ethical guidelines.

- **Goal**: To maintain legal and ethical integrity, ensuring responsible and sustainable use of AI in agriculture.

Applicable Constraints

- **Data Collection:** Adapting to varied farming environments.
- **Technical Accessibility:** User-friendly design for farmers with varying tech skills.
- **Budget-Friendly:** Affordable for small farm budgets.
- **Versatility:** Effective across various crops and farming practices.
- **User Adoption:** Simplifying technology adoption for farmers.

Goal: To ensure SmartCrop AI is accessible, practical, and valuable for small to medium-sized farms.

Business Model

- **Subscription-Based Service:**
  - Tiered Plans: Offer different subscription levels, catering to various farm sizes and needs. Basic plans could include essential monitoring features, while premium plans offer advanced analytics and personalized recommendations.

  - Free Trial Period: Encourage adoption by providing a free trial period, allowing farmers to experience the value of the service before committing.

- **Pay-Per-Use Services:**
  - Custom Analysis: For farmers who need occasional in-depth analysis, such as soil health or pest risk assessment.

  - On-Demand Consultations: Access to experts for personalized guidance and problem-solving.

- **Data Insights and Reports:**
  - Sale of Aggregated Data: Offer anonymized, aggregated data insights to agricultural researchers or companies, while adhering to privacy regulations.

  - Customized Reports: Provide detailed reports and analytics for a fee, helping farmers with long-term planning and strategy.

Continue…

- **Integrated Marketplace:**
  - Agricultural Products: Sell seeds, fertilizers, and equipment recommended by SmartCrop AI, tailored to the specific needs of each farm.

  - Partnerships with Suppliers: Collaborate with agricultural suppliers to offer discounts or exclusive products to SmartCrop AI users.

- **Community Engagement and Advertising:**
  - Sponsorship and Ads: Partner with agricultural brands for targeted advertising within the platform. Community
  - Forums: A premium community feature where farmers can share insights, with potential for sponsored content and brand partnerships.

- **Government and NGO Collaborations:**
  - Subsidized Access: Partner with government bodies or NGOs to offer subsidized or free access to SmartCrop AI in underprivileged regions, enhancing sustainability and food security.

- **Educational Workshops and Webinars:**
  - Training Programs: Offer paid workshops or webinars on sustainable farming practices, leveraging AI for agriculture, etc.

Concept Generation

Process:

- **Collaboration Sessions:** Engage with agricultural experts, AI developers, and farm owners, especially focusing on those from rural areas, to understand firsthand the challenges they face, such as crop damage and resource limitations.
- **Market Research:** In-depth analysis of current trends and technological gaps in agricultural technology, with a special focus on how these challenges are more pronounced in rural settings.
- **Observational Studies:** Conduct field visits and observational studies in rural areas to directly witness the issues farmers face, such as crop damage due to pests, diseases, or unoptimized farming practices. Idea
- **Brainstorming:** Host workshops with the project team and stakeholders to brainstorm solutions that specifically address crop damage and other critical issues identified during collaboration sessions and field studies.
- **Feedback Loops:** Incorporate feedback from rural farmers to refine the concept, ensuring the solution is tailored to their unique challenges and needs.

Goals: To develop a concept for SmartCrop AI that not only leverages AI technology but also directly addresses the real-world problems of crop damage and resource management, especially in rural areas. To create a solution that is both technologically innovative and deeply rooted in the practical realities and needs of farmers in rural settings.

Concept Development

SmartCrop AI is an AI-driven solution designed to assist small to medium-sized, especially rural, farms in optimizing crop management and mitigating damage. The service will utilize advanced machine learning algorithms to analyze data from various sources, such as soil conditions, weather patterns, and crop health indicators.

Key Features:

- **Predictive Analysis:** Using AI to forecast potential crop diseases and pest infestations, enabling early intervention.
- **Customized Recommendations:** Offering tailored advice on crop rotation, irrigation, and fertilizer usage based on individual farm data.
- **Integration Capability:** Designed to seamlessly integrate with existing farm management systems and IoT devices used in agriculture.
- **User-Friendly Interface:** A simple, intuitive dashboard accessible even to farmers with limited technical expertise.
- **Scalable and Adaptable:** Flexible enough to be effective for various crops and adaptable to different farm sizes and practices.

Development Approach:

- **Iterative Process:** Emphasizing agile development with continuous testing, feedback incorporation, and refinements based on real-world usage.
- **Collaborative Design:** Involving end-users in the design process to ensure the product meets their specific needs and challenges.

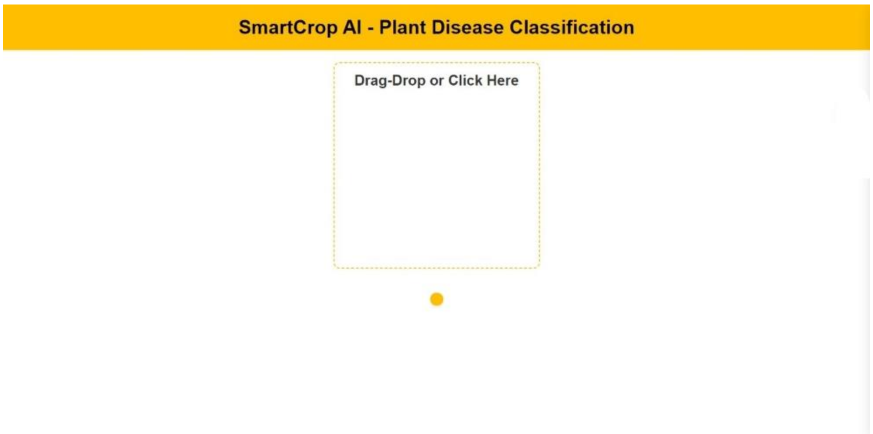
Abstract Description: SmartCrop AI is a Flask-based web application designed to provide predictive agricultural insights to small and medium-sized farms. It combines AI-driven analytics with a user-friendly interface, offering real-time data and suggestions for improved crop management and resource optimization.

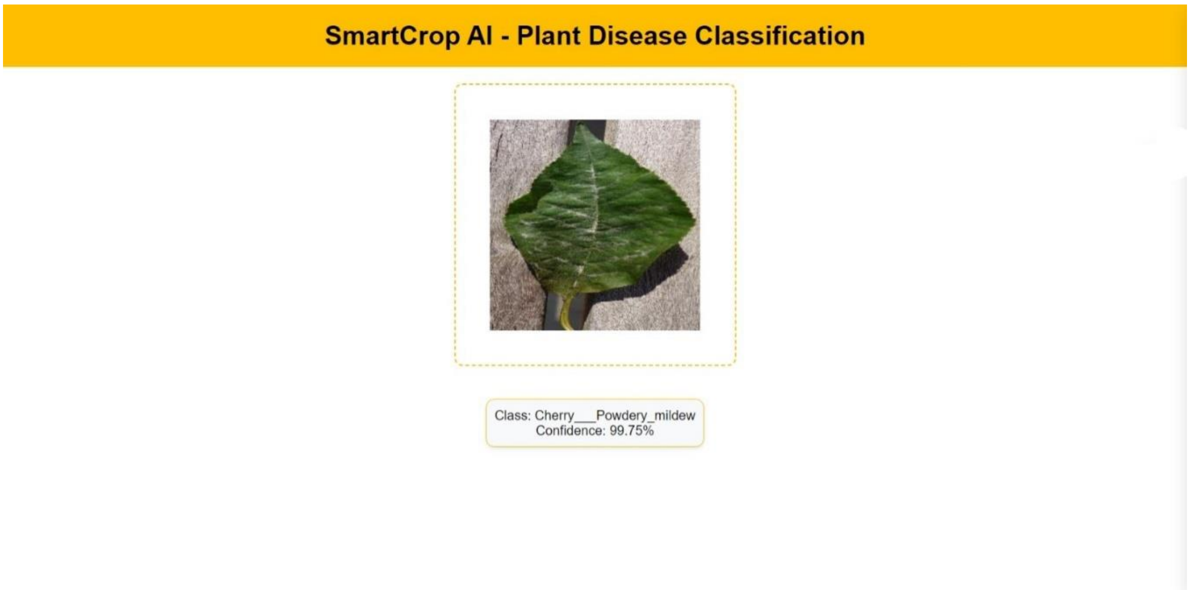
Schematic Diagram Overview:

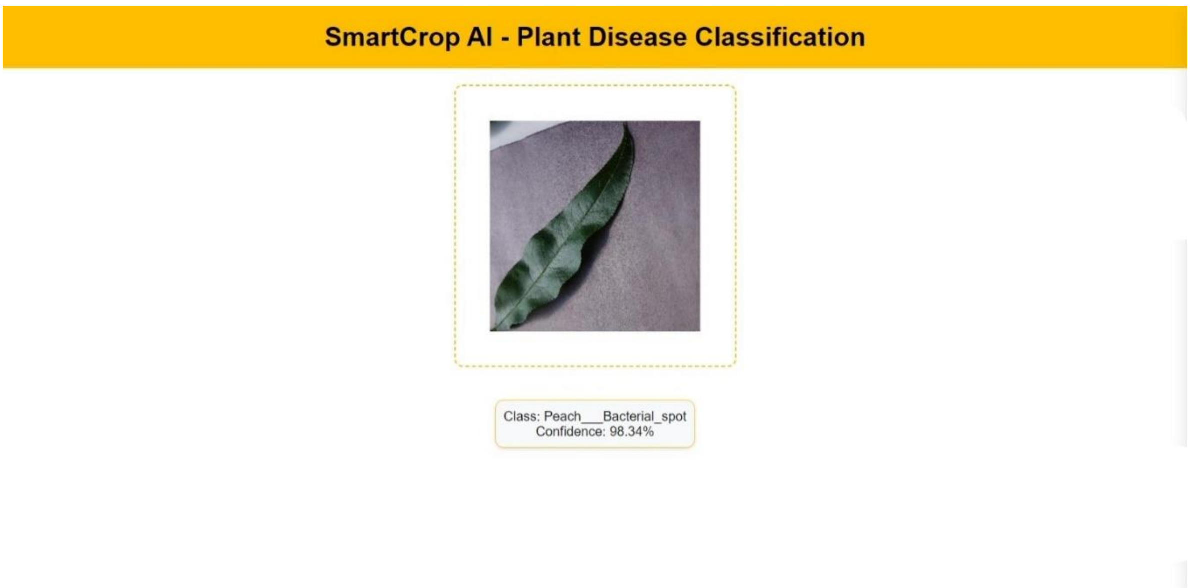
The system architecture consists of several key components:

- **Web Interface:** Provides a user-friendly platform for farmers to interact with SmartCrop AI. Delivers a responsive design for both desktop and mobile access, ensuring ease of use regardless of device.
- **Flask Web Server:** Serves as the backbone of the application, managing requests and responses between the user interface and the AI analytics engine. Handles all server-side logic including data processing and responding to user inputs. AI
- **Analytics Engine:** Integrates with the web server, processing data from various sources like IoT devices, satellite imagery, and user inputs. Employs machine learning algorithms to generate predictive insights about crop health, potential diseases, and best farming practices.
- **Data Storage and Management:** Safely stores user data, farm data, and the results from the AI engine for analysis and historical reference. Ensures data integrity and security.
- **External API Integrations:** Connects to external APIs for additional functionalities like weather forecasting and accessing remote sensing data.

Continue…


SmartCrop AI - Plant Disease Classification
Drag-Drop or Click Here


SmartCrop AI - Plant Disease Classification
Class: Cherry___Powdery_mildew
Confidence: 99.75%


SmartCrop AI - Plant Disease Classification
Class: Peach___Bacterial_spot
Confidence: 98.34%

Product Details

Core Functionality: SmartCrop AI is currently focused on plant disease classification. It helps farmers identify potential diseases affecting their crops through image analysis.

How It Works:

- **Data Input:** Farmers upload images of their crops via the web interface.
- **Image Analysis:** The uploaded images are processed by the Flask backend, which interfaces with the machine learning model.
- **Disease Identification:** The AI model, utilizing image recognition algorithms, analyses the images to identify signs of disease.
- **Output:** The system provides the farmer with the results, identifying the type of disease and offering basic guidance on potential remedies.

Frontend Development:

- **Design:** Simple and intuitive, tailored for easy image uploading and results viewing.
- **Technologies:** Built with HTML, CSS, JavaScript; React may be used for enhanced interactivity.
- **Features:** A straightforward interface for image upload, a section for displaying results, and an informational guide about common plant diseases.

Continue…

Backend Development:

- **Framework:** Python Flask, managing image data processing and integration with the AI model.
- **Image Processing:** Efficient handling of image data, ensuring quality and compatibility with the ML model.

Machine Learning Model:

- **Algorithm:** Specialized in image recognition, likely using convolutional neural networks (CNNs) to classify plant diseases from images.
- **Model Training:** Utilizes a dataset of plant images labelled with various disease types for training accuracy.

Data Source:

- **User-Provided Images:** Photographs of crops uploaded by farmers, serving as the primary data source for disease classification.

Team and Development Costs:

- **Team Composition:** Involves software developers, machine learning engineers, and agricultural experts for domain-specific guidance.
- **Budgeting:** Consideration for development, operational, and potential scaling costs.

Intended Application: This initial version of SmartCrop AI aims to assist farmers in early disease detection, a crucial step in effective crop management. The tool's functionality will help in reducing crop damage and improving overall yield quality.

Overview: The Plant Disease Classification system is developed using Python and TensorFlow, and the codebase is structured to facilitate training, validation, and testing of a machine learning model that can identify and classify plant diseases from images.

Data Handling:

- The dataset is split into training, validation, and testing subsets, ensuring a proper evaluation of the model's performance.
- Image data is pre-processed to a uniform size, and data augmentation techniques are applied to create a more robust dataset for training.

```
# Set Constants
BATCH_SIZE = 32
IMAGE_SIZE = (256, 256)
CHANNELS = 3
EPOCHS = 30
```

```
# Define a function to split the dataset
def get_dataset_partitions_tf(ds, train_split=0.8, val_split=0.1, test_split=0.1, shuffle=True, shuffle_size=10000):
    assert (train_split + test_split + val_split) == 1

    ds_size = len(ds)

    if shuffle:
        ds = ds.shuffle(shuffle_size, seed=12)

    train_size = int(train_split * ds_size)
    val_size = int(val_split * ds_size)

    train_ds = ds.take(train_size)
    val_ds = ds.skip(train_size).take(val_size)
    test_ds = ds.skip(train_size).skip(val_size)

    return train_ds, val_ds, test_ds
```

```
class_names = dataset.class_names
class_names
```

```
['Cherry___Powdery_mildew',
 'Cherry___healthy',
 'Peach___Bacterial_spot',
 'Peach___healthy',
 'Pepper__bell___Bacterial_spot',
 'Pepper__bell___healthy',
 'Strawberry___Leaf_scorch',
 'Strawberry___healthy']
```

Continue…

Model Architecture:

- The model is based on the EfficientNet B0 architecture, which is pre-trained on the ImageNet dataset.
- This model is fine-tuned for the classification task with additional layers to adapt it to plant disease identification.

```
# Data Augmentation
data_augmentation = tf.keras.Sequential([
    layers.experimental.preprocessing.RandomFlip("horizontal_and_vertical"),

    layers.experimental.preprocessing.RandomContrast(factor=(0.5, 1.5)),

    layers.experimental.preprocessing.RandomRotation(0.2),
    layers.experimental.preprocessing.RandomZoom(0.2),
])

train_ds = train_ds.map(
    lambda x, y: (data_augmentation(x, training=True), y)
).prefetch(buffer_size=AUTOTUNE)
```

```
# Model Architecture
input_shape = IMAGE_SIZE + (CHANNELS,)
n_classes = len(class_names)

# Load EfficientNet B0 pre-trained on ImageNet
base_model = tf.keras.applications.EfficientNetB0(
    include_top=False, weights='imagenet', input_shape=input_shape
)

# Freeze only the bottom layers for faster convergence
for layer in base_model.layers[:100]:
    layer.trainable = False

# Create a sequential model with base model and new layers
model = tf.keras.Sequential([
    base_model,
    layers.GlobalAveragePooling2D(),
    layers.Dense(128, activation='relu'),
    layers.Dense(n_classes, activation='softmax')
])
```

Continue…

Training Process:

- The model is trained using the prepared dataset, with the batch size and epochs set to optimize learning.
- During training, data augmentation is applied to the input images, enhancing the model's ability to generalize.

```
# Compile the model
model.compile(
    optimizer='adam',
    loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=False),
    metrics=['accuracy']
)
```

```
# Define an EarlyStopping callback
early_stopping = callbacks.EarlyStopping(
    monitor='val_loss',     # Monitor validation loss
    patience=5,              # Stop after 5 epochs of no improvement
    restore_best_weights=True  # Restore best model weights when stopping
)

# Train the model with early stopping
history = model.fit(
    train_ds,
    batch_size=BATCH_SIZE,
    validation_data=val_ds,
    verbose=1,
    epochs=EPOCHS,
    callbacks=[early_stopping]  # Include the EarlyStopping callback
)
```

Continue…

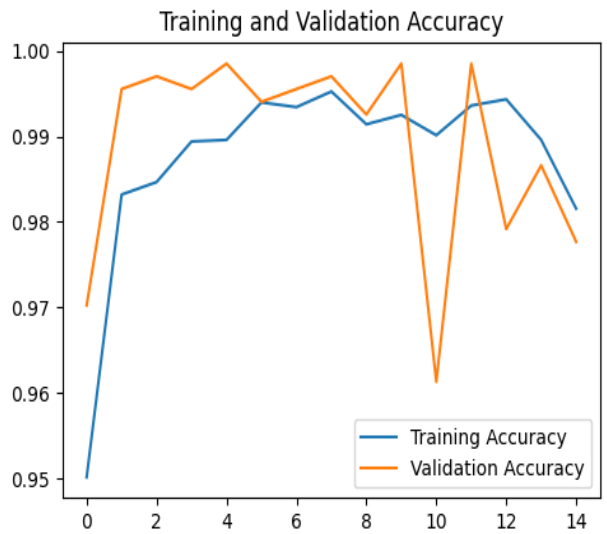
Validation and Testing:

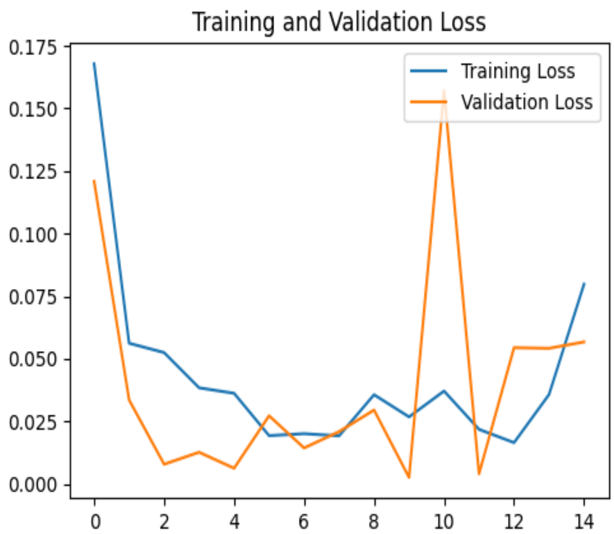
- Early stopping is employed to halt the training if the validation loss does not improve, preventing overfitting.
- After training, the model's performance is evaluated on the test set to assess its accuracy and loss, providing an indication of how well it will perform in real-world scenarios.

```
# Evaluate the model on the test dataset
scores = model.evaluate(test_ds)

print("Test loss:", scores[0])
print("Test accuracy:", scores[1])
```

```
23/23 [==============================] - 182s 517ms/step - loss: 0.0048 - accuracy: 0.9986
Test loss: 0.004824520088732243
Test accuracy: 0.998641312122345
```

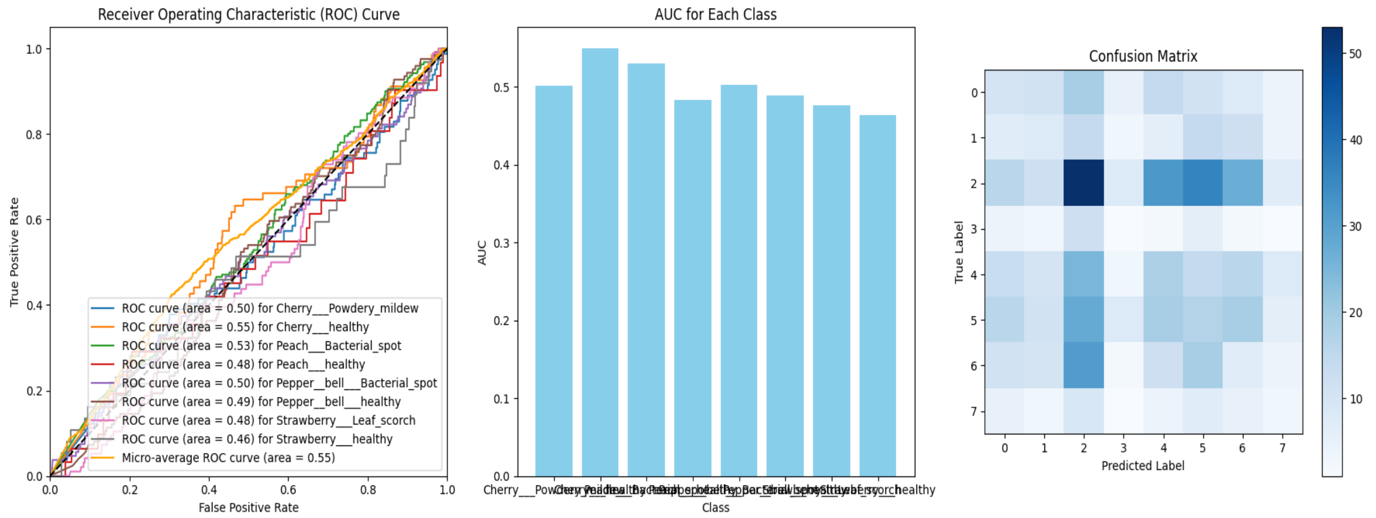




Continue…

Performance Metrics:

- Accuracy, loss, and other relevant metrics such as the Receiver Operating Characteristic (ROC) curve, Area Under the Curve (AUC) for each class, and confusion matrices are calculated to validate the model's effectiveness.
- A classification report provides detailed metrics including precision, recall, f1-score, and support for each class, offering insights into the model's strengths and areas for improvement.



Continue…

```
Classification Report:
                              precision    recall  f1-score   support

      Cherry___Powdery_mildew       0.12      0.12      0.12        82
             Cherry___healthy       0.12      0.12      0.12        68
       Peach___Bacterial_spot       0.28      0.28      0.28       191
              Peach___healthy       0.03      0.03      0.03        31
Pepper__bell___Bacterial_spot       0.17      0.17      0.17       107
       Pepper__bell___healthy       0.14      0.14      0.14       124
     Strawberry___Leaf_scorch       0.07      0.07      0.07        96
         Strawberry___healthy       0.08      0.08      0.08        37

                     accuracy                           0.16       736
                    macro avg       0.13      0.13      0.13       736
                 weighted avg       0.16      0.16      0.16       736
```

Git Repo:

https://github.com/hsallrounder/Plant-Disease-Classification

Simple EDA:

Actual: Potato__Early_blight,
Predicted: Potato__Early_blight.
Confidence: 99.99%

Actual: Potato__Early_blight,
Predicted: Potato__Early_blight.
Confidence: 100.0%

Actual: Potato__Late_blight,
Predicted: Potato__Late_blight.
Confidence: 99.75%

Actual: Potato__Early_blight,
Predicted: Potato__Early_blight.
Confidence: 99.98%

Actual: Potato__Early_blight,
Predicted: Potato__Early_blight.
Confidence: 100.0%

Actual: Potato__Early_blight,
Predicted: Potato__Early_blight.
Confidence: 100.0%

Actual: Potato__Late_blight,
Predicted: Potato__Late_blight.
Confidence: 94.43%

Actual: Potato__healthy,
Predicted: Potato__healthy.
Confidence: 99.94%

Actual: Potato__Early_blight,
Predicted: Potato__Early_blight.
Confidence: 99.92%

Conclusion

The SmartCrop AI project stands as a pivotal innovation in the field of agricultural technology, aimed at empowering small to medium-sized farms with the ability to accurately diagnose and manage plant diseases. The project's progression from concept to a validated prototype illustrates the impactful role that machine learning can play in modernizing agricultural practices.

The developed system features an accessible platform, courtesy of a Flask-based application, which simplifies the process for farmers to upload plant images and receive diagnostic insights. This ensures that the solution is practical for farmers of all technical backgrounds.

Initial test results are promising, reflecting the system's capability to provide reliable disease classifications. This paves the way for SmartCrop AI to evolve into an essential aid for farmers, potentially leading to better disease control, improved crop yields, and sustainable farming practices. With its potential for scalability and adaptability, SmartCrop AI is poised to make a meaningful contribution to agricultural productivity and sustainability.