Politecnico di Torino

Automatic Classification of healthy/ disease plant leaves using multispectral images of plants

Supervisor: Candidate:
Prof. Maurizio Morisio   Abdullah Hassan

Academic Year 2022/2023

Table of Contents

Chapter 1-Introduction

As more crops are needed to feed the growing population, it is difficult to check the health of plants in traditional ways like direct visual inspection of plants and taking samples to laboratory analysis could be time consuming and expensive. This time-consuming process can lead to delayed decision making where there are infectious diseases that can cause the loss of multiple plants.

In this thesis our focus is on the dronut project, which aims to create the hardware and the software infrastructure capable of providing the automated health of classification of individual plant using remote sensing services. Using the multispectral images of plants obtained by a Drones in various phases, different approaches will be used to develop the software that classify the health status of each plant of hazelnut.

Initially there is a brief discussion on what are multispectral images and how these images are taken and which type of information they provide. Although hyperspectral sensors despite not utilized in this project, a brief discussion on hyperspectral sensors characteristics will be presented as they may offer significant information in the future.

In Chapter 2, we describe the experimental context, focusing on the location and characteristics of the crop field where the study was conducted. The field is situated at specific coordinates (latitude, longitude) to allow accurate positioning. To provide a thorough visual depiction, panoramic images of the field are included, showcasing the landscape, topography, and overall environment. This detailed information is crucial for understanding the experimental setup and interpreting the outcomes of our study.

In Chapter 3, we introduce multispectral images, describing their characteristics and the various spectral bands they encompass. We highlight their significance and explore their

numerous applications in agriculture. This chapter presents an overview of how multispectral imaging enhances our understanding of crop health, growth stages, and stress levels, eventually resulting to improved crop management and increased agricultural productivity.

In Chapter 4, we demonstrate the image dataset, detailing drone-captured images from three distinct seasons of a year. We describe the labeling process and provide an overview of the dataset's composition, offering valuable insights for our analysis.

In Chapter 5, we outline the tools and languages used in our study, such as Python, which enable the classification of healthy and unhealthy plant leaves and facilitate effective analysis decision-making.

In Chapter 6, we discuss our objectives and describe the approaches taken, focusing on image pre-processing and the separation of healthy and unhealthy plant images based on their labels.

In Chapter 7, we explore the detailed architecture of Convolutional Neural Networks (CNN), providing an overview of their structure and play as a key role in our image analysis and classification methodology.

In Chapter 8, we discuss the training, validation, and results of the CNN model, highlighting the accuracy metrics achieved and the overall performance of our approach in addressing the research objectives.

In Chapter 9,10 we demonstrate the concept of transfer learning and the VGG16 pre-trained model, implemented to extract features from our dataset for our classification. In

In Chapter 11, we discuss the limitations encountered in our classification research, dealing challenges related to gathering the drone-based dataset and potential areas for improvement. In Chapter 12, we explore satellite images, providing detailed information and their resolution.

In Chapter 13, we demonstrate conclusions from our research and outline possible future improvements. We highlight the implications of our findings and discuss how

advancements in technology and methodology can further strengthen the study's impact and relevance in agriculture.

Chapter 2-Description of Experimental Context

In this chapter, we provide a detailed description of the experimental context, focusing on the location of the fields and their geographical coordinates. We begin by identifying the specific region where the crops are grown, elaborating on the environmental conditions and relevant agricultural practices in the area.

The dronut approach is experimented on the following hazelnut orchards chosen for their diversity.

1.Borgata Piancerreto Sottana, Farigliano Location:

https://goo.gl/maps/cAPnCMCCVdJ2HwpWA

Earth type: flat

Plants: medium-sized, relatively distinct from each other

Health status: well-maintained plants

2.Azienda Agricola Caru

Location: https://goo.gl/maps/uuejH4aYvpXsn2xh8

Here is the panorama view of this field.

Earth type: flat

Plants: younger, well distinct from each other

Physio pathological state: well-kept and generally healthy plants

3. AGRION, Cravanzana

Location: https://goo.gl/maps/uA6HhUdNg4Zgcwv6A

Earth type: hilly

Plants: larger, not very distinct from each other

Pathophysiological state: cured plants

These images of plants are taken in 2021 and 2022 in August and April in the phase of full vegetation.

Chapter 3- Multispectral Images

Multispectral images are special images that are used to capture the reflection or emission of light in different parts of electromagnetic spectrum beyond what is visible to human eye. These images are taken by different cameras and sensors that detects the light in different wavelength such as near infrared (NIR), red, green, and blue. These images are used in different applications like medical, remote sensing, agriculture, and environmental monitoring. In recent studies these images are being used to detect the stress of plant.

Multispectral images typically have a limited number of spectral bands (usually 3-10), covering specific, non-contiguous regions of the electromagnetic spectrum, such as visible, near-infrared (NIR), and shortwave infrared (SWIR). This provides a broad overview of the spectral information of the imaged objects or surfaces.

Here are some applications of multispectral in Agriculture.

Plant Health Monitoring: Multispectral images of plants help us to monitor the health of plants. Healthy plants generally reflect more light in the near-infrared (NIR) region and absorb more light in the red and blue regions of the spectrum. By evaluating these reflectance values, it is possible to detect early signs of stress or disease in plants, allowing for timely interventions.

Nutrient Management: Multispectral images can help us to detect nutrient deficiencies in plants by monitoring their reflectance patterns. For instance, plants with nitrogen deficiencies often display reduced reflectance in the NIR region and increased reflectance in the red region. By identifying these patterns, it is possible to optimize their fertilization strategies and improve crop yield.

Irrigation Management: Using multispectral images, it can be possible to detect plant water content by analyzing the reflectance properties of plants in the infrared regions of the spectrum. This information can help us to make better decisions about irrigation scheduling and water resource allocation.

Pest and Disease Detection: Multispectral images can also be used to detect the existence of pests and diseases in crops. Some disease and pest have certain impacts on the reflectance property of the plants, which can be examine by the help of multispectral images, early detection of pest and disease enable us to take appropriate actions and control measures and minimize crop losses.

Crop Yield Estimation: Multispectral images can help us to determine the growth stage by providing reflectance properties of plants. This information can help us to make better decisions about harvest timing and resource allocation.

3.2 Hyperspectral Images

Hyperspectral images provide much higher resolution which consist of hundred or thousand of contiguous spectral bands. This means that hyperspectral images can help us to identify the specific substance and small variation in the makeup of object because each material has a unique pattern of reflected light across different wavelengths.

Both multispectral and hyperspectral images belong to remote sensing images that are used to capture information across different wavelengths of electromagnetic spectrum.

Hyperspectral images, with their higher spectral resolution, are more suitable for applications requiring detailed spectral information, such as identifying specific materials, detecting subtle changes in vegetation health, and characterizing geological features. They are used in fields like precision agriculture, environmental monitoring, mineral exploration, and defense.

Here are some names of different sensors, along with their corresponding wavelength ranges and spectral bandwidths. Each sensor is designed to capture light within specific parts of the electromagnetic spectrum.

sensor name	Wavelength range	Spectral bandwidth
RGB	S500-700	NA
NIR	710-885	16
Red edge	560	26
Green	650	35
Blue	450	26

3.1

3.3-Aerial Imagery with the DJI Phantom 4

The Dji phantom 4 is a quadcopter drone that is advanced and popular that has ability to take photography and videography. In our case, a survey is carried out by the DJI P4 drone and is being used to capture the images of plants. This drone has the following characteristics.

DJI p4 for agriculture purposes is it ability to capture detailed aerial imagery of large areas in short amount of time. It has 12-megapixel camera, equipped with a 1/22.3-inch CMOS sensor, can capture high quality images and 4K video.

Weight: The P4's weight of 1.87 kg (4.12 lbs) provides an optimal balance between portability and stability, making it ideal for agricultural professionals who need a reliable and maneuverable drone for monitoring their crops. Its lightweight design enables easy transportation to and from field sites, while its sturdy construction ensures stable flight performance even in challenging weather conditions. This combination of features allows users to capture clear and steady aerial imagery of their fields, essential for accurate crop analysis and health assessment.

Flight Time: With a flight time of 27 minutes, the P4 can cover large areas of farmland on a single battery charge, increasing the efficiency of crop monitoring and reducing downtime. This extended flight time is made possible by the drone's high-capacity 5,350mAh battery, enabling users to gather more data and insights during each flight.

Max Flight Range: The P4's maximum flight range of 7 km (4.35 miles) is a significant advantage in agricultural applications, as it allows users to survey vast areas of farmland without losing connection to the drone. This extended range ensures that farmers can maintain full control over the drone and obtain high-resolution aerial imagery from a safe distance, essential for identifying and addressing potential issues in crop health or growth.

Max Speed: The DJI P4's top speed of 72 km/h (45 mph) in Sport mode is particularly useful in agricultural settings, as it enables users to cover large areas quickly and efficiently. This impressive speed allows for faster completion of crop surveys, which is crucial during time-sensitive tasks such as disease detection and treatment application. Additionally, the P4's speed makes it possible to keep up with fast-moving agricultural equipment, enabling users to capture crucial data and imagery on-the-go.

Chapter 4- Dataset

4.1 Introduction

The dataset of plant images was collected by a team of professional botanists. The images were taken during the months of May, June, and July 2022. The dataset contains images of plants from two different locations: Carru and Fariligiana. The images were captured using drones, which allowed for a high level of precision in capturing each plant's details.

The images in the dataset are more accurately labeled. Each image is divided into various sections, and each section is described as either healthy or sick.

This makes the dataset more reliable, as it provides detailed information about the status of each part of the plant. Moreover, the dataset has been further categorized based on the date the images were taken. Each folder in the dataset contains an XLS file that describes the section status of each plant captured on a particular date.

The Excel file contains labels for each section, where a value of 1 indicates that the section is healthy, and a value of 0 indicates that it is unhealthy. This makes it easier to analyze the changes in plant health over time.

The images in the dataset are in JPEG format and have a high resolution, making it easier to identify the plants' details. In our case survey is carried out by the DJI P4 drone is being used to capture the images of plants.

Here are some key features of DJI P4, it has weight of 1.83kg and has max speed 72km/h, flight time is 27 minutes and range are 7km.

The new dataset is a significant improvement over the previous one, as it provides a more detailed and accurate description of the plants. With the precise labeling of each plant's section, it becomes easier to identify the exact area of the plant that is sick, and hence, take appropriate measures to treat it. The dataset also enables us to track the health of the plant over time, which can help us better understand the causes and effects of plant diseases.

The images are captured in two formats, JPG and TIF. The JPG format is widely used and can be easily viewed on different devices. The TIF format, on the other hand, is lossless format that provide the high-quality images with rich detail. The use of both formats ensures that the images are of high quality and can be used for the different purposes.

The TIF images are in greyscale form, meaning that they only contain shades of grey. This format is suitable for images that do not require color such as scientific images and aerial photographs.

Each set includes 6 photographs, consisting of JPG format image with the three RGP channels, three separate TIF format images for the distinct R,G,B channels and one TIF format image for each of the red edge and NIR bands.

Fig4.1

Pianta31_1	0
Pianta31_2	0
Pianta31_3	0
Pianta31_4	0

Pianta42_1	0
Pianta42_2	0
Pianta42_3	0
Pianta42_4	1

Pianta1_1	1
Pianta1_2	1
Pianta1_3	0
Pianta1_4	0
Pianta1_5	0
Pianta1_6	0
Pianta1_7	1
Pianta1_8	1
Pianta1_9	1

Fig4.2

4.2 Correlation Matrix of Plant Position, ID, and Physiological Measurements

4.2.1

		E1			PSI	Clorofilla
27.05.21		SCAMBI GASSOSI				
mappa (fila.posiz)	ci (ppm)	E (mol m^{-2} s^{-1})	gs (mol m^{-2} s^{-1})	A (μmol m^{-2} s^{-1})	Ψ (MPa)	CCI (°SPAD)
2,28	306.75	1.53	0.13	12.85	-0.40	7.03
2,29	380.00	0.90	0.05	3.46	-1.00	5.65
3,10	307.33	1.83	0.13	9.26	-0.82	10.38
4,10	261.50	0.56	0.05	3.71	-1.00	10.25
4,35	221.50	1.06	0.07	9.25	-0.80	6.36
6,26	255.33	1.62	0.16	11.16	-0.40	5.70
7,16	282.75	1.71	0.12	11.11	-0.70	15.15
8,23	198.50	1.50	0.15	12.52	-0.40	5.00
8,24	206.25	1.49	0.10	8.02	-0.65	9.90
8,29	349.67	1.31	0.09	6.41	-0.50	5.60
9,16	329.00	0.96	0.14	6.96	-0.50	7.03
9,17	261.50	1.04	0.10	6.26	-0.40	4.53
9,28	304.67	0.82	0.05	2.16	-0.60	9.98
10,03	211.00	0.77	0.05	2.75	-1.10	11.56
10,11	241.50	0.92	0.11	6.19	-0.55	9.78
media	274.48	1.20	0.10	7.47	-0.65	8.26
se	12.24	0.09	0.01	0.79	0.05	0.67

4.2.2

	rilievo 22.07.21
mappa (fila.posiz)	Sintomo
7-12	lesioni fungine
	no sintomi
8-13	lesioni fungine
8-14	clorosi internervale (batteri?), no lesioni su picciolo
	no sintomi
	no sintomi
	no sintomi
11-8	lesioni fungine
	no sintomi
	no sintomi

	altre piante con sintomi in zona demarcata, senza
8-15	clorosi-necrosi su nervature di alcune foglie
11-11	fungo
12-10	leaf rolling senza altri sintomi su alcuni getti

4.2.3

STRADA STERRATA

posizione	1	2	3	4	5	6	7	8	9	10	11	12
1												
2												
3										P_0016		
4												
5												
6												
7												P_0037
8											P_0033	P_0038
9										P_0029	P_0034	P_0039
10				P_0003	P_0004				P_0025	P_0030	P_0035	P_0040
11								P_0021	P_0026	P_0015	P_0036	
12							P_0017	P_0022	P_0027	P_0032		
13							P_0018	P_0023	P_0028			
14							P_0019	P_0024				
15							P_0020					
16							P_0007		P_0012			
17									P_0013			
18												
19												
20												
21												
22												
23									P_0011			
24									P_0010			
25												
26							P_0006					
27												
28			P_0001					P_0008		P_0014		
29			P_0002						P_0009			
30												
31												

The Professional botanist has provided a labeled file that contains data related to both plant physiology and pathology. The information is present in the form of table includes the date on which the was collected, the location of the plant, and various physiological measurements, such as ci (carbon dioxide concentration), E (water vapor exchange), gs(stomatal conductance), A(net CO2 assimilation rate), water potential, and cci(Chlorophyll Content Index).

The final column of the table in Fig4.2.2 provides information on the presence or absence of the symptoms or diseases in the plants, as well as any other relevant notes.

The table 4.2.3 that shows how each plant in the field is uniquely identified through plant id and linked to its specific location in the field based on it row number and position within the row. However, not all rows and positions in the field are represented in the table, only those where the plants have identified and tracked. The table helps keep tack of each plant and its location over time, even as the field changes and grows.

4.2.4

Each set includes 6 photographs, consisting of JPG format image with the three RGP channels, three separate TIF format images for the distinct R,G,B channels and one TIF format image for each of the red edge and NIR bands.
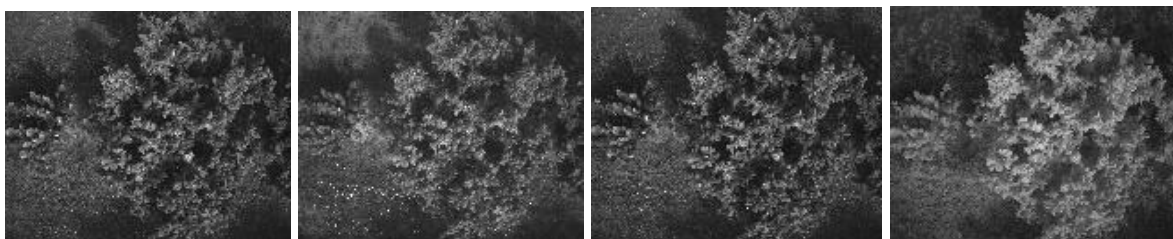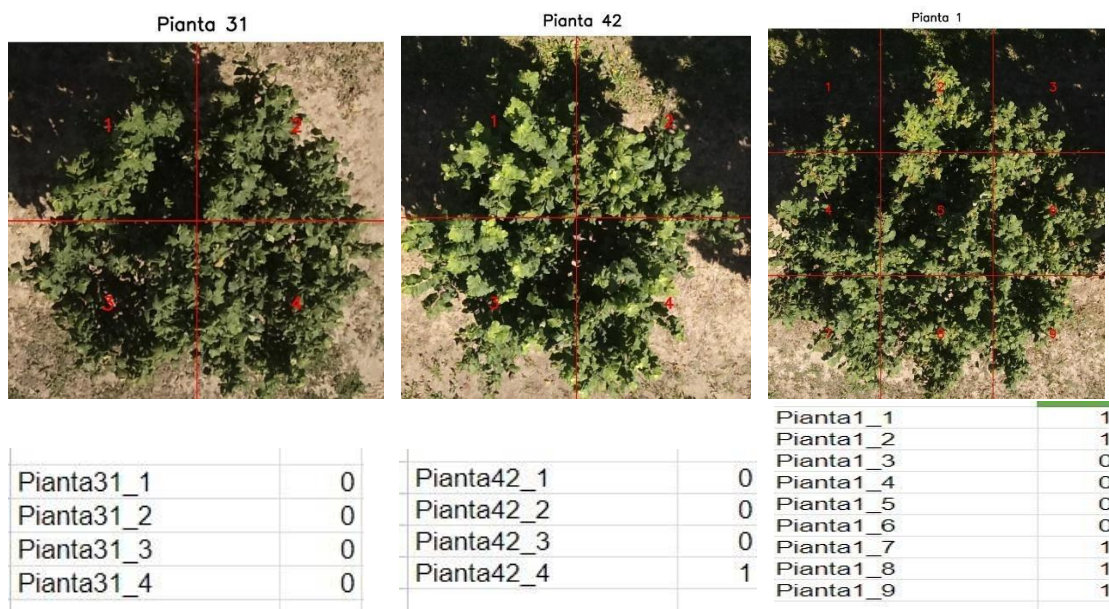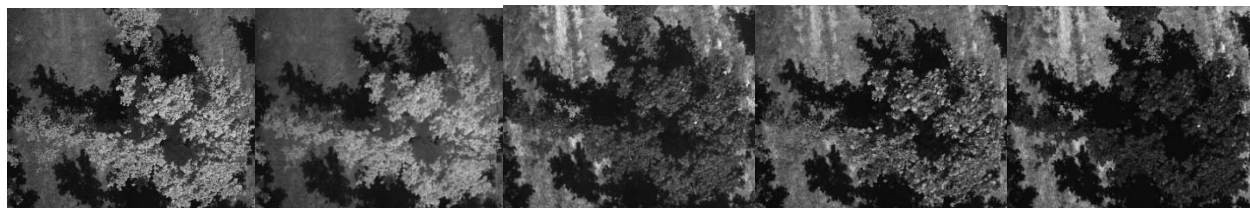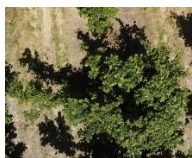
Chapter 5- Tools and Languages

In addition, the project involves the construction of machine learning models for classifying the health status of individual plants based on their photos. To enable this, we have utilized Python 3.7 programming/scripting language with relevant libraries, and the Google Colab suite.Python is widely used for the machine learning algorithms. Python has a large collection of libraries and frameworks that make it easier to work with data and build complex applications.

Google Colab is used as editor which is cloud-based platform which provide access to free GPU, which make it possible to perform complex machine learning tasks without need for the expensive hardware. It has a variety of pre-installed libraries and tools that make the data analysis and machine learning easy to understand.

Another advantage of using Google Colab is its frequent updates and maintenance, assuring that users always have access to the most recent versions of Python libraries and frameworks without the need to manually update their local environment. This feature saves time and ensures consistency across different projects.

Moreover, Colab's support for interactive visualizations using libraries like Matplotlib, Seaborn, and Plotly enables users to create engaging and informative plots that can aid in the understanding and interpretation of their data and results.

Overall, the combination of Python's extensive ecosystem and Google Colab's cloud-based platform with free GPU access creates a powerful and accessible environment for developing and deploying machine learning models, democratizing the field and enabling a wider range of individuals to participate in cutting-edge research and applications.

In summary, our project involves Jupyter Notebook, Python, and Google Colab to perform data analysis and machine learning tasks on plant images. We have imported several libraries to help you work with numerical data, structured data, files, directories, and

images. We have also used these libraries to separate the healthy and unhealthy images based on their labels.

Chapter 6- Objectives and Description of Approaches

As previously mentioned, our main goal is to perform image classification of healthy and unhealthy plants based on the labels provided by the botanists.

As part of our project, we have datasets available for different months, from May to July. Each dataset is in a separate folder with the name of the respective month.

Step 1- Data pre-processing for 2022.

The preprocessing steps we took on the images of plants to prepare them for classification will be described in this section. Our approach includes the division of the images into sections based on the labels provided in the accompanying Excel file. As depicted, Plant 31 has been segmented into four distinct sections, and we have divided the corresponding image into four exact sections accordingly. This approach was replicated for all other plants in the dataset.

Pianta31_1	0
Pianta31_2	0
Pianta31_3	0
Pianta31_4	0

Pianta42_1	0
Pianta42_2	0
Pianta42_3	0
Pianta42_4	1

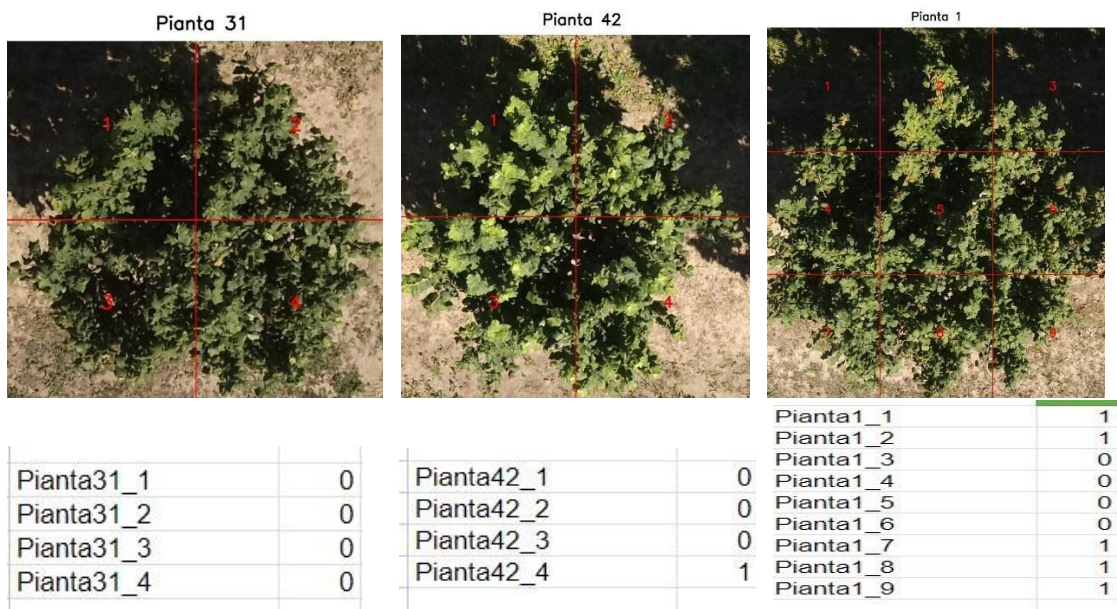| Pianta1_1 | 1 |
Pianta1_2	1
Pianta1_3	0
Pianta1_4	0
Pianta1_5	0
Pianta1_6	0
Pianta1_7	1
Pianta1_8	1
Pianta1_9	1

Fig.6.1

```
Plant No.101 successfully divided into 9 parts
Plant No.102 successfully divided into 9 parts
Plant No.103 successfully divided into 9 parts
Plant No.104 successfully divided into 9 parts
Plant No.105 successfully divided into 9 parts
Plant No.106 successfully divided into 9 parts
Plant No.107 successfully divided into 9 parts
Plant No.108 successfully divided into 5 parts
Plant No.109 successfully divided into 5 parts
Plant No.110 successfully divided into 5 parts
Plant No.111 successfully divided into 5 parts
Plant No.112 successfully divided into 6 parts
Plant No.113 successfully divided into 5 parts
Plant No.114 successfully divided into 2 parts
Plant No.115 successfully divided into 6 parts
Plant No.116 successfully divided into 7 parts
Plant No.117 successfully divided into 5 parts
Plant No.118 successfully divided into 3 parts
Plant No.119 successfully divided into 6 parts
Plant No.120 successfully divided into 3 parts
Plant No.121 successfully divided into 5 parts
Plant No.122 successfully divided into 7 parts
Plant No.123 successfully divided into 7 parts
Plant No.124 successfully divided into 7 parts
Plant No.125 successfully divided into 5 parts
Plant No.126 successfully divided into 4 parts
Plant No.127 successfully divided into 5 parts
```

Fig 6.2

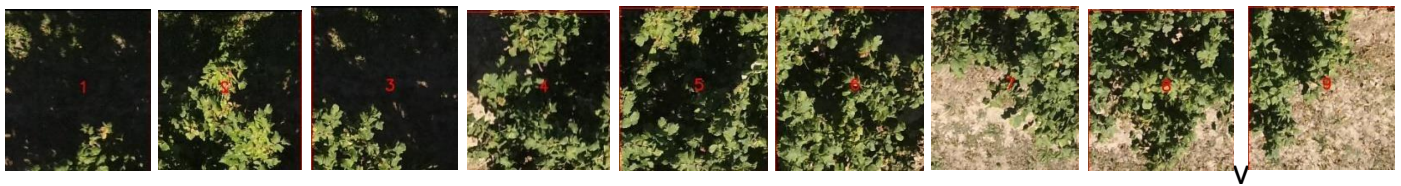Here are the examples of section images extracted from the images of plant 1 that were previously mentioned.

Fig 6.3

Once the images were divided, we then separated them into healthy and unhealthy categories based on the labeling provided in the Excel file. For instance, if the file indicate that the first four sections of a plant were healthy and the remaining two were unhealthy, we separated the corresponding image sections into healthy and unhealthy groups.

Steps that involve in the slicing.

First we load the label Excel file and drop any rows where the first column is NaN. Then, extract the relevant information, such as the plant IDs and the number of sections to crop the image into.

Next, load the image and remove any unwanted regions, such as a white color region or a top section that is not relevant to the analysis. Then, divide the remaining image into the desired number of sections based on the information from the label Excel file.

Finally, save each section as a separate image file in the appropriate folder with a naming convention that reflects the plant ID and the section number.

In addition to this, we also removed any objects and ground from the images to isolate the plant sections and reduce the impact of background noise. This helped us to obtain a cleaner dataset that was better suited for classification purposes.

Step-2-Ground and Object Removal for Plant Leaf Isolation in Images

In this section, we discuss about the method implemented for ground and object removal in the sections of image. For this approach we used the OpenCV library for image processing tasks and the NumPy library for array manipulation.

This involves several steps:

1. Color space conversion: We Convert the input image from the standard RGB color space to the HSV (Hue, Saturation, Value) color space. This conversion simplifies the identification of the green color in the image.

2. Defining the green color range: Then we Specify the lower and upper bounds for the green color in the HSV color space to create a binary mask. This mask will isolate the green regions in the image, corresponding to plant leaves.

3. Morphological operations: We applied the morphological operations like closing and opening to refine the mask and remove noise. These operations helped us to eliminate small objects, fill holes, and reduce any irregularities in the mask.

4. Mask application: We apply the binary mask to the original image, isolating the green-colored objects, which represent the plant leaves. This step removes the ground and other objects from the image, remain only the green leaves.

In the lower bound of the green color range, np.array([30, 50, 50]), the value 30 represents the lower limit of the hue component for the green color in the HSV color space. It specifies the starting point of the green hue range, which includes yellowish-green colors.

It is important to note that the hue component in the HSV color space ranges from 0 to 179, with different color types falling within specific hue ranges. The values 30 to 90 encompass the most common shades of green found in plant leaves, with 30 being the lower limit and 90 being the upper limit. This range helps in identifying and isolating green regions in an image, which corresponds to plant leaves in this context.
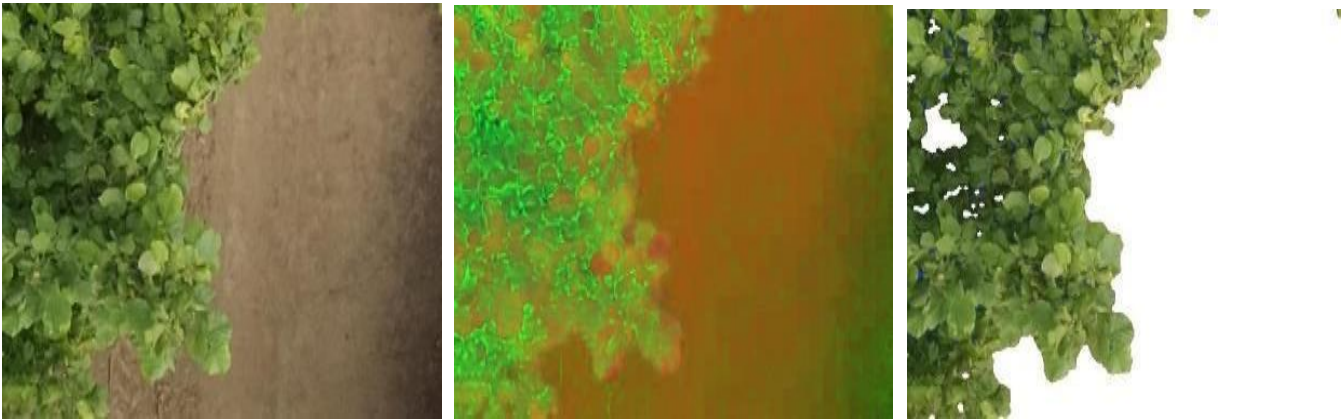
Some example images are as follow.

Fig 6.4

Fig 6.5

Chapter 7-Convolutional neural network (CNN)

"CNN is a subset of deep learning used for image and video recognition. Its architecture is inspired by the structure of the human brain and consists of a series of layers of neurons connected to each other in a specific way. These layers include convolutional, pooling, and fully connected layers that work together to learn the features and patterns of the input data. By using a convolution operation, CNNs can learn the spatial relationships and local patterns of the data, making them effective in image and video analysis tasks."

The input image is processed through a series of convolutional layers, pooling layers, and fully connected layers, each of which extracts and learns increasingly complex features from the image. These layers use filters to identify features such as edges, textures, and shapes. The final layer of the network produces the output, which can be a classification, prediction, or decision.

Convolutional Neural network (CNN) consist of three blocks:

1.Convolutional layer

2.Max Pooling

3.Fully connected layer

1.Convolutional Layers

These layers perform the convolution operation on the input data, which applies a filter or kernel to the input to extract features. The result is a set of feature maps that highlight the important parts of the input.

In convolutional layers, neurons are connected only to those within their receptive field, which is a narrow rectangular area. This connection pattern is repeated recursively for successive layers. A filter, which is a set of trainable weight values (convolution kernels), is applied to the input to highlight certain image features.

As the operation is repeated on different layers, the network learns increasingly complex filters that activate in the presence of specific types of features. Filters in the initial layers help the network identify simple lines, while deeper filters help recognize more complex features such as leaves, branches, and terrain.

Each layer containing neurons with the same filter produces a feature map, which highlights areas in the image that activate that specific filter the most. Overall, the convolutional layers play a crucial role in detecting and extracting features from input images for further processing by the neural network.

The convolutional layer applies a filter to the input image to extract features and create a new feature map. The size of the filter can vary, but common filter sizes include 3x3 and 5x5. Each filter is applied to the input image by performing a convolution operation, which involves multiplying the filter values with the pixel values of the image and summing the result. This process is repeated for each location in the input image to produce a new feature map.

2.Max pooling is a technique used in convolutional neural networks to reduce the size of the input by only keeping the maximum value within a defined window. It helps in reducing the computational load, and the number of parameters that need to be learned.

3.A fully connected layer is a layer in which all the neurons are connected to all the neurons in the preceding layer, as in a traditional neural network. This layer is usually added at the end of the CNN architecture and is responsible for making a final prediction based on the features extracted by the convolutional and pooling layers. It maps the high-level features learned by the convolutional layers to the final output classes. The fully connected layer is

often followed by a softmax activation function, which provides a probability distribution over the output classes.

7.2- Hyperparameters of our CNN Model

- filters: 32 is the parameter of 2D convolutional window.

- kernel_size: (3, 3) is the parameters of 2D convolutional window.

- input_shape: (500, 500, 1) is the input shape of images.

- activation: 'relu' is the activation function used in the layer.

- pool_size: (2, 2) is the size of the pooling window.

- dropout: 0.2: is the fraction of input units that are randomly dropped out at each update during training.

we used the Keras library to build a convolutional neural network (CNN) model. The model consisted of three convolutional layers, each followed by a ReLU activation function to increase non-linearity. The output from each convolutional layer was passed to a max-pooling layer with a grid size of 2x2 to reduce the spatial dimensions of the output. We also added a dropout layer with a rate of 0.2 to randomly disconnect some nodes between this layer and the next to avoid overfitting. Finally, the output of the last convolutional layer was flattened and passed to a fully connected layer, which produced the final output.

Chapter 8-Traing and Validation and Results of Model

The process of machine learning involves feeding data to a model so that it can learn and make predictions on new data. The data is split into two main sets: the training set and the validation set. The training set is used to train the model, while the validation set is used to evaluate the model's performance and prevent overfitting. In this case, we have used this process to train a model that can classify images of plants as healthy or unhealthy.

The model was trained on a dataset of 605 images and validated on a dataset of 152 images. We trained the model and the validation set together for 100 epochs. During the training process, the loss function and the accuracy were monitored to evaluate the performance of the model. The loss function measures the difference between the predicted output and the actual output, while the accuracy measures the percentage of correct predictions.
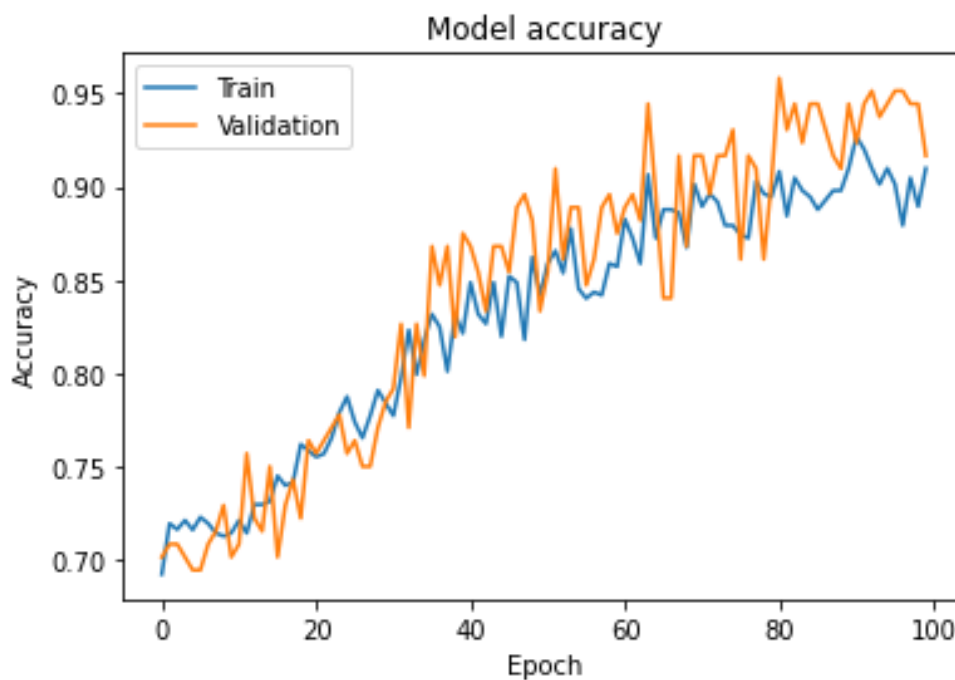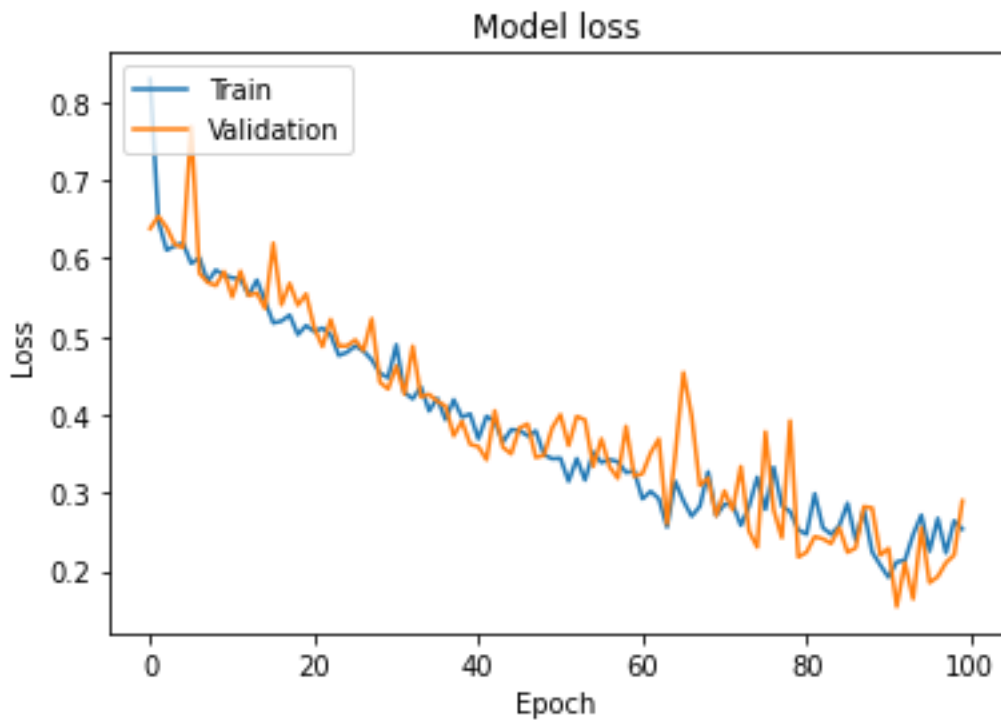

Fig 8.3

Fig 8.4

The ROC curve for both the training and testing datasets indicate that the model works well. The curve starts from zero and increases to one, which shows that the model can effectively distinguish between positive and negative cases.
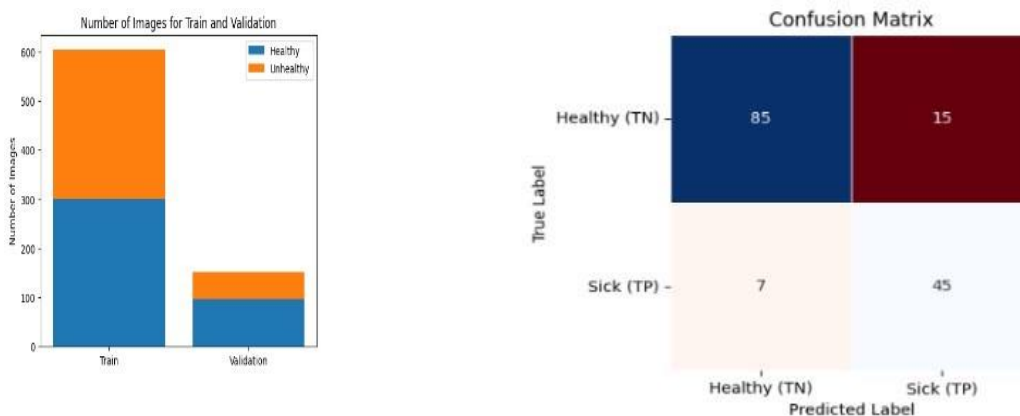

Fig 8.5

As we can see that our model has trained over 100 epochs for both training and validation as number of epochs are increasing the loss for both training and validation is decreasing, and accuracy is increasing throughout the training and validation. The final loss for validation is

reached 0.25 with accuracy of 85%. Relatively there is small difference between the training and validation datasets suggesting that our mode is performing well to unseen data. We evaluated its performance using confusion matrix for validation dataset.

True Positives (TP): For the instances of 45 where the mode correctly predicted sick leaves.

True Positives (TP): For the 85 instances where model correctly predicted healthy leaves.

False Negatives (FN): For): For the 7 instances where model incorrectly predicted healthy leaves. False Positives (FP): 15 instances where the model incorrectly predicted healthy leaves as sick.

Chapter 9-Approaches to Transfer Learning: Fine-Tuning and feature extraction

Transfer learning is a widely used technology in machine learning and deep learning in which pre-trained model is used to solve a new problem. It exploits the knowledge gained from a previous task to improve the performance on a target task. In transfer learning, pretrained model serves as a feature extractor and capturing meaningful representation from the new dataset. We implement Vgg16 on our dataset to extract the features from the data, the core idea of this approach to overcome the issue of insufficient data for training complex model from scratch.

9.1-Pre-trained VGG16 Model

The VGG16 model necessitates images with dimensions of 224x224 pixels.

To employ unsupervised algorithms, we utilize the pre-trained VGG16 model, a deep convolutional neural network trained on millions of images from various classes. VGG16 is widely used for tasks such as image and object classification. The model consists of 16 layers, including convolutional layers, fully connected layers, and pooling layers. It has already learned numerous features from the ImageNet dataset.

9.2-Feature Extraction and Fine-tuninng

The VGG16 model, which is already trained, to find important details from images in the training and validation sets. By removing the top layer and using global average pooling, the base model becomes better at identifying key features from the images, which can help create a custom model. The training set uses techniques like shear, zoom, and flip to make the model more adaptable. The ImageDataGenerator objects prepare the images based on VGG16's needs and create generators for training and validation sets. Finally, the base model finds features for the training and validation sets, which can be used in a custom model for more training, making the transfer learning process more effective.

9.3-Images Resizing

We utilized this script to resize images, employing the OpenCV library in Python. The primary element for resizing images is the cv2.resize function. To adjust image sizes, this function relies on a technique known as interpolation. In this specific script, the type of interpolation employed is called bilinear interpolation, which is represented by the cv2.INTER_LINEAR method. Bilinear interpolation works by determining the new pixel values in the resized image through a process that considers the nearest four pixels from the original image. It calculates a weighted average of these pixels to generate the new pixel values. As a result, the resized image maintains a smooth appearance while still being computationally efficient. To provide more context, here are some steps that elaborates the resizing process into simpler terms:

1. First, the script accesses an image file that is stored in a designated folder.

2. Next, it specifies the new dimensions (width and height) in our case we set the image size 224x224.

3. The cv2.resize() function is then used to change the image's size according to the new dimensions.

4. This function relies on bilinear interpolation (cv2.INTER_LINEAR) to adjust the image smoothly and efficiently.

5. Bilinear interpolation calculates the new pixel values by considering the four closest pixels in the original image.

6. It forms a weighted average of these four pixels to determine the new pixel value in the resized image.

7. This process is repeated for every pixel in the resized image to ensure a uniform and smooth appearance.

Chapter 10: Results of Binary Classification Algorithms

We also implemented several supervised learning algorithms, and the following sections present their results and evaluation metrics.

10.1-K-Nearest Neighbors (KNN)

K- nearest neighbors (K-NN) algorithm is supervised machine learning algorithm which is used for classification tasks, including binary and multi-classification. It relies on the label data to make prediction on new instances. It selects the k nearest neighbors and when a new instance is given, the algorithm classifies the new instances based on the majority of the data points in the training set.

In our study, we experimented with different k values in k-Nearest Neighbors (k-NN) algorithm, specifically k = 3, 5, and 7, to evaluate the model performance in predicting the labels for new instances. After analyzing the results, we found that k = 7 produced the best accuracy metrics compared to k = 3 and k = 5. Therefore, we concluded that k = 7 was the optimal choice for our binary classification task.

1.K-Nearest Neighbors (KNN): Here are the results for K-Nearest Neighbors, including the confusion matrix, ROC curve, F1 score, precision, and other evaluation metrics.
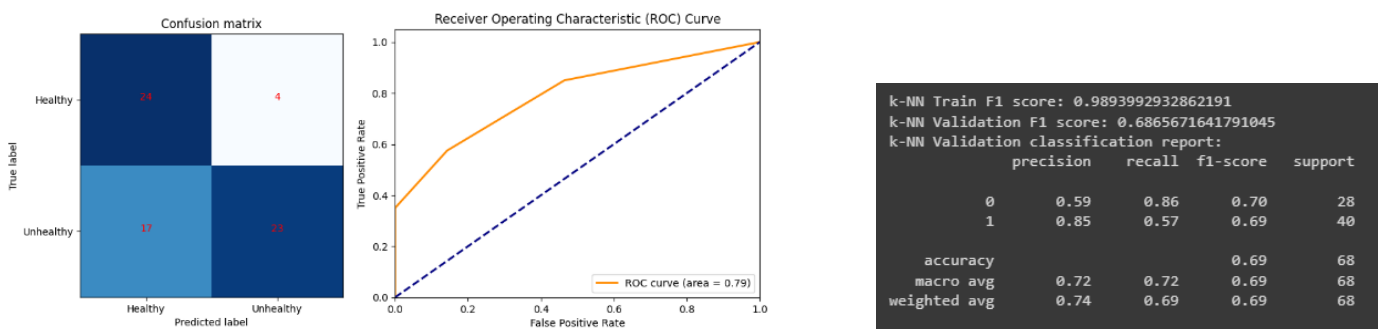

Fig 10.1

True Positives (TP): For the instances of 24 where the mode correctly predicted sick leaves. True Negative (TN): For the 23 instances where model correctly predicted healthy leaves. False Negatives (FN): For): For the 17 instances where model incorrectly predicted healthy leaves. False Positives (FP): 4 instances where the model incorrectly predicted healthy leaves as sick.

2-Support Vector Machine (SVM):

Support Vector machine (SVM) is a supervised algorithm and used for both classification and regression. It works in the manner to find the best hyperplane that separates two classes in the dataset, maximize the margin between two classes. The margin is the distance between hyperplane and the data points of each class. The datapoints that are close to decision boundaries are called support vectors. SVM can be used for both binary and multi classification. In multi-class classification algorithms use multiple boundaries to separate the data points. There are multiple parameters that must use when training the svm model. The kernel type, the regularization parameter C, and the kernel parameters, such as gamma parameter. In our research, we employed the Radial Basis Function (RBF) kernel in the Support Vector Machine (SVM) algorithm for binary classification. The default value of 'C' parameter, which controls the trade-off between maximizing the margin and minimizing the classification error, was set to 1.0.

Here are the results for the Support Vector Machine (SVM) classifier, including the confusion matrix, ROC curve, F1 score, precision, and other evaluation metrics.
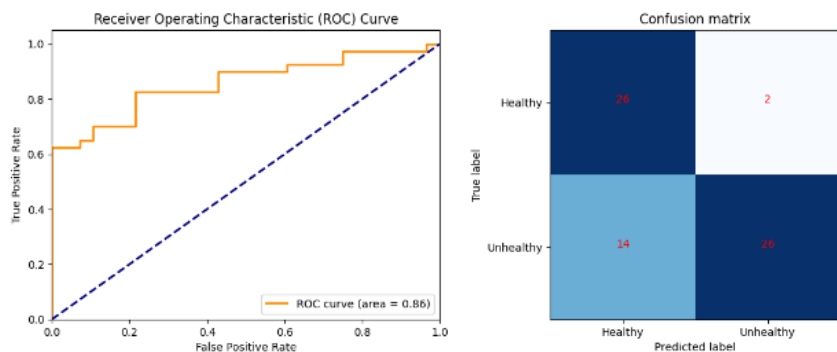
Fig 10.2

True Positives (TP): For the instances of 26 where the mode correctly predicted sick leaves.

True Negative (TN): For the 26 instances where model correctly predicted healthy leaves.

False Negatives (FN): For): For the 14 instances where model incorrectly predicted healthy leaves. False Positives (FP): 2 instances where the model incorrectly predicted healthy leaves as sick.

3-Random forest(RF):

Here are the results for the Random Forest classifier, including the confusion matrix, ROC curve, F1 score, precision, and other evaluation metrics
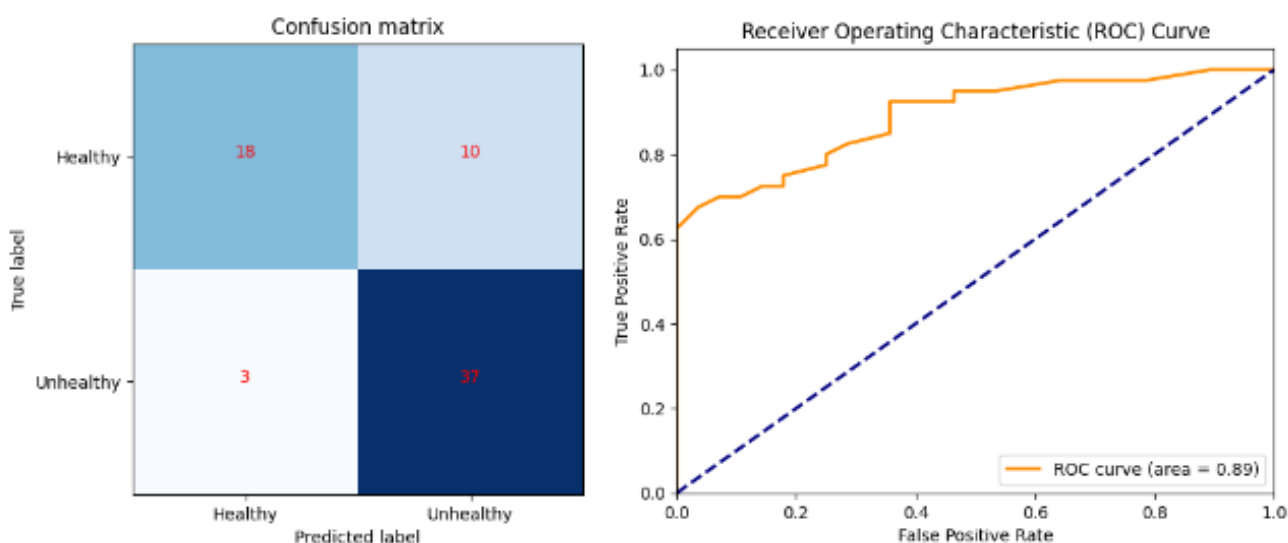

Fig. 10.3

```
Random Forest Train F1 score: 1.0
Random Forest Validation F1 score: 0.8095238095238095
Random Forest Validation classification report:
              precision    recall  f1-score   support

           0       0.75      0.64      0.69        28
           1       0.77      0.85      0.81        40

    accuracy                           0.76        68
   macro avg       0.76      0.75      0.75        68
weighted avg       0.76      0.76      0.76        68
```

Fig. 10.4

True Positives (TP): For the instances of 37 where the mode correctly predicted sick leaves. True Negative (TN): For the 18 instances where model correctly predicted healthy leaves. False Negatives (FN): For): For the 10 instances where model incorrectly predicted healthy leaves. False Positives (FP): 3 instances where the model incorrectly predicted healthy leaves as sick.

10.2-Comparative Analysis of KNN, SVM, and Random Forest Models

To determine which model performs better in predicting the health of leaves, we can compare the results of the K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Random Forest models based on their respective True Positives (TP), True Negatives (TN), False Negatives (FN), and False Positives (FP).

KNN showed 24 TP, 23 TN, 17 FN, and 4 FP, indicating that it correctly predicted 24 sick leaves and 23 healthy leaves. However, it incorrectly predicted 17 healthy leaves as sick and 4 sick leaves as healthy. SVM outperformed KNN with 26 TP, 26 TN, 14 FN, and 2 FP, correctly predicting 26 sick leaves and 26 healthy leaves while making fewer incorrect predictions compared to KNN. Lastly, the Random Forest model demonstrated superior performance with 37 TP, 18 TN, 10 FN, and 3 FP. It correctly predicted 37 sick leaves and 18 healthy leaves, with the lowest number of false negatives among the three models.

Based on these results, the Random Forest model exhibits the best overall performance in predicting the health of leaves, as it has the highest number of True Positives and the lowest number of False Negatives. Additionally, it has a relatively low number of False Positives, which further highlights its superior prediction accuracy. In conclusion, the Random Forest model appears to be the most suitable choice for accurately predicting leaf health among the three models evaluated.

Chapter 11- Limitations

One of the primary limitations encountered during this study was the limited dataset size. A larger and more diverse dataset would have allowed for more robust training and validation of the models, ultimately leading to better generalization and prediction accuracy.

Moreover, due to the slicing of images, the resolution of the individual images was reduced, which may have compromised the ability to discern finer details and subtle patterns indicative of plant health.

1.Drone Flight Restrictions

Another limitation is related to the drone flight regulations imposed in certain regions. These restrictions can hinder the application of drone-based monitoring services in some areas, reducing the overall coverage and utility of this approach. Furthermore, flying drones in restricted areas may lead to legal consequences, which further limits the applicability of drone-based monitoring.

2.Battery Autonomy and Pilot Requirements

Drones have a battery power limit of 20 to 30 minutes, which allows them to cover only 60 plants during this time, or approximately 120 plants per hour. Considering an average of 400 plants per hectare, it would take about 4 hours to cover one hectare. This battery autonomy constraint significantly affects the efficiency of the monitoring process and necessitates frequent battery replacements or multiple drones to cover larger areas. Additionally, a licensed pilot is required to fly the drone in the appropriate field, which not only reduces flexibility but also increases costs due to the need for skilled personnel.

3.Affordability for Small Farms

In Italy, hazelnut farms are typically small and generate lower income, which may render drone-based monitoring services unaffordable for many farmers. To address this issue, it is recommended to initially target larger farms that can more readily afford the service. As drone technology and image processing techniques continue to advance, it is expected that the costs associated with these services will decrease, making them more accessible to smaller farms.

Potential Solutions and Future Directions

To overcome some of the limitations mentioned above, several strategies can be employed:

1. Expanding the dataset: Collecting more data from various regions, under different weather conditions, and across diverse plant species can help improve the performance and generalizability of the models.

2. Higher-resolution imaging: Utilizing drones equipped with higher-resolution cameras can help capture more detailed images, potentially improving the accuracy of health assessments.

3. Multi-sensor systems: Integrating other sensors, such as hyperspectral, thermal, and LiDAR sensors, with the drone-based system can provide additional information on plant health and improve the overall monitoring process.

4. Improving drone technology: Advances in drone technology, including increased battery life and automated flight capabilities, can help address some of the limitations related to battery autonomy and pilot requirements.

5. Collaborative efforts: Partnering with local agricultural organizations and farmers can help facilitate the adoption of drone-based monitoring services, making them more affordable and accessible to smaller farms.

Chapter No 12- : Satellite Images

Satellite images are high-resolution visuals captured by Earth-orbiting satellites that provide detailed snapshots of the planet's surface from a bird's-eye view. These images are obtained through a variety of remote sensing technologies, such as multispectral and hyperspectral imaging, which allow scientists and researchers to analyze land use, monitor environmental changes, and track natural disasters, among other applications.

1.Observing Agricultural Health through Satellite Images

Satellite images play a significant role in monitoring the health of crops. These images can provide us with valuable information for our project, Dronut. Some benefits of satellite images are as follows.

1.Satellite imagery can help us to detect any signs of nutrient deficiency, pest attacks, and water deficiency. This allows the farmer to take early action before visible to naked eye.

2.Health of crops can be determined by using vegetation Indexes such as Normalize difference Vegetation Indexes (NDVI) which gives us the information about the photosynthesis activities and stress level of the plant.

3.Satellite imagery help us to determine the irrigation over or under within the field which enables us to optimize the irrigation system and save water.

In summary satellite images plays an important role in monitoring in modern agriculture by providing us essential data for the and managing crops, optimizing agricultural practices, and promoting sustainable and efficient farming.

2.Satellite Images Resolution

Satellite image resolution is typically measured using Ground Sampling Distance (GSD) or spatial resolution. GSD indicate the size of the smallest object or feature on the ground that can be clearly visible and differentiated in the image. Lower GSD value means higher resolution of image, similarly higher gsd value means low resolution of image.

GSD is usually expressed in terms of length, such as meters or feet. For instance there is an image that has value of GSD 10 meters it means each pixel has the size of 10x10 meters. Higher spatial resolution images have smaller GSD values, enabling the identification of finer details and smaller objects on the ground.

In summary, satellite image resolution is measured using GSD or spatial resolution, which indicates the level of detail that can be observed in the image by representing the size of the smallest distinguishable features on the ground.

3.Satellite Image Frequencies and their Resolution

Most satellites are equipped with imaging sensors that capture data across various bands of the electromagnetic spectrum. Here are the primary types of image frequencies offered by satellite constellations:

1. Panchromatic: Panchromatic image acquisition data across a wide range of visible light wavelengths, resulting in a single grayscale image. These images typically have a higher spatial resolution compared to multispectral images. These images have resolution up to 30cm GSD.

2. Multispectral: Multispectral images capture data in various separate spectral bands of the electromagnetic spectrum, typically including visible and near-infrared (NIR) wavelengths. These images provide information on the spectral properties of the Earth's surface, allowing multiple remote sensing applications, such as vegetation health analysis, land use classification, and water quality monitoring. The multispectral resolution is usually lower than the panchromatic resolution. for example, offers a multispectral resolution of 1.24 m GSD.

3. Shortwave Infrared (SWIR): WorldView-3 is equipped with a SWIR sensor that captures data in multiple bands within the shortwave infrared region of the spectrum. SWIR imagery can penetrate haze, smoke, and dust, providing clearer images in challenging atmospheric conditions. This capability is useful for applications

like mineral exploration mapping, fire monitoring, and land cover classification. SWIR resolution images are up to of 3.7 m GSD.

4. CAVIS (Clouds, Aerosols, Vapors, Ice, and Snow): Some satellites are equipped with a CAVIS sensor, which gather information on atmospheric conditions, such as clouds, aerosols, water vapor, and snow/ice. This data can be used to improve image quality.

Chapter 13 -Conclusion and Improvements in the future

To conclude, our model is performing well and shows better results in classifying plant leaves as healthy or sick. We implemented several unsupervised algorithms for an enhanced classification, with these methods the Random Forest algorithm demonstrated exceptional results, achieving an accuracy of up to 80 percent. We then also tried ResNet 50 model, but due to the small amount of data available in our case it did not perform well.

We encountered several limitations as well that affected our ability to achieve optimal results. To name a couple of these limitations are limited dataset size and low resolution after the slicing of images. These limitations presented with challenges in the implementation of certain models and algorithms. Despite these limitations our results demonstrate the potential of this approach for the plant health monitoring and highlight the need for future research in this area.

Some future improvements for our project could include, increasing the size of image dataset, having more accurate labeling methods etc. Having a bigger number of images can potentially help us to train more accurate models. Another worthwhile improvement is incorporating Thermal images which can play an important role for the monitoring of plant health by representing the distribution of temperature on the surface of leaves. Which indicates the stress and disease. Additionally, thermal images can be used to identify the spots on the leaves that may identify the presence of insect infection.

Overall, this project demonstrates the potential of multispectral images and machine learning models for plant health monitoring and lays a solid foundation for future endeavors in this area. With the aid of this technology, we are not far from providing farmers and researchers with a powerful tool for detecting and addressing plant stress, improving crop yields, and contributing to sustainable and environmental friendly agricultural practices.

Bibliography

[1] DJI Site:

https://www.dji.com/fr/phantom-4/info

[2] Introduction to Multispectral images:

https://towardsdatascience.com/what-are-hyper-spectral-images-a5de5d9fa91

[3] Data Augmentation:

https://www.datacamp.com/tutorial/complete-guide-data-augmentation

[4] Images and masks splitting into multiple pieces python with colab:

https://towardsdatascience.com/images-and-masks-splitting-into-multiple-pieces-in-python-with-google-colab-2f6b2ddcb322

[5] Transfer Learning with deep convolutional neural network:

https://journalofbigdata.springeropen.com/articles/10.1186/s40537-022-00652-w

[6] Convolutional Neural Network:

https://insightsimaging.springeropen.com/articles/10.1007/s13244-018-0639-9

[7] TensorFlow official website:

https://www.tensorflow.org/

[8] Accuracy matrices in machine learning:

https://medium.com/@kkaran0908/accuracy-matrices-in-machine-learning-ad77818e50c3

[8] Satellite Images:

https://www.maxar.com/products/satellite-imagery