

Leafinspect

- Case Study

Introduction

An Automated Leaf Disease Detection System was developed to assist farmers and agronomists in detecting and diagnosing plant diseases efficiently and accurately. Leveraging computer vision and machine learning techniques, LeafInspect provides real-time analysis of leaf images, allowing for early disease identification and timely interventions.

Problem Statement

Accurately and efficiently detecting healthy and unhealthy leaves using labeling for a classification problem presents a significant challenge in plant analysis. Current manual inspection and diagnosis methods are time-consuming, subjective, and prone to human error, leading to delayed responses and potential crop losses.

Why Leaf Inspect ?

- To identify and categorize anomalies in leaves.
- Detect a range of issues including diseases, pests, nutrient deficiencies, and physical damage.
- Identify and classify defects, enabling timely intervention to protect crop health.
- Aid farmers and researchers in improving agricultural practices by facilitating early detection and treatment of plant-related problems.
- Contribute to sustainable agriculture by optimizing crop health, minimizing yield losses, and reducing the need for manual inspection.

At a Glance

Challenges

- Dataset Variability
- Annotation & Labeling
- Class Imbalance
- Generalization to new
- Defects Real-Time Detection

Benefits

- Improved Accuracy
- Increased Efficiency
- Enhanced Crop Health and Yield

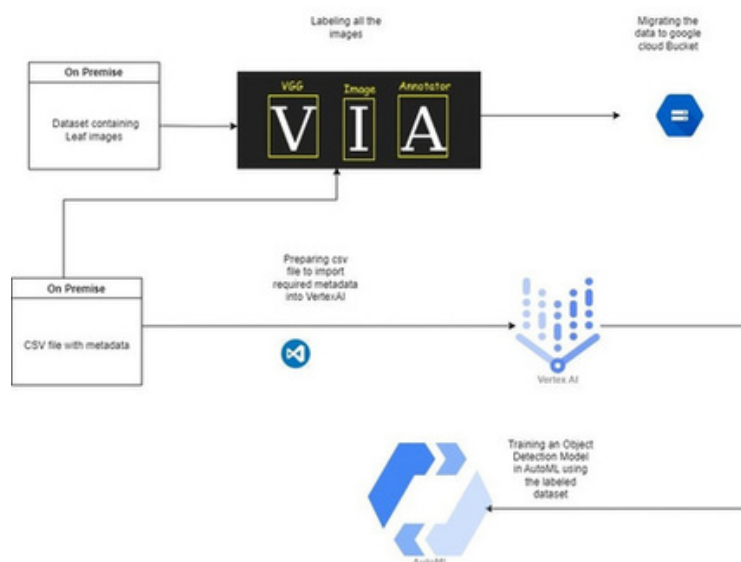
Leafinspect

- Case Study

Methodology

- **Data Collection:** The dataset comprised numerous annotated images and videos of diverse leaf samples, including both diseased and healthy leaves, ensuring a comprehensive training set for leaf defect detection.
- **Pre-Processing:** The collected dataset underwent preprocessing steps to standardize image resolutions, remove noise, and augment the dataset for improved model generalization. Techniques such as image resizing, noise reduction, and data augmentation were applied to enhance the quality and diversity of the training data.
- **Model Development:** The vehicle detection model was developed using deep learning techniques, specifically convolutional neural networks (CNNs). By training the CNN architecture on the annotated dataset, the model acquired the ability to recognize distinctive features and spatial relationships, ensuring precise vehicle detection.
- **Training and Validation:** The training of the model involved a combination of training and validation data, employing an iterative approach to fine-tune the model's parameters and enhance its performance.
- **Model Evaluation:** The leaf defect detection model undergoes a thorough evaluation using an independent test dataset, measuring crucial metrics including accuracy, precision, recall, and Intersection over Union (IoU).

Project Architecture



Leaf inspect - Case Study

Solution

The task is to develop an efficient and accurate system for leaf health detection. The system should be capable of analyzing images of plant leaves and determining whether they are healthy or unhealthy based on the presence of diseases, pests, nutrient deficiencies, or other abnormalities. The goal is to provide an automated solution that can assist farmers, researchers, and agronomists in identifying and addressing leaf health issues in plants, thereby promoting timely interventions to prevent crop damage and yield loss. The system should leverage computer vision techniques, machine learning algorithms, and a diverse dataset of leaf images to achieve robust and reliable detection and classification of leaf health conditions.

Our Approach

- **Leaf Photos on Premises:** Collect a dataset of leaf photos on your local system.
- **VGG Image Annotator (VIA) Tool:**
 - Use the VGG Image Annotator tool to load the leaf photos.
 - Manually label and annotate the images using the VIA tool to identify and classify the different leaf types.
 - Save the annotated images, along with their corresponding labels, in a designated format.
- **Google Cloud Platform (GCP) Bucket:** Create a GCP bucket to store the labeled leaf photos.
- **Exporting Labeled Data:**
 - Export the labeled data from the VIA tool in either CSV or JSON format.
 - The exported file should contain the metadata of each labeled image, including the image location in the GCP bucket and its associated labels.
- **Preparing Vertex AI-Compatible Metadata:**
 - Transform the exported file into a format compatible with Google Cloud's Vertex AI.
 - Modify the metadata file to adhere to the specific format required by Vertex AI for importing labeled data.
- **Vertex AI Object Detection Model Training:**
 - Utilize Vertex AI's AutoML capabilities to train an object detection model.
 - Use the labeled dataset, stored in the GCP bucket, as the training input for the AutoML model.
- **Testing and Accuracy Evaluation:**
 - After training the model, test its accuracy by using sample images.
 - Provide the sample images as input to the trained object detection model in order to evaluate its performance and accuracy.