

# Early Diagnosis and Severity Assessment of Weligama Coconut Leaf Wilt Disease and Coconut Caterpillar Infestation using Deep Learning-based Image Processing Techniques

SAMITHA VIDHANAARACHCHI<sup>1,\*</sup>, (Member, IEEE), JANAKA L. WIJEKOON<sup>2,3,\*</sup>, (Senior Member, IEEE), W. A. SHANAKA P. ABEYSIRIWARDHANA<sup>4</sup>, (Member, IEEE), AND MALITHA WIJESUNDARA<sup>1</sup>, (Member, IEEE)

<sup>1</sup>Sri Lanka Institute of Information Technology, New Kandy Rd, Malabe 10115, Sri Lanka

<sup>2</sup>Victorian Institute of Technology, Adelaide Campus, South Australia

<sup>3</sup>Department of System Design Engineering, Keio University, Yokohama, Japan

<sup>4</sup>ACSL Ltd 3-6-4 Rinkaicho Edogawa, Tokyo 134-0086 Japan

Corresponding author: Samitha Vidhanaarachchi (e-mail: samithapva@gmail.com).

**ABSTRACT** Global Coconut (*Cocos nucifera* (L.)) cultivation faces significant challenges, including yield loss, due to pest and disease outbreaks. In particular, Weligama Coconut Leaf Wilt Disease (WCWLD) and Coconut Caterpillar Infestation (CCI) damage coconut trees, causing severe coconut production loss in Sri Lanka and nearby coconut-producing countries. Currently, both WCWLD and CCI are detected through on-field human observations, a process that is not only time-consuming but also limits the early detection of infections. This paper presents a study conducted in Sri Lanka, demonstrating the effectiveness of employing transfer learning-based Convolutional Neural Network (CNN) and Mask Region-based-CNN (Mask R-CNN) to identify WCWLD and CCI at their early stages and to assess disease progression. Further, this paper presents the use of the You Only Look Once (YOLO) object detection model to count the number of caterpillars distributed on leaves with CCI. The introduced methods were tested and validated using datasets collected from Matara, Puttalam, and Makandura, Sri Lanka. The results show that the proposed methods identify WCWLD and CCI with an accuracy of 90% and 95%, respectively. In addition, the proposed WCWLD disease severity identification method classifies the severity with an accuracy of 97%. Furthermore, the accuracies of the object detection models for calculating the number of caterpillars in the leaflets were: YOLOv5-96.87%, YOLOv8-96.1%, and YOLO11-95.9%.

**INDEX TERMS** Coconut, Pest control, Transfer Learning, Deep Learning, Image Processing, Mask R-CNN, YOLOv5, YOLOv8, YOLO11

## I. INTRODUCTION

The *Cocos nucifera* (L.) (commonly known as coconut) tree is a member of the palm family, which has a 50-60-year economic lifespan and starts bearing fruits within 5-6 years after being planted. Usually, the tree grows up to 30 m with a crown of 30-40 leaves, making an ample amount of food and shelter for the survival and reproduction of many insects and pests [1]. Pests and diseases in coconut plants cause fatal damage to the palm and can result in outbreaks, causing heavy economic losses [1]. Therefore, efficient disease and

pest-controlling methods are vital for coconut cultivation [2].

Recent studies [3]–[8] discuss that Weligama Coconut Leaf Wilt Disease (WCLWD) and Coconut Caterpillar Infestation (CCI) pose the greatest threats to coconut plantations in Sri Lanka and regional countries, causing rapid and severe damage to the coconut cultivation. According to [4]–[6], WCLWD, which is an incurable disease, has been detected to be spreading at an alarming rate in Sri Lanka due to lack of early detection capabilities. During the expert survey, it was confirmed by principal entomologists from

the Coconut Research Institute of Sri Lanka (CRISL) [9] WCLWD symptoms, specifically uneven yellowing [3], [10], [11], are frequently misclassified as the yellowing caused by Magnesium (Mg) deficiency [10], [11], making it difficult for even trained individuals to differentiate the two [3]. Similarly, the dried appearance of leaves caused by CCI is also difficult to distinguish from Leaf Scorch Decline (LSD) [3], which is a physiological disorder. For these reasons, the majority of coconut growers are incapable of identifying the WCLWD and CCI infestations at their early stages.

The principal entomologists also confirmed that both WCLWD and CCI are detected using on-field human observations, which is not only time-consuming but also limits the early detection of infections. ICT tools have been successfully incorporated into developing pest and disease management practices in the recent past to early identification and treatments [12], [13]. Also, recent literature has shown that Machine Learning (ML) and artificial intelligence (AI) are thriving in agricultural research studies, significantly improving precision farming through automated analysis and decision-making [14], [15]. Among these advancements, the use of computer vision and deep learning (DL) techniques, such as feature extraction to automate pest and disease classification, is gaining momentum as a valuable tool to modernize conventional observation methods [1], [2], [16]–[20]. Consequently, as an extension of the preliminary study [13], this paper presents a study on the effectiveness of using DL techniques for the early identification of CCI, automating the detection of caterpillars, and the early identification of WCLWD.

## II. BACKGROUND SURVEY

This section explores the application of DL and Machine Learning (ML) in agriculture, focusing on disease and pest management. The section covers the use of DL models, CNNs, instance segmentation, and object detection techniques in identifying and classifying plant diseases, nutrient deficiencies, and pests. The review also highlights specific studies and their findings in these areas, emphasizing the potential and current limitations in the field. A comprehensive summary is presented in Table 1.

Several studies have explored the use of DL techniques for disease identification and management. For instance, Sladojevic et al. [21] and Han et al. [17] have highlighted the success of CNN-based models in classifying healthy and diseased leaves, as well as detecting nutrient deficiencies in black gram. The work of Francl et al. [19], further underscores the application of Artificial Neural Network (ANN) in predicting disease appearances and identifying specific plant diseases with high precision and recall rates. Additionally, studies by Miriyagalla et al. [32], Wijekoon et al. [22], Hahn et al. [20], and Huang [23] have contributed to the development of platforms for predicting disease spread and detecting specific plant diseases using advanced technologies like near-infrared spectroscopy and deep neural networks. Even though such studies justify that the use of DL technologies together

with image processing is very effective, none of those studies focused on the coconut industry, making it a niche sector to apply such advanced techniques for the early identification of diseases.

Instance segmentation and object detection technologies have also become prominent tools for pest and disease detection in plants. Wang et al. [24] detailed the utilization of Faster R-CNN and Mask R-CNN to identify and segment plant diseases accurately. Geetha et al. [25] advanced this field by addressing the challenges of detecting small pests in natural environments and classifying them into multiple categories. Their methodologies have significantly enhanced the accuracy and efficiency of in-field pest detection systems, which are crucial for sustainable crop management. Manoharan et al. [26] employed Mask R-CNN to calculate the extent of nutritional deficiencies (Nitrogen, Phosphorous, and Potassium) in crops like Guava, Groundnut, and Citrus by assessing the degree of leaf unhealthiness through masked images. Hewawitharana et al. [27] addressed the persistent threat of Blister Blight (BB) in the Ceylon tea industry and provided classification and severity assessment using CNNs and Mask R-CNN. Additionally, Tiwari et al. in [28] reported the detection of Fall Armyworm (FAW) insects in field crops using Mask R-CNN, leading to the creation of bounding boxes and segmentation frames for each insect in the image. Unfortunately, none of these studies were focused on coconuts.

The automation of pest management has also been a focus in recent literature, as evidenced by the work of Lins et al. [16], who developed software for automated counting and classifying aphids. This method addresses the inefficiencies and errors associated with manual counting. However, there remains a gap in the automation of counting coconut caterpillars, as they still resort to manual counts in current practices.

It was identified that there is limited literature specifically focused on coconut diseases and pest. Among them, Chandy's study [2] proposed a DL-based system identifying coconut pest infestations, but it did not extend to disease classification. However, several studies have employed DL techniques for classifying diseases in coconut trees. Maray et al. [29] introduced the AIE-CTDDC model for disease detection, but not for severity assessment. Kadethankar et al. [30] presented an end-to-end pipeline for detecting rhinoceros beetle infestations in coconut trees using drone imagery. Their approach focused on detecting and extracting individual tree crowns from drone images, enabling targeted analysis. Singh et al. [31] conducted disease classification using CNN, but they identified severity assessment and DL segmentation methods as future research direction. These are also two main objectives of the present study.

As summarized in Table 1, there is a clear need for further research to leverage advanced AI techniques for improved pest and disease management in coconut cultivation because as stated perviously, research on coconut pest and disease management is relatively limited, with most studies focusing on disease classification rather than severity assessment or

TABLE 1: Summary of significant ML and DL applications in agriculture

Ref	Crop	Objectives	Methods	Respective Accuracy's	Year
[21]	Pear, Cherry etc...	Disease classification	CNN-based models	96.3%	2016
[17]	Black Gram	Nutrient deficiency classification	Transfer learning-based CNN	65.44%	2019
[19]	Wheat	Disease propagation prediction	ANN, Logistic Models	93% & 90%	1997
[22]	Tomato, Beans etc...	Disease detection & dispersion	CNN, Mask R-CNN, Gaussian Plume	90% - 94%	2022
[20]	Tomato	Disease detection	Near-infrared spectroscopy	78%	2004
[23]	Moth Orchids	Disease detection	Gray level co-occurrence matrix	89.6%	2007
[24]	Tomato	Disease detection	Faster R-CNN, Mask R-CNN	88.53% & 99.64%	2021
[25]	Cabbage	Pest Detection	CNN, Mask R-CNN	95.3% & 98.3%	2024
[26]	Guava, Citrus etc...	Nutrition deficiency classification	CNN-based model	88%	2020
[27]	Tea	Disease severity assessment	CNN, YOLOv8, Mask R-CNN	89.9%, 98% & 89%	2023
[28]	Maize	Pest Detection	Mask RCNN	94.21%	2023
[16]	Wheat, Maize etc...	Pest classification & counting	OpenCV, ANN	97.9% & 92.5%	2020
[2]	Coconut	Pest and disease identification	DNN	Only Proposed	2019
[29]	Coconut	Disease Classification	Gated Recurrent Unit	97.75%	2022
[30]	Coconut	Infestation Classification	CNN, Faster R-CNN	84.64% & 97.30%	2021
[31]	Coconut	Disease Classification	CNN	96.94%	2021

advanced detection techniques.

Therefore, this study was conducted with three main objectives:

- 1) Analyze the effectiveness of using DL techniques to identify and classify WCLWD and CCI at their early stages
- 2) Calculate the degree of diseased conditions of WCLWD using transfer learning-based CNN
- 3) Measure the progression of CCI using advanced deep learning segmentation techniques such as Mask R-CNN and You Only Look Once (YOLO) object detection models, helping farmers determine the necessary precautionary management actions.

The experiments were conducted in hot spot areas in the Matara and Puttalam districts of Sri Lanka, and the datasets were collected from different locations and times to reduce class imbalances; the dataset is published in [33].

### III. MATERIALS AND METHODS

#### A. DATA COLLECTION

To develop the ML models for early detection and classification of CCI and WCLWD, we first collected images across a range of conditions, including the following:

- Healthy and disease-infected trees of different diseases, including flattened leaflets, uneven yellowing, tip browning, CCI, and WCLWD.
- Tree fronds of different sizes and growth.
- Different locations in the country.
- Different stages of the growth of both diseases.
- Different times of the day (morning and evening).
- Diverse weather conditions.
- Different image qualities and resolutions.

It should be noted that the two conditions below were applied to collect data from disease-infected leaves, as we were informed by field experts from the CRISL.

- For WCLWD-infected leaves, images were collected from the upper leaf surface.
- For CCI-infected leaves, leaflet images were acquired from the lower surface.

Images were captured using mobile phones and Digital Single Lens Reflect (DSLR) cameras. To cover a wider range of sensor and image qualities, iPhone6 with 8MP camera [34], iPhone11 with 12MP camera [35], and a high-quality Canon ESO 3000D DSLR with 18MP camera with a larger APS-C sensor [36] were used for data collection. Note that auto settings with auto-post processed JPEG output formatting were used in both scenarios. These conditions were aimed at aligning the data collection process with the traditional, manual assessments of diseases that have been practised within the CRISL [4].

The acquisition of leaf samples was carried out in collaboration with research personnel affiliated with the CRISL [9], within the time frame of April to May 2021. The sample images were classified and verified by skilled personnel from the CRISL, and this was considered the ground truth for this study. This approach was undertaken to mitigate potential similarities between image samples, thus ensuring a no class imbalance in the dataset.

A summary of the acquired dataset is given in Table 2. The WCLWD dataset employed for disease classification comprised a total of 9,258 images, while 3,307 images were used to evaluate symptom severity. The CCI infestation classification task was performed using a dataset consisting of 1,600 images, while the ML model for quantifying the number of caterpillars on a leaf was trained using 1,400 images. As one of the contributions of this study, we made the curated dataset available for researchers in this domain via Kaggle [33].

The coconut fronds infected by WCLWD were sourced from the city of Matara, Sri Lanka, an area known for disease prevalence. Matara is located within the low country wet agro-ecological zone of the southern province of Sri Lanka (latitude 6°-10° and longitude 79°-82°). According to research [37], WCLWD is isolated to these areas, appearing in pockets or clusters. Extensive containment measures, such as the removal and burning of infected palms, have been implemented. Hence, this area was selected for data collection, with the few remaining affected areas kept under strict observation and made accessible only for research purposes,

TABLE 2: Summary of Data Samples

Condition	Location	Purpose	Training	Validation	Testing (collected separately)
WCLWD	Weligama, Matara	Disease Classification	7406	1852	117
		Symptom Severity	2645	662	106
CCI	Puttalam, Lunuwila	Infestation Classification	1280	320	100
		Caterpillar Calculation	1120	280	140

allowing data to be gathered under carefully controlled conditions. Fronds exhibiting symptoms such as flattened leaflets, uneven yellowing, and tip browning were selected for image collection, as they are the initial symptoms of WCLWD [3].

For CCI, experimental images were obtained from multiple estates in Lunuwila and Makandura, Sri Lanka. Both these estates are situated within the low-country dry zone of the Puttalam district, a part of the area known for significant coconut cultivation, commonly referred to as the Coconut Triangle [38]<sup>1</sup>. Caterpillar damage is prevalent within the Coconut Triangle, thus this area was considered for data collection. We were careful to select leaflets displaying dried brown patches with caterpillar galleries for the dataset.

### B. DATA PREPROCESSING

The collected raw data underwent a series of preprocessing steps to enhance accuracy and to reduce calculation complexity. Image sizes ranging from  $150 \times 150$  to  $750 \times 750$  dimensions were collected during the field visit. To provide consistent input dimensions for neural networks, all images were resized to a uniform size of  $300 \times 300$  pixels. Further, the pixel values of the images were normalized by dividing them by 255, scaling them to a range between 0 and 1. Subsequently, data augmentation techniques of rotation, filling, shearing (both horizontally and vertically), flipping (both horizontally and vertically), and zooming were applied to expand the sample size and mitigate the potential of model overfitting. The deep learning models were trained using pre-processed coconut leaflet images, with the dataset split into an 80% training set and a 20% validation set. The test data were then collected separately to avoid manual pre-selection and biasness during field visits.

### C. DATA ANNOTATION

Initially, all preprocessed and augmented images were manually annotated by skilled professionals at CRISL. VGG Image Annotator (version 2.0.11) [39] was utilized by the CRISL experts to leverage their specialized domain knowledge for accurately identifying and labelling disease symptoms. This tool was selected owing to the effectiveness and the accuracy presented in agricultural research [40], [41]. The polygon annotation labelling technique was then used to mark the area of the leaflets. During the process, leaf images with and without infestation damage were labelled, and the

remaining region was set to the background, and the output file contains the coordinates for the polygonal regions. To train the model, these defined regions were fed into the Mask R-CNN model as input neurons.

It is important to note that the images used to train the YOLOv5 [42], YOLOv8 [43] and YOLO11 [44] models were annotated using MakeSense.ai [45], an open-source tool widely adopted in several similar studies, such as [46] and [47], for creating precise annotations in the field of plant disease detection. The labeling technique employed was box annotation. MakeSense.ai also generates a single JSON file containing the coordinates of the annotated box regions and associated metadata. These defined regions were subsequently used as input data for the YOLO models during training.

### D. CLASSIFY WCLWD AND ASSESS SEVERITY LEVEL

WCWLD can be distinguished using four key symptoms as listed below:

- 1) Flaccidity<sup>2</sup>
- 2) Uneven yellowing
- 3) Leaf tip browning
- 4) Breaking off the leaf tips

The fourth symptom, breaking off the leaf tips, is distinguishable, and at this stage, there is no value in the coconut tree. Therefore, as one of the main scientific contributions of this study is to detect the disease at an early stage, the first three symptoms, which are common in the early phases, were chosen as the most important [3]. The symptoms of WCWLD are shown in Fig. 1.

In this study, WCWLD was initially classified, and then disease severity was identified based on the progression of symptoms. The disease classification was performed using the flaccidity of the leaves, as it is the first symptom to appear, as shown in Fig. 1. This binary classification distinguished between healthy leaves and those exhibiting flaccidity. After classifying the disease based on flaccidity, disease severity was assessed using the remaining two symptoms: uneven yellowing and tip browning, which appear a few weeks later as the disease advances. To assess severity, a multi-class classification model with three classes—flaccidity, uneven yellowing, and tip browning—was trained. This strategy led us to identify and classify the disease at its early stages.

We trained ResNet50, DenseNet121, ResNet50V2, InceptionResNetV2, InceptionV3, MobileNetV2, Xception, and

<sup>1</sup>This geographical triangle includes Puttalam, Kurunegala, and Gampaha districts, which have excellent climatic conditions for coconut palms.

<sup>2</sup>Flaccidity is the initial symptom that aids in detecting infected palms during the early stages.





FIGURE 1: Symptoms of WCLWD. Images include (a) Flaccidity, (b) Uneven yellowing, (c) Drying of the leaflets, and (d) Breaking off the leaf tips.

VGG16 pre-trained CNN models in this study using the sample data provided in Table 2. These architectures were compared to determine the best fit for classifying the disease and its severity. The models were trained to predict the output of the CNN with binary classification to identify flaccidity and classify the disease.

Similarly, a multi-class classification model featuring three classes was trained to evaluate the severity of symptoms. CNNs are highly effective in image classification due to their ability to learn hierarchical features from images through a series of convolutional and pooling operations. The convolution operation involves applying a filter to the input image to extract important features like edges and textures. This operation can be expressed using (1).

$$(I * h)(x, y) = \sum_{i=-k}^k \sum_{j=-k}^k I(x+i, y+j) \cdot h(i, j) \quad (1)$$

where  $I$  represents the input image matrix, with  $I(x, y)$  denoting the pixel value at coordinates  $(x, y)$ , and  $h$  is the filter or kernel matrix applied to detect features within the image. The indices  $i$  and  $j$  represent the coordinates within the filter, ranging from  $-k$  to  $k$ , where  $k$  defines the filter size. The notation  $\sum_{i=-k}^k \sum_{j=-k}^k$  signifies the summation over all values in the filter region, allowing the weighted sum of pixel values centered at each  $(x, y)$  location.

After the convolution operation, the Rectified Linear Unit (ReLU) activation function is applied. The ReLU function is

defined using (2)

$$f(x) = \max(0, x) \quad (2)$$

where  $f(x)$  represents the ReLU activation function, which outputs the maximum of 0 and function's input variable  $x$ . This ensures that all negative values in the feature maps are set to zero, retaining only the positive activations, which helps the network capture complex patterns relevant to disease detection. To further reduce the dimensionality of the feature maps, we applied max pooling (3), which selects the maximum value from a defined window.

$$P = \max\{x_{i,j}\} \quad \text{for } i, j \in \text{window size} \quad (3)$$

where  $P$  denotes the output of the max pooling operation, which selects the maximum value  $x_{i,j}$  within a specified window size. The indices  $i$  and  $j$  represent the coordinates within this window. Max pooling reduces the spatial dimensions of the input while retaining the most prominent features, thereby enhancing computational efficiency and controlling overfitting in deep learning models. In the final stages of the network, a fully connected layer combines the features extracted from the convolutional layers to form a classification decision. The fully connected layer (dense layer) computes the output using (4)

$$z = W^T x + b \quad (4)$$

where  $z$  represents the output of a dense layer. In this equation,  $W$  is the weight matrix,  $x$  is the input vector, and  $b$  is the bias vector.

To optimize the performance of the CNN models, the Adam and Stochastic Gradient Descent (SGD) optimization algorithms were applied through hyperparameter tuning. This allowed the network to iteratively minimize the loss function, improving the accuracy of the disease classification. The symptoms in each image were analyzed to determine the stage of the disease. If all three symptoms—flaccidity, uneven yellowing, and tip browning—were present, it indicated that the disease had progressed to a severe stage. However, if only flaccidity was observed, it suggested that the disease was still in its early stages. This approach helped assess the severity and progression of the disease based on the visible symptoms in the images.

### E. IDENTIFY AND CLASSIFY CCI

Figure 2 illustrates the proposed Mask R-CNN method, to detect CCI-infected areas and filter out those regions to classify the infested leaf. This process aids in segmenting the leaflet and in calculating disease progression. As depicted in the figure, first, a Region Proposal Network (RPN) was used to scan the feature maps to propose candidate bounding boxes that represent region of interest (RoIs) for further analysis. The RPN minimizes the objective function as expressed in [48].

Based on the studies conducted in [18] and [26], Resnet101 was used as the backbone CNN network for CCI classification due to its effectiveness compared to other available models. Resnet101 was combined with a Feature Pyramid Network (FPN) for feature extraction. As illustrated in Fig. 2, each stage of the ResNet101 architecture correlates to a different scale of feature maps. The feature pyramid of the FPN network is constructed using these feature maps.

The feature maps returned from the ResNet101 backbone were sent to the RPN to determine regions of interest (RoIs) where the leaflets exist. Intersection over Union (IoU) values [49] were used to mark the RoIs. As calculated using (5), IoU is the ratio between the intersection and union of the ground truth bounding box and the model's bounding box. The ground truth bounding box and predicted bounding box are represented by  $A$  and  $B$ , respectively.  $|A \cap B|$  is the area of overlap or the intersection, while  $|A \cup B|$  is the area of union. If the IoU of each predicted region is greater than or equal to 0.5, RoIs are considered. The rest are defined as the background (refer to Fig. 2).

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|} \quad (5)$$

A bounding box rectangles were used to represent each RoI region acquired on the leaflet image, and Non-maximum suppression (NMS) [50] was used to choose the bounding box with the highest foreground value when several bounding boxes overlap (when several bounding boxes represent one detection). Above process was followed by adjusting the

dimensions of the anchor boxes with RoIAlign to provide a standard-sized output. Finally, the RoI features are used in the Fully Connected (FC) layer to conduct target classification of healthy and infected leaflets and bounding-box regression to mark the region.

### F. CALCULATE CCI PROGRESSION LEVEL

The extent of infestation throughout the leaflet is referred to as the progression level. A pixel-level instance segmentation conducted using a Fully Convolutional Network (FCN) was utilized to mask the infested leaflets (refer to Fig. 2), in addition to the current classification and bounding box regression methods used to detect CCI. Instance segmentation was specifically considered here to differentiate each instance of CCI identification. This technique identifies each object's instance by using a mask representation in the image, which simultaneously performs object class prediction and mask extraction providing distinct labels for separate occurrences of objects within the same class.

Consequently, the leaf area of the image was masked, and only the class label defined as CCI was processed to calculate the disease progression level. Then, image crop segmentation was applied using the mask regions to extract the infected leaflet from the background using the algorithm 1, based on the method presented in the research paper [51]. The result is an image where only the infected leaflets are visible, while the rest of the background is rendered black.

After crop segmentation, color segmentation was utilized to replace a range of HSV colors corresponding to green (healthy) and brown (necrotic leaf regions due to feeding of caterpillars) with single RGB pixel values ((20,255,10) & (19,69,139) respectively). This approach addresses the issue of color variations in the image, where multiple shades of green and brown are present. The remaining (background)

---

#### Algorithm 1 Apply Crop Segmentation

---

```

1: Input: RGB_Leaf_image, Masks
2: Output: Masked_images
3: Initialize
4:  $i \leftarrow 0$  {Mask index}
5:  $n\_masks \leftarrow$  number of Masks
6:  $Masked\_images \leftarrow$  empty array
7:  $empty\_matrix \leftarrow$  size of RGB_Leaf_image
8: while  $i < n\_masks$  do
9:    $Masked\_images.add(empty\_matrix)$ 
10:   $j \leftarrow 0$  {RGB channel index}
11:  for  $j < 3$  do
12:     $j = j + 1$ 
13:     $channel \leftarrow RGB\_Leaf\_image[:, :, j]$ 
14:     $Masked\_images[i][:, :, j] \leftarrow channel * mask[i]$ 
15:  end for
16:   $i = i + 1$ 
17: end while
18: return Masked_images

```

---

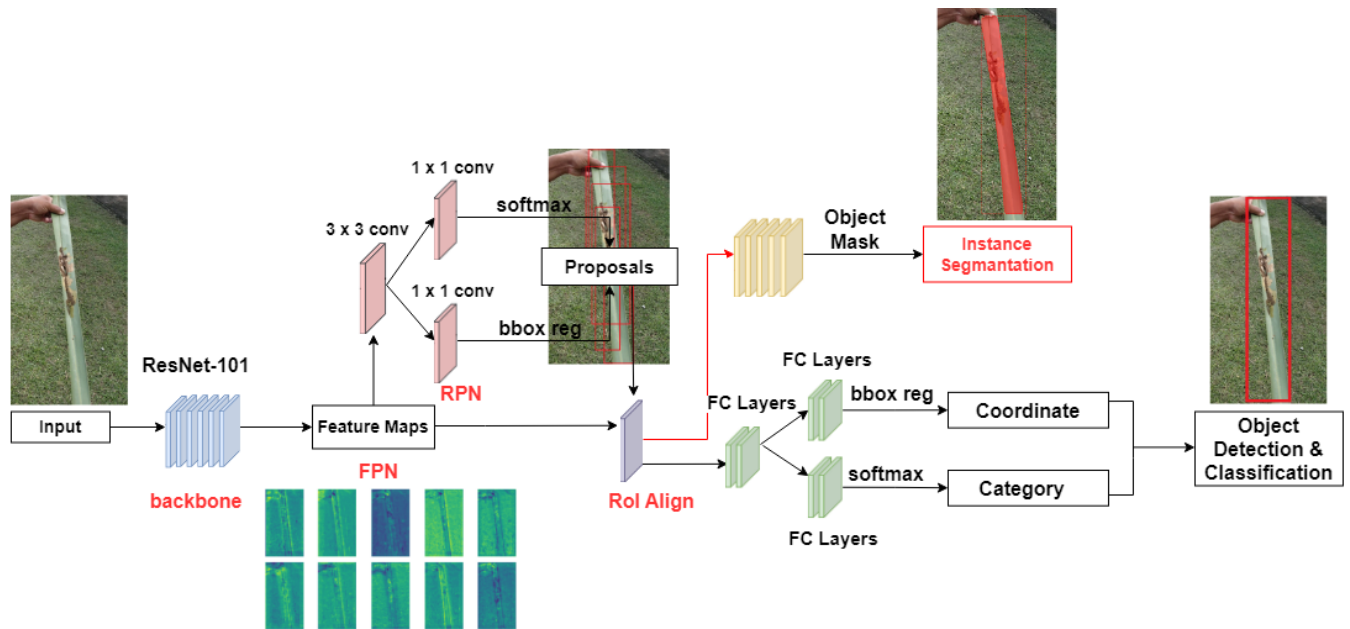


FIGURE 2: The training procedure of Mask R-CNN model for CCI instance segmentation

#### Algorithm 2 Color Segmentation, K-means Clustering, and Progression Calculation

```

1: Input: (img) Crop Segmented Image
2: Output: Progression %
3:  $k \leftarrow$  no_of_categories (Number of clusters/categories)
4:  $hsv\_img \leftarrow$  TO_HSV(img)
5:  $low\_g, up\_g \leftarrow$  Green HSV thresholds
6:  $low\_b, up\_b \leftarrow$  Brown HSV thresholds
7:  $g\_pixels \leftarrow$  IN_RANGE( $hsv\_img, low\_g, up\_g$ )
8:  $b\_pixels \leftarrow$  IN_RANGE( $hsv\_img, low\_b, up\_b$ )
9:  $img[g\_pixels] \leftarrow (20, 255, 10)$ 
10:  $img[b\_pixels] \leftarrow (19, 69, 139)$ 
11:  $img[!(g\_pixels \text{ OR } b\_pixels)] \leftarrow (255, 255, 255)$ 
12:  $kmeans \leftarrow$  INIT_KMEANS( $k$ )
13:  $lbls \leftarrow$   $kmeans.labels\_$ 
14:  $counts \leftarrow$  INIT_COUNT_ARRAY( $k$ )
15: for each  $lbl$  in  $lbls$ :
16:    $counts[lbl] \leftarrow counts[lbl] + 1$ 
17:    $g\_color \leftarrow [20, 255, 10]$ 
18:    $b\_color \leftarrow [19, 69, 139]$ 
19:    $g\_count, b\_count \leftarrow 0$ 
20: for each  $idx, count$  in  $counts$ :
21:   if  $centers[idx] = g\_color$  then  $g\_count \leftarrow count$ 
22:   else if  $centers[idx] = b\_color$  then  $b\_count \leftarrow count$ 
23:   end if
24: end for
25:  $total\_count \leftarrow g\_count + b\_count$ 
26:  $green\_perc \leftarrow$  ROUND( $g\_count / total\_count * 100$ )
27:  $brown\_perc \leftarrow$  ROUND( $b\_count / total\_count * 100$ )
28: return ( $green\_perc, brown\_perc$ )

```

was turned white (255,255,255) to obtain a clear image. Therefore, after color segmentation, only three clusters of colors remained in the image: green, brown, and white. According to experts at CRISL, these are the three categories that should be present in an infected leaflet. Their rationale is that the color indicative of CCI can be effectively differentiated from colors associated with other types of infections. Furthermore, when CCI is present, it tends to dominate the leaf, suggesting that there are typically no other pests present. As a result of this insight, the brown pixels represent only the necrotic leaf regions.

K-means has also been proven effective in similar agricultural studies for color segmentation [52], [53]. Therefore, K-means clustering was used to assign each pixel a specific cluster label. Given that our segmentation relied on clear color distinctions—green representing healthy regions and brown indicating infected areas (necrotic regions), Additionally, since we had a prior knowledge of the required clusters, we set the number of clusters (K) to three, corresponding green, brown, and background areas. Using algorithm 2, we then counted the pixels labeled as green and brown, enabling a clear and accurate quantification of healthy and infected areas. Finally, the progression of the disease was calculated by determining the ratio of the necrotic region to the entire leaf region (which includes the total pixel values of both brown and green). Consequently, in addition to the above described the algorithm 2 outlines the process of applying color segmentation, performing K-means clustering, and calculating the progression level.



### G. DETECT AND COUNT COCONUT CATERPILLARS

According to the expert survey during our discussion with the CRISL officers, they currently count the caterpillars manually, by collecting them onto a piece of paper. Based on the number of caterpillars on each coconut leaf, they identify the severity of the infestation. However, despite employing this method, we found it to be not only inaccurate at times but also a tedious task, as the caterpillars are constantly moving., which makes counting them a difficult task. Moreover, this process is time-consuming and prone to human error. Therefore, as one of the objectives of this study, the manual process was automated using image processing techniques. Consequently, the YOLO object detection model [42]–[44] was used, as an improvement of the manual process, to count the caterpillars in their natural habitat (leaflet itself) without forcefully extracting them onto paper. Both methodologies were implemented and evaluated to determine which method is more efficient and accurate.

#### 1) Counting caterpillars using image processing

Initially, Gaussian Blur was applied after converting the image into greyscale. Next, the image was converted to a binary image using thresholding after flipping the pixel values using bitwise conversion to see the available objects. All the dark regions were turned black while the light regions were turned white. This was used to prevent a wide range of colors from being present in the image. After that, erosion (i.e., a morphological operation) was used to reduce the features of the image (shrinking the foreground), as it is capable of removing minor noises from images. Erosion can eliminate dust and soil particles that come with the caterpillars. It was also used to separate and identify caterpillars that were closely attached.

Finally, the connected components were detected, which are defined as neighboring pixel regions with the same input value. Methodically, the algorithm searches for white pixels, assigning each pixel with a unique label ID and applying the same label to all neighboring white pixels iteratively until the entire caterpillar is covered and labeled. Then it moves on to the next accessible white pixel and repeats the process until the entire image is analyzed. Through this process, all the connected components were marked and calculated to detect the number of caterpillars available in the image.

#### 2) Counting caterpillars using object detection (YOLO)

As stated previously, this study used YOLOv5, YOLOv8, and YOLO11 object detection models to calculate caterpillars owing their considerably high accuracy rates [54]–[56]. The extra large (x) variants (i.e., YOLOv5x, YOLOv8x, and YOLO11x) were employed in this study because the x models achieve significantly higher accuracy compared to other variants, making them well-suited for tasks requiring precise detections [44]. For each version, the `depth_multiple` and `width_multiple` parameters were maintained at their default values as specified in the original implementations of the respective models [44]. This ensured that the

TABLE 3: Selecting The Best CNN Architecture by Comparing Testing Accuracies for Each Architecture (%)

DL Architectures	WCLWD	
	Classification	Symptom Severity
ResNet50	76.92	61.32
<b>DenseNet121</b>	<b>90.00</b>	88.67
ResNet50V2	85.95	87.73
<b>InceptionResNetV2</b>	81.20	<b>97.00</b>
InceptionV3	84.62	77.35
MobileNetV2	78.63	95.28
Xception	84.62	82.07
VGG16	-	90.56

structural integrity of each model remained consistent with its design. The YOLO models were trained on a custom dataset of the coconut caterpillar class. The detection technique was altered to compute the number of caterpillars for each classification.

Similar to Mask R-CNN, the regions of interest are marked based on IoU values. Following non-max suppression, which ensures the object detection algorithm detects each object only once, it returns detected objects with bounding boxes.

## IV. TEST SETUP IMPLEMENTATION, RESULTS AND DISCUSSION

The tests were conducted on a server with the following configurations: GPUs (NVIDIA Tesla T4 [57], 16 GB GDDR6 memory @ 300 GBps), CPUs (2 × vCPU), and 24 GB of RAM. The server was implemented in the Google Colab environment using Python version 3.10.12 and performed under the deep learning development framework of TensorFlow and Keras. The training and testing were conducted primarily on the GPU to leverage its parallel processing capabilities for deep learning tasks.

### A. EVALUATION OF WCLWD IDENTIFICATION AND CLASSIFICATION

As shown in Table 2, the dataset was initially divided into training and validation. Then a separate dataset was collected for the testing purposes. The models for WCLWD, and its symptom severity were trained using several architectures as summarized in Table 3, and by considering the accuracy, the best architecture was chosen. The loss values were considered to define the model’s performance after each iteration of optimization. Table 3 shows the testing accuracies for WCLWD classification and symptom severity.

Accuracy (6), confusion matrix, precision (7), recall (8), and F1-score (9), as explained in [58], are used to analyze the performance of both WCLWD classification and symptom severity models. The Accuracy is defined as the ratio of correct predictions to the total number of predictions, as given by (6), where True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) represent cases when the model predicts the positive class as positive (i.e., TP) or as negative (i.e., FN), and predicts the negative class



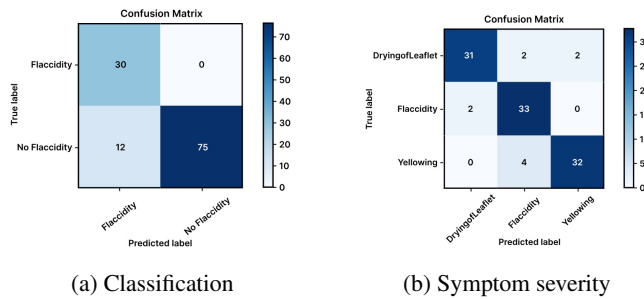


FIGURE 3: Confusion matrices WCLWD

as positive (i.e., FP) or as negative (i.e., TN), respectively.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (6)$$

The Precision (7) is defined as the ratio of correctly predicted positive observations to the total predicted positive observations:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (7)$$

The Recall (8), also known as sensitivity, is the ratio of correctly predicted positive observations to all actual positives:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (8)$$

The F1 score (9) is the harmonic mean of Precision and Recall, providing a balance between the two metrics:

$$F_1 \text{ score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (9)$$

The WCLWD classification model using the DenseNet121 achieved 99.03% training accuracy and 90% testing accuracy which performs the best test accuracy among several architectural models. Moreover, the InceptionResNetV2-based symptom severity model achieved training and testing accuracies of 99.68% and 97%, respectively. Figure 3 illustrates the confusion matrices related to WCLWD classification (a) and symptom severity (b), which help to evaluate the performance of the trained models. In addition, Fig. 4 represents the loss and accuracy plots of both models DenseNet121 (a and b) and InceptionResNetV2 (c and d) respectively.

The precision, recall, and F1-score of the DenseNet121 model used for WCLWD identification and classification were 71%, 100%, and 83%, respectively. Similarly, the InceptionResNetV2 model, which was used to determine symptom severity achieved 99%, 99%, and 98%.

## B. EVALUATION OF CCI IDENTIFICATION AND CLASSIFICATION

As explained previously, the identification, classification and progression level determination of CCI was achieved using Mask R-CNN and K-means clustering. The models were

trained in the above-mentioned environment using 60 validation steps, a detection confidence of 0.9, 100 steps per epoch, and three classes namely, with CCI, without CCI, and background.

The trained model was able to accurately distinguish between infested and non-infested leaflets (healthy and other diseases). Fig. 5 illustrates the outcome of the model. Suspicious leaflets (See Fig. 5. (a)) were sent through the Mask-R CNN model for detection. After training the model, as shown in Fig. 5. (b), the results of instance segmentation (masking) of either infested or non-infested leaflets were created. Simultaneously, identification, classification and bounding-box regression were performed to mark the region of leaflets with labels. A rectangular box was drawn around the object after classification. Finally, the FCN layer masks the images according to the coordinates.

After the leaf has been classified and masked, the class label of the image was checked. If the label identifies the image as infested, crop segmentation was performed to separate the mask region (leaflet) from the background (c) as shown in Fig. 5. Only the leaf was extracted while the background was separated using the mask coordination and dimensions. This is a clear example where instance segmentation of Mask RCNN is useful. Following crop segmentation, the HSV range of green colors was converted into a single RGB value (20. 255. 10). Similarly, the brown color caused by caterpillar damage (necrotic regions) was also converted to single RGB color (19. 69. 139). The rest of the colors were turned to white before applying k-means clustering algorithm (see Fig. 5. (d)) to label each pixels within same cluster. The reason for the color conversion is to minimize the number of clusters needed.

Finally, using the 3 clusters of colors (white, green and brown), the number of brown pixels with respect to the leaf area (brown + green) is calculated providing the progression level of the infestation throughout the leaflet. The analysis is more reliable because all pixels are analyzed, including small patches.

For the evaluation of models, performance was assessed by comparing the annotated images with the prediction results during the training process by considering the loss values. After training, metrics such as precision (7), recall (8), Average Precision (AP) (10) and mean Average Precision (mAP) (11) value were computed to assess the performance as explained in [59] and [58]. In object detection, AP is often computed based on the precision-recall curve, which aggregates precision values at different recall levels using (10), Where  $P(r_i)$  is the precision at a given recall level  $r_i$ .

$$AP = \frac{1}{n} \sum_{i=1}^n P(r_i) \quad (10)$$

The mAP scores across all classes in the dataset were calculated using (11), Where  $C$  is the total number of classes, and  $AP_i$  is the Average Precision for class  $i$ .

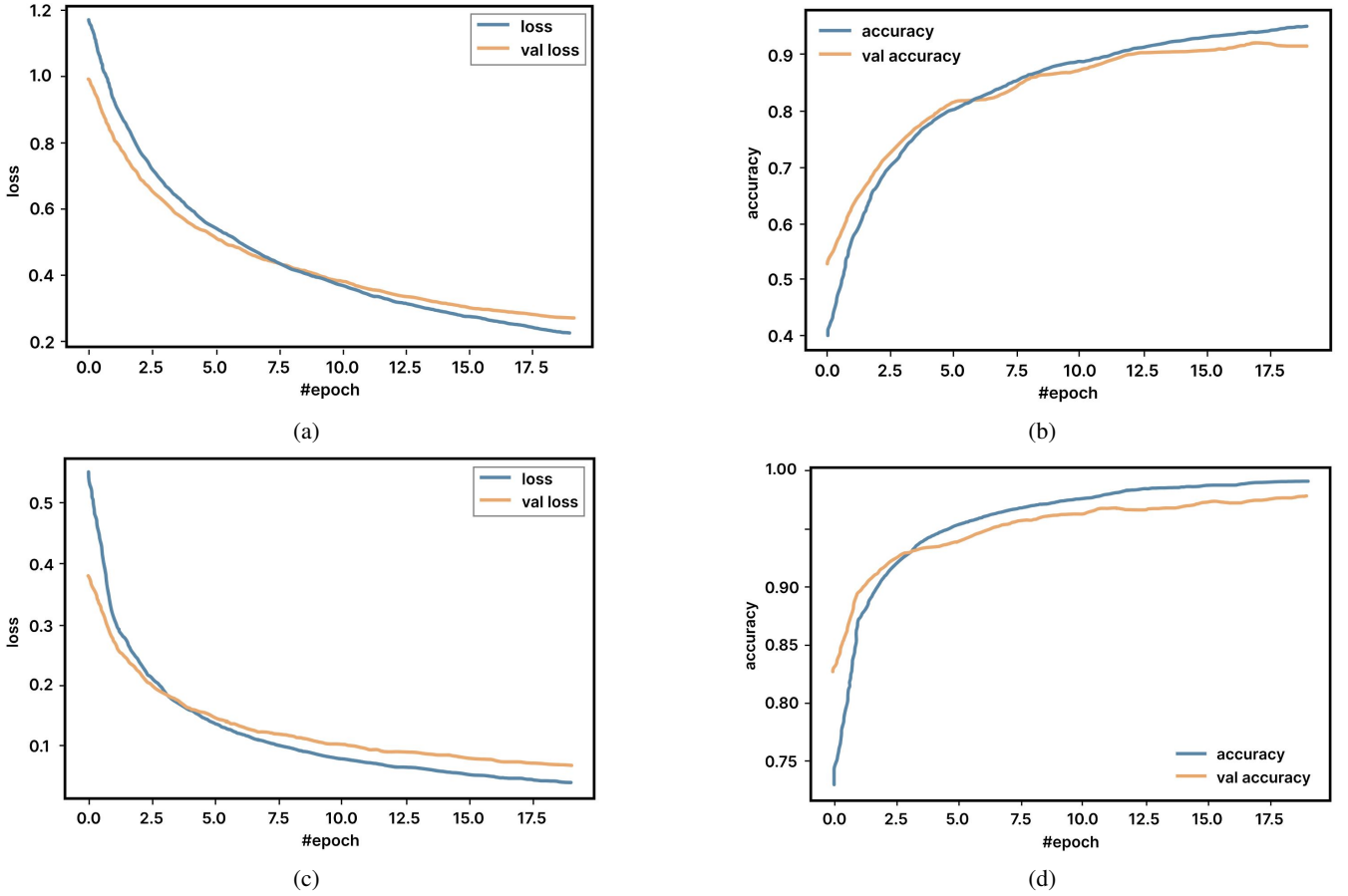


FIGURE 4: Loss and accuracy plots: a. Loss curve for WCLWD classification, b. Accuracy curve for WCLWD classification, c. Loss curve for symptom severity, d. Accuracy curve for symptom severity

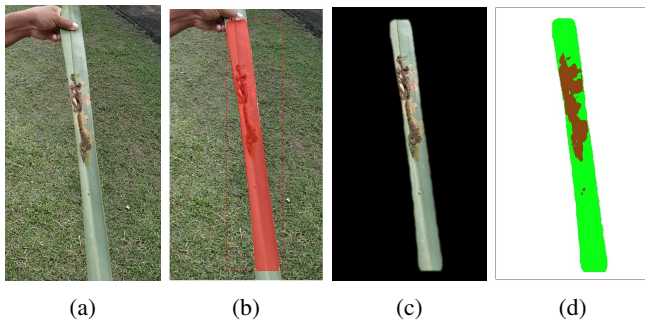


FIGURE 5: Process of calculating the extent of damage: a. Original image, b. Masking, c. Crop segmentation, d. Color segmentation

$$mAP = \frac{1}{C} \sum_{i=1}^C AP_i \quad (11)$$

The Mask R-CNN model for CCI identification and classification has performed well in identifying true predictions by achieving precision and recall values of 92% and 89% respectively. The respective graphs are shown in Fig. 6.

Finally, with the use of the above metrics mAP is calculated, and the ResNet101 architecture-based Mask R-CNN model achieved high detection rates with a mAP value of 95.26%.

### C. EVALUATION OF COCONUT CATERPILLAR DETECTION AND COUNTING

#### 1) Counting caterpillars using object detection (YOLO)

The YOLO algorithms generated output for both categorization and localization of caterpillars in leaflets. All the YOLO models were trained using hyperparameter configurations on the same computational system to maintain consistency and ensure fair comparability throughout the process. Key training parameters included setting the initial and final learning rates to 0.0001 and 0.01, respectively, ensuring a gradual and controlled learning process. Momentum was fixed at 0.937 to maintain stable updates across epochs, while a minimal weight decay of 0.0005 helped prevent overfitting. The batch size was set at 8, optimizing the balance between memory usage and processing speed. The models were trained on a custom coconut caterpillar dataset, the detection technique was modified to classify two classes (caterpillars and background) and finally, the detection technique was altered to

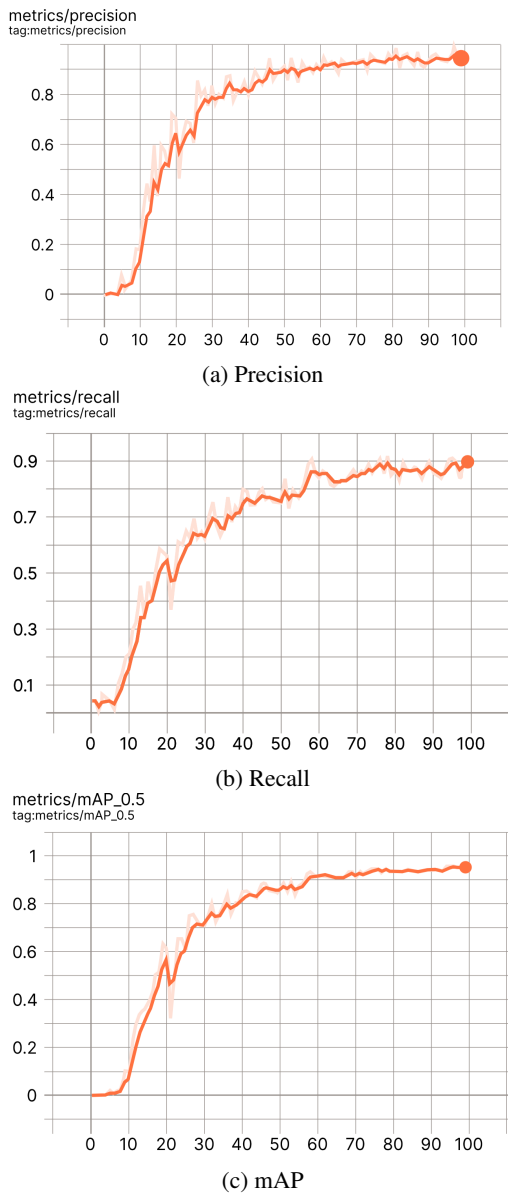


FIGURE 6: Accuracy graphs for Mask R-CNN model: a. Precision graph, b. Recall graph, c. mAP graph.

compute the number of caterpillars for each classification. Each model was trained for 100 epochs, providing sufficient time to learn and adapt to the dataset's complexities.

Fig. 7 shows a sample image of identified caterpillars as well as the result of calculating the number of caterpillars in the image. The comparison among the YOLO modes is given in Figs. 8, 9, and 10 for YOLOv5, YOLOv8 and YOLO11 respectively. The summary of the results for each model is shown in Table 4. As shown in the table, it can be observed that the YOLOv5 clearly outperforms the other two models.

## 2) Counting caterpillars using image processing

The automated caterpillar counting process, which simulates the existing manual method of placing caterpillars on paper

Model	Precision (%)	Recall (%)	mAP (%)
YOLOv5	93	92.5	96.87
YOLOv8	90.3	92.3	96.1
YOLO11	91	92.2	95.9

TABLE 4: Performance metrics for YOLOv5, YOLOv8, and YOLO11.

for counting, is illustrated in Fig. 11. As illustrated in Fig. 11. (a) the captured image was turned into greyscale while using Gaussian Blur to remove background noises to a certain extent. Then thresholding was applied, as shown in Fig. 11. (b), and that helped to clearly distinguish all caterpillars and small dust particles (if available) with the background. To remove the remaining noises (dust particles) erosion was used and the resultant image was given in Fig. 11. (c). This technique was also effective in separating caterpillars that are close or attached. Finally, all the caterpillars are detected by finding the connected components (See Fig. 11. (d)).

Although this process was fairly accurate, the counting results can be incorrect when the image lighting conditions vary, as this method relies on consistent image processing factors to provide accurate results. Additionally, if the image perception differs, small caterpillars can be removed by the erode function. This research demonstrated that the automated manual process is less accurate compared to the state-of-the-art object detection model mentioned in IV-C1.

## V. DISCUSSION

When comparing the existing studies on coconut disease classification, such as [29], [30], and [31], they mainly focused on identifying diseases in later stages, when symptoms have progressed visibly. However, it is a fact that identifying diseases in the early stages will give farmers a comparative advantage to apply remedial solutions. Consequently, the proposed method can detect WCLWD with an accuracy of 90% in the early stages by correctly identifying flaccidity. Flaccidity, which is subtle and often indistinguishable even for experts, marks the initial stages of WCLWD. Hence, the early detection capability of this study addresses a critical gap in current practices and provides significant potential for timely intervention.

Moreover, the proposed method extends beyond disease classification to include severity assessment, an area that has received limited attention in the prior literature. By utilizing inceptionResNetV2 for severity assessment, we achieved an accuracy of 97%. This dual approach not only enhances the precision of detection but also provides actionable insights into disease progression, enabling more informed decision-making in disease management.

Similarly, traditional and manual methods of counting coconut caterpillars are labor-intensive and prone to inconsistencies, especially under variable conditions such as light conditions and movements of caterpillars. Although, image-processing techniques offer some automation, their accuracy is heavily dependent on the parameters (i.e., lighting conditions, image perception, noise, etc.). Also, limitations such as



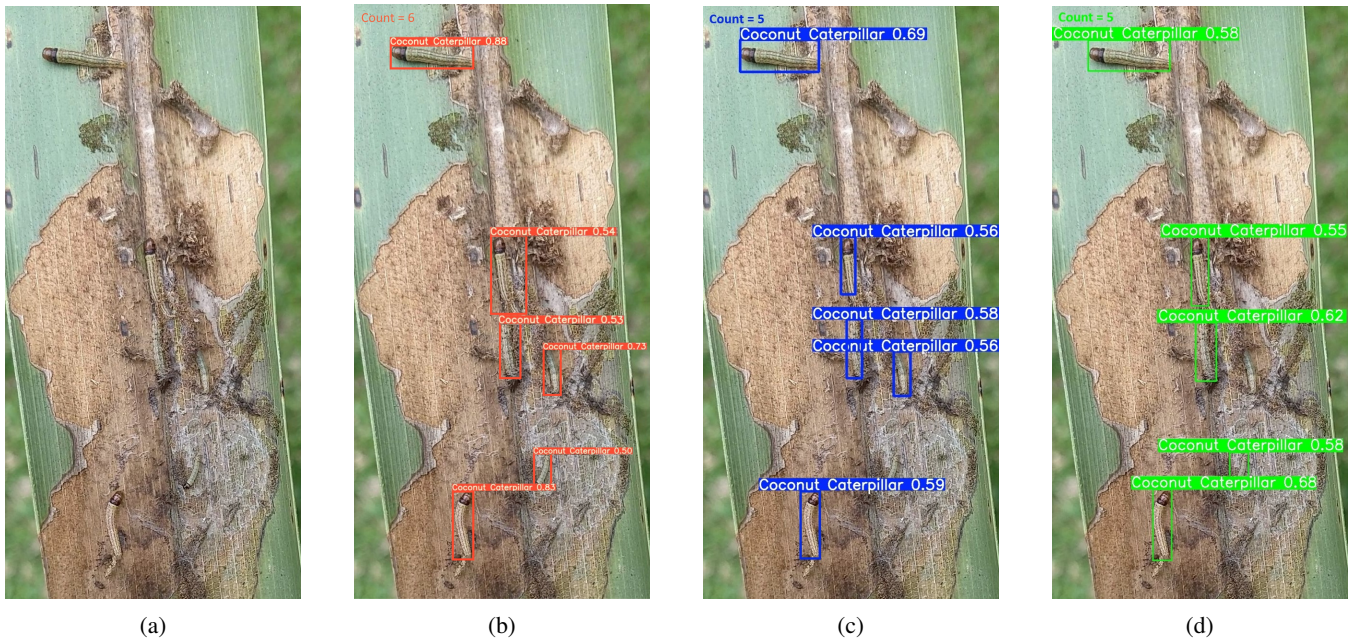


FIGURE 7: Counting the number of caterpillars using YOLO models: (a) Original image with 6 caterpillars, (b) YOLOv5x detection (6/6 correctly identified), (c) YOLOv8x detection (5/6 correctly identified), (d) YOLOv11x detection (5/6 correctly identified).

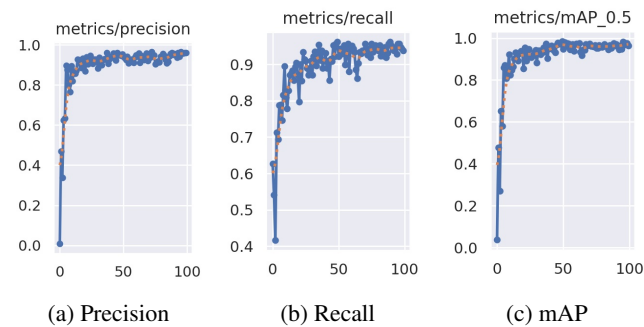


FIGURE 8: Graphs for YOLOv5 model: a. Precision graph, b. Recall graph, c. mAP graph

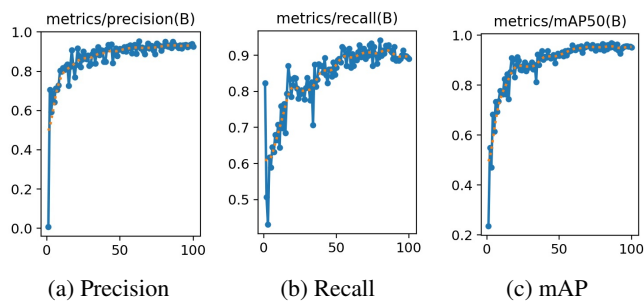


FIGURE 9: Graphs for YOLOv8 model: a. Precision graph, b. Recall graph, c. mAP graph

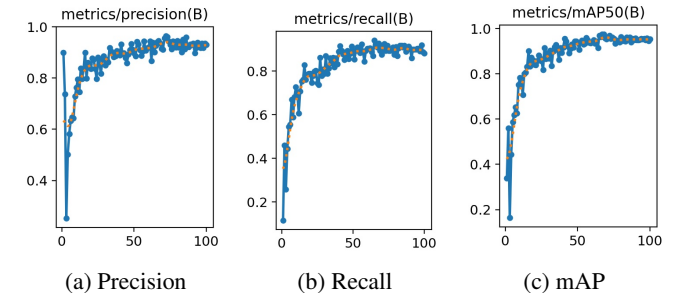


FIGURE 10: Graphs for YOLO11 model: a. Precision graph, b. Recall graph, c. mAP graph

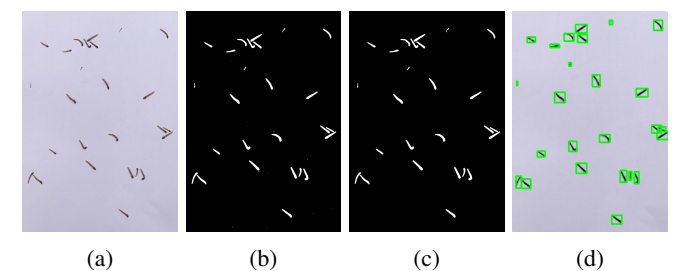


FIGURE 11: Results of calculating caterpillars using image processing: (a) Original image, (b) Thresholding, (c) Erosion, (d) Connected components

removal of smaller caterpillars during preprocessing is highly affecting the accuracy of the caterpillar counts.

Our adoption of YOLO-based object detection models significantly improved both accuracy and efficiency of the CCI detection. An analysis was conducted, using 100 images, to



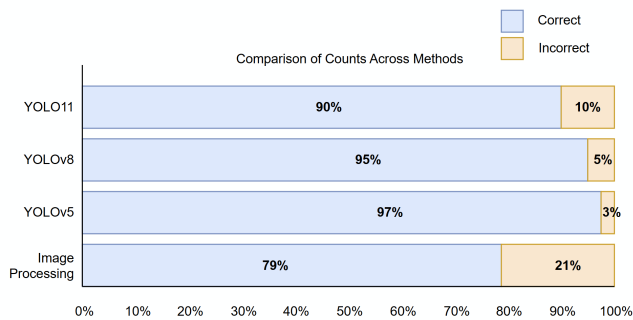


FIGURE 12: Comparison between Image Processing Technique and YOLO (YOLOv5, YOLOv8 & YOLO11) Object Detection Model on 100 images against ground truth results.

compare the accuracies of both YOLO models and the image processing technique against a human counting performed by scientists in CRISL (this was considered as the ground truth). The results obtained are given in Fig. 12. As illustrated in the figure, the YOLOv5 model correctly identified the number of caterpillars in 97 of 100 images, leaving 3 images with incorrect values. The YOLOv8 model correctly identified the number of caterpillars in 95 of 100 images, leaving 5 images with discrepancies, while the YOLO11 model accurately identified caterpillars in 90 of 100 images, leaving 10 images with incorrect values.

Subsequently, among the tested YOLO models, YOLOv5 emerged as the most effective model by achieving a mAP of 96.87%. Hence, as given in Table 4 and Fig. 12, it can be observed that YOLOv5 outperforms both YOLOv8 and YOLO11 in accuracy, precision and recall; YOLOv5 model performance surpasses even the more recent YOLO11 model, underscoring the importance of context-specific model evaluation despite we used the default `depth_multiple` and `width_multiple` parameters for all the tested YOLO versions.

Notably, the image processing technique was capable of accurately identifying only 79 of 100 images. This shows that automated caterpillar detection with YOLO models not only addresses the inefficiencies of manual counting but also ensures reliable results under diverse environmental conditions.

The variety of image resolutions in our dataset played a major role in the success of our models. By using images of different qualities and from different places of the country, we ensured that the models could handle real-world scenarios effectively. This approach allowed our models to perform well even when tested on lower-quality images, highlighting the importance of diverse data during training and testing.

## VI. CONCLUSION

This research study was carried out in Matara, Puttalam, and Makandura in Sri Lanka to find the effectiveness of using Deep Learning techniques such as classification, instance segmentation, and object detection along with image processing techniques for early identification of WCLWD and

CCI. The disease classification and the disease severity were determined using CNN, Mask R-CNN, and YOLO models and were able to yield accuracy ranging from 90% to 97%. Furthermore, the manual and tedious caterpillar identification process was automated to minimize human inaccuracies and comparative results show that the automated method implemented using YOLOv5 identified caterpillars with only a 3% error rate.

In the future, we will work closely with the Coconut Research Institute of Sri Lanka to enhance the early identification of other coconut diseases and pest infestations. We will also explore the use of different architectures for disease detection and severity assessment, to evaluate and compare the performance of models for both WCLWD and CCI. To address real-world applicability, the study will be extended to handle more complex field conditions, such as overlapping leaves, varying orientations (front and back surfaces). Additionally, as a key aspect of future work, Explainable Artificial Intelligence (XAI) techniques will also be incorporated to provide insights into the decision-making process of the models, improving their interpretability. Data from other regions of the country will also be utilized to refine and enhance model accuracy. Finally, the testing will be expanded to other coconut-growing regions, including Kerala, India, to provide sustainable solutions globally. We aim to encourage international researchers to freely use and contribute to the dataset published in [33], further automating this critical industry.

## VII. ACKNOWLEDGEMENTS

This research was supported by the Coconut Research Institute of Sri Lanka (CRISL) and the Sri Lanka Institute of Information Technology (SLIIT). The authors would also like to express their sincere gratitude to Dr. Mrs. Nayanie Aratchige, Deputy Director (Research) at the Coconut Research Institute of Sri Lanka, for facilitating the initial field research experiments. Special thanks are extended to Dr. I. M. S. K. Idirisinghe, Mr. P. H. P. Roshan de Silva, Dr. Mrs. Vijitha Vidhanarachchi, Mr. Anura Pathirana, and Mr. S. P. Manoj for their support in field testing and for providing technical expertise for field sampling. Additionally, we wish to extend our gratitude to R. P. T. I. Gunasekara, P. K. G. C. Akalanka, and H. M. U. D. Rajapaksha for their assistance in conducting the initial experiments and data collection.

## REFERENCES

- [1] A. D. N. T. Kumara, M. Chandrashekharaiah, S. B. Kandakoor, and A. K. Chakravarthy, *Status and Management of Three Major Insect Pests of Coconut in the Tropics and Subtropics*. New Delhi: Springer India, 2015, pp. 359–381. [Online]. Available: [https://doi.org/10.1007/978-81-322-2089-3\\_32](https://doi.org/10.1007/978-81-322-2089-3_32)
- [2] A. Chandy, "Pest infestation identification in coconut trees using deep learning," *Journal of Artificial Intelligence and Capsule Networks*, vol. 01, pp. 10–18, 09 2019.
- [3] T. Wijesekara, L. Perera, I. Wikramananda, I. Herath, M. Meegahakumbura, W. Fernando, and P. de Silva, "Preliminary investigation on weligama coconut leaf wilt disease: A new disease in southern part of sri lanka," in *Proceedings of the Second Plantation Crop Symposium*, N. Nainanayake and J. Everard, Eds., Colombo, Sri Lanka, 2008, pp.

- 336–341. [Online]. Available: [https://www.researchgate.net/publication/277010023\\_Preliminary\\_Investigation\\_on\\_Weligama\\_Coconut\\_Leaf\\_Wilt\\_Disease\\_A\\_New\\_Disease\\_in\\_Southern\\_Part\\_of\\_Sri\\_Lanka](https://www.researchgate.net/publication/277010023_Preliminary_Investigation_on_Weligama_Coconut_Leaf_Wilt_Disease_A_New_Disease_in_Southern_Part_of_Sri_Lanka)
- [4] P. de Silva, N. Aratchige, C. Ranasinghe, and L. Perera, “Diseased palm removals as a strategy for the successful management of weligama coconut leaf wilt phytoplasma disease of coconut in sri lanka,” *Phytopathogenic Mollicutes*, vol. Vol. 11 (2), pp. 79–85, 03 2022.
- [5] N. t. Kumara, L. Perera, M. Meegahakumbura, N. Aratchige, and L. Fernando, “Identification of putative vectors of weligama coconut leaf wilt disease in sri lanka,” pp. 137–146, 04 2015. [Online]. Available: [https://www.researchgate.net/publication/276122244\\_Identification\\_of\\_Putative\\_Vectors\\_of\\_Weligama\\_Coconut\\_Leaf\\_Wilt\\_Disease\\_in\\_Sri\\_Lanka](https://www.researchgate.net/publication/276122244_Identification_of_Putative_Vectors_of_Weligama_Coconut_Leaf_Wilt_Disease_in_Sri_Lanka)
- [6] L. Perera, M. Meegahakumbura, T. Wijesekara, W. Fernando, and M. Dickinson, “A phytoplasma is associated with the waligama coconut leaf wilt disease in sri lanka.” *Journal of Plant Pathology*, vol. 94, pp. 205–209, 03 2012.
- [7] S. P. P. J. S. Company, “Coconut damaging black-headed caterpillar,” 2023. [Online]. Available: <http://www.spchcmc.vn/EN/Plant-doctor-Detail/Coconut-Damaging-Black-Headed-Caterpillar-5-8392.html>
- [8] N. B. V. Rao, A. Nischala, G. Ramanandam, and M. H.P., “Biological suppression of coconut black headed caterpillar opisina arenosella outbreak in east godavari district of andhra pradesh-eco friendly technology,” *Current science*, vol. 115, p. 1588, 11 2018.
- [9] S. L. Coconut Research Institute. (2023) Coconut research institute sri lanka. [Online]. Available: <https://cri.gov.lk/en/our-team/>
- [10] P. de Silva, C. Perera, B. Bahder, and R. Attanayake, “Nested pcr-based rapid detection of phytoplasma leaf wilt disease of coconut in sri lanka and systemic movement of the pathogen,” *Pathogens*, vol. 12, 02 2023.
- [11] N. D, T. Weerakkody, W. HTR, P. Waidyaratne, and S. WGR, “Impact of weligama coconut leaf wilt disease (wclwd) on morphological, physiological and yield aspects of coconut palms,” in *Third symposium on plantation crop research Stakeholder empowerment through technological advances*, 01 2010. [Online]. Available: [https://www.researchgate.net/publication/281445930\\_Impact\\_of\\_Weligama\\_Coconut\\_Leaf\\_Wilt\\_Diseases\\_e\\_WCLWD\\_on\\_Morphological\\_Physiological\\_and\\_Yield\\_Aspects\\_of\\_Coconut\\_Palms](https://www.researchgate.net/publication/281445930_Impact_of_Weligama_Coconut_Leaf_Wilt_Diseases_e_WCLWD_on_Morphological_Physiological_and_Yield_Aspects_of_Coconut_Palms)
- [12] P. Dharmadhikari, P. Perera, and T. Hassen, “A short account of the biological control of promecotheca cumingi [col.: Hispidae] the coconut leaf-miner, in sri lanka,” *Entomophaga*, vol. 22, pp. 3–18, 1977. [Online]. Available: <https://doi.org/10.1007/BF02372985>
- [13] S. P. Vidhanaarachchi, P. Akalanka, R. Gunasekara, H. Rajapaksha, N. Aratchige, D. Lunugalage, and J. L. Wijekoon, “Deep learning-based surveillance system for coconut disease and pest infestation identification,” in *TENCON 2021 - 2021 IEEE Region 10 Conference (TENCON)*, 2021, pp. 405–410. [Online]. Available: <https://doi.org/10.1109/TENCON54134.2021.9707404>
- [14] A. Kocian and L. Inerocci, “Learning from data to optimize control in precision farming,” *Stats*, vol. 3, no. 3, pp. 239–245, 2020. [Online]. Available: <https://www.mdpi.com/2571-905X/3/3/18>
- [15] S. Coulibaly, B. Kamsu-Foguem, D. Kamissoko, and D. Traore, “Deep learning for precision agriculture: A bibliometric analysis,” *Intelligent Systems with Applications*, vol. 16, p. 200102, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2667305322000400>
- [16] E. A. Lins, J. P. M. Rodriguez, S. I. Scolski, J. Pivato, M. B. Lima, J. M. C. Fernandes, P. R. V. da Silva Pereira, D. Lau, and R. Rieder, “A method for counting and classifying aphids using computer vision,” *Computers and Electronics in Agriculture*, vol. 169, p. 105200, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0168169919306039>
- [17] K. A. M. Han and U. Watchareeruetai, “Classification of nutrient deficiency in black gram using deep convolutional neural networks,” in *2019 16th International Joint Conference on Computer Science and Software Engineering (ICSSSE)*, 2019, pp. 277–282.
- [18] Y. Yu, K. Zhang, L. Yang, and D. Zhang, “Fruit detection for strawberry harvesting robot in non-structural environment based on mask-rcnn,” *Computers and Electronics in Agriculture*, vol. 163, p. 104846, 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0168169919301103>
- [19] L. Francl and S. Panigrahi, “Artificial neural network models of wheat leaf wetness,” *Agricultural and Forest Meteorology*, vol. 88, no. 1, pp. 57–65, 1997. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0168192397000518>
- [20] F. Hahn, I. Lopez, and G. Hernandez, “Spectral detection and neural network discrimination of rhizopus stolonifer spores on red tomatoes,” *Biosystems Engineering*, vol. 89, no. 1, pp. 93–99, 2004. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1537511004000406>
- [21] S. Sladojevic, M. Arsenovic, A. Anderla, and D. Stefanovi, “Deep neural networks based recognition of plant diseases by leaf image classification,” *Computational Intelligence and Neuroscience*, vol. 2016, pp. 1–11, 06 2016.
- [22] J. L. Wijekoon, D. Nawinna, E. Gamage, Y. Samarawickrama, R. Miriyagalla, D. Rathnaweera, and L. Liyanage, *An Intelligent System for Crop Disease Identification and Dispersion Forecasting in Sri Lanka*. Singapore: Springer Singapore, 2022, pp. 187–205. [Online]. Available: [https://doi.org/10.1007/978-981-16-9991-7\\_12](https://doi.org/10.1007/978-981-16-9991-7_12)
- [23] K.-Y. Huang, “Application of artificial neural network for detecting phalaenopsis seedling diseases using color and texture features,” *Computers and Electronics in Agriculture*, vol. 57, pp. 3–11, 05 2007.
- [24] Q. Wang, F. Qi, M. Sun, J. Qu, and J. Xue, “Corrigendum to “identification of tomato disease types and detection of infected areas based on deep convolutional neural networks and object detection techniques” ,” *Computational Intelligence and Neuroscience*, vol. 2021, pp. 1–1, 02 2021.
- [25] B. Geetha, V. Jiwatode, R. Raut, M. Hussan, M. Tiwari, and D. Dohhal, “An innovative method for detection of insect based on mask-rcnn approach,” pp. 559–564, 02 2024. [Online]. Available: [https://www.researchgate.net/publication/378122425\\_An\\_Innovative\\_Method\\_for\\_Detection\\_of\\_Insect\\_Based\\_on\\_Mask-R-CNN\\_Approach](https://www.researchgate.net/publication/378122425_An_Innovative_Method_for_Detection_of_Insect_Based_on_Mask-R-CNN_Approach)
- [26] S. Manoharan, B. Sariffodeen, K. Ramasinghe, L. Rajaratne, D. Kasthurirathna, and J. L. Wijekoon, “Smart plant disorder identification using computer vision technology,” in *2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2020, pp. 0445–0451. [Online]. Available: <https://doi.org/10.1109/IEMCON51383.2020.9284919>
- [27] G. Hewawitharana, U. Nawarathne, A. Hassan, L. M. Wijerathna, G. Sinniah, S. P. Vidhanaarachchi, J. Wickramaratne, and J. L. Wijekoon, “Effectiveness of using deep learning for blister blight identification in sri lankan tea,” in *2023 International Research Conference on Smart Computing and Systems Engineering (SCSE)*, vol. 6, 2023, pp. 1–6. [Online]. Available: <https://doi.org/10.1109/SCSE59836.2023.10215029>
- [28] T. Kasinathan and S. R. Uyyala, “Detection of fall armyworm (spodoptera frugiperda) in field crops based on mask r-cnn,” *Signal, Image and Video Processing*, vol. 17, no. 6, pp. 2689–2695, September 2023. [Online]. Available: <https://doi.org/10.1007/s11760-023-02485-3>
- [29] M. Maray, A. A. Albraikan, S. S. Alotaibi, R. Alabdan, M. A. Duhayyim, W. K. Al-Azzawi, and A. alkhayyat, “Artificial intelligence-enabled coconut tree disease detection and classification model for smart agriculture,” *Computers and Electrical Engineering*, vol. 104, p. 108399, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0045790622006164>
- [30] A. Kadethankar, N. Sinha, A. Burman, and V. Hegde, “Deep learning based detection of rhinoceros beetle infestation in coconut trees using drone imagery,” in *Computer Vision and Image Processing*, S. K. Singh, P. Roy, B. Raman, and P. Nagabhushan, Eds. Singapore: Springer Singapore, 2021, pp. 463–474.
- [31] P. Singh, A. Verma, and J. S. R. Alex, “Disease and pest infection detection in coconut tree through deep learning techniques,” *Computers and Electronics in Agriculture*, vol. 182, p. 105986, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0168169921000041>
- [32] R. Miriyagalla, Y. Samarawickrama, D. Rathnaweera, L. Liyanage, D. Kasthurirathna, D. Nawinna, and J. L. Wijekoon, “On the effectiveness of using machine learning and gaussian plume model for plant disease dispersion prediction and simulation,” in *2019 International Conference on Advancements in Computing (ICAC)*, 2019, pp. 317–322. [Online]. Available: <https://doi.org/10.1109/ICAC49085.2019.9103383>
- [33] S. Vidhanaarachchi, “Coconut diseases dataset,” <https://www.kaggle.com/datasets/samitha96/coconutdiseases>, Year of access, accessed: 2023.
- [34] GSMarena, “Apple iphone 6 - full phone specifications,” 2024, accessed: 2024-11-17. [Online]. Available: [https://www.gsmarena.com/apple\\_iphone\\_6-6378.php](https://www.gsmarena.com/apple_iphone_6-6378.php)
- [35] GSMarena, “Apple iPhone 11 - Full phone specifications,” [https://www.gsmarena.com/apple\\_iphone\\_11-9848.php](https://www.gsmarena.com/apple_iphone_11-9848.php), 2024, accessed: 2024-11-17.
- [36] Canon Australia, “Canon EOS 3000D - Digital Camera,” <https://www.canon.com.au/cameras/eos-3000d>, 2024, accessed: 2024-11-17.
- [37] K. Kumara, L. Perera, C. Ranasinghe, W. Fernando, R. Peries, and N. Adikaram, “Prevalence of weligama coconut leaf wilt disease in

- southern sri lanka," *Coconut Research Institute of Sri Lanka*, pp. 1–8, 2015. [Online]. Available: <https://core.ac.uk/download/pdf/52174498.pdf>
- [38] Grow by Coco, "The coconut triangle of sri lanka," <https://growbycoco.com/the-coconut-triangle-of-sri-lanka/>, Year of access, accessed: 2023.
- [39] Visual Geometry Group, University of Oxford, "VGG Image Annotator (VIA)," <https://www.robots.ox.ac.uk/~vgg/software/via/>, Year of access, accessed: 2023.
- [40] S. Bondre and D. Patil, "Crop disease identification segmentation algorithm based on mask-rcnn," *Agronomy Journal*, vol. 116, no. 3, pp. 1088–1098, 2024. [Online]. Available: <https://access.onlinelibrary.wiley.com/doi/abs/10.1002/agj2.21387>
- [41] A. Gawade, "Early-stage apple leaf disease prediction using deep learning," *Bioscience Biotechnology Research Communications*, vol. 14, pp. 40–43, 03 2021.
- [42] Ultralytics, "ultralytics/yolov5: Yolov5 - a state-of-the-art object detection model," 2020, accessed: 2024-07-22. [Online]. Available: <https://github.com/ultralytics/yolov5>
- [43] Autogyro, "Yolo-v8," 2024, accessed: 2024-12-10. [Online]. Available: <https://github.com/autogyro/yolo-V8>
- [44] Ultralytics, "Ultralytics yolo11," 2024, accessed: 2024-12-10. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [45] MakeSense AI, "MakeSense AI," <https://www.makesense.ai/>, accessed: 2023.
- [46] L. Jia, T. Wang, Y. Chen, Y. Zang, X. Li, H. Shi, and L. Gao, "Mobilenet-ca-yolo: An improved yolov7 based on the mobilenetv3 and attention mechanism for rice pests and diseases detection," *Agriculture*, vol. 13, p. 1285, 06 2023.
- [47] B. N. Naik, M. Ramanathan, and P. Ponnusamy, "Refined single-stage object detection deep-learning technique for chilli leaf disease detection," *Journal of Electronic Imaging*, vol. 32, no. 3, p. 033039, 2023. [Online]. Available: <https://doi.org/10.1117/1.JEI.32.3.033039>
- [48] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2961–2969.
- [49] M. A. Rahman and Y. Wang, "Optimizing intersection-over-union in deep neural networks for image segmentation," in *Advances in Visual Computing*, G. Bebis, R. Boyle, B. Parvin, D. Koracin, F. Porikli, S. Skaff, A. Entezari, J. Min, D. Iwai, A. Sadagic, C. Scheidegger, and T. Isenber, Eds. Cham: Springer International Publishing, 2016, pp. 234–244.
- [50] A. Neubeck and L. Van Gool, "Efficient non-maximum suppression," in *18th International Conference on Pattern Recognition (ICPR'06)*, vol. 3, 2006, pp. 850–855.
- [51] S. Wang, G. Sun, B. Zheng, and Y. Du, "A crop image segmentation and extraction algorithm based on mask rcnn," *Entropy*, vol. 23, no. 9, 2021. [Online]. Available: <https://www.mdpi.com/1099-4300/23/9/1160>
- [52] S. M. Javidan, A. Banakar, K. A. Vakilian, and Y. Ampatzidis, "Diagnosis of grape leaf diseases using automatic k-means clustering and machine learning," *Smart Agricultural Technology*, vol. 3, p. 100081, 2023. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S272375522000466>
- [53] D. Bashish, M. Braik, and S. Bani-Ahmad, "Detection and classification of leaf diseases using k-means based segmentation and neural networks based classification," *Information Technology Journal*, 2011.
- [54] M. P. Mathew and T. Y. Mahesh, "Leaf-based disease detection in bell pepper plant using yolo v5," *Signal, Image and Video Processing*, vol. 16, no. 3, pp. 841–847, 2022. [Online]. Available: <https://doi.org/10.1007/s11760-021-02024-y>
- [55] A. Morbekar, A. Parihar, and R. Jadhav, "Crop disease detection using yolo," in *2020 International Conference for Emerging Technology (IN-CET)*, 2020, pp. 1–5.
- [56] R. Sapkota and M. Karkee, "Comparing yolo11 and yolov8 for instance segmentation of occluded and non-occluded immature green fruits in complex orchard environment," 2024. [Online]. Available: <https://arxiv.org/abs/2410.19869>
- [57] NVIDIA, "Tesla T4 Tensor Core Product Brief," <https://www.nvidia.com/content/dam/en-zz/Solutions/Data-Center/tesla-t4/t4-tensor-core-product-brief.pdf>, 2024, accessed: 2024-11-17.
- [58] L. Alzubaidi, J. Zhang, A. Humaidi et al., "Review of deep learning: concepts, cnn architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, p. 53, 2021.
- [59] H. He, H. Xu, Y. Zhang, K. Gao, H. Li, L. Ma, and J. Li, "Mask r-cnn based automated identification and extraction of oil

well sites," *International Journal of Applied Earth Observation and Geoinformation*, vol. 112, p. 102875, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1569843222000772>



SAMITHA VIDHANAARACHCHI (Member, IEEE) received his B.Sc. degree from the Sri Lanka Institute of Information Technology (SLIIT), Sri Lanka. He has been a lecturer in the Department of Computer Science and Software Engineering at SLIIT. His research interests include developing advanced AI and deep learning solutions in agriculture, enhancing educational tools for diverse needs, and exploring innovative applications across various domains.



JANAKA L. WIJEKON (Senior Member, IEEE) received his B.Sc. degree from SLIIT, Sri Lanka, in 2010, and went on to complete his M.Sc. (2013) and Ph.D. (2016) degrees at Keio University, Japan. With a background as a postdoctoral researcher with extensive experience in academia, his expertise centres around leveraging AIoT to advance smart societies. His research focuses on collaborative smart infrastructural design and implementations for secure and efficient resource allocation in smart societies and protocol design for data exchange among stakeholders to improve agricultural productivity.



W.A. SHANAKA P. ABEYSIRIWARDHANA (Member, IEEE) received his B.Sc. (Hons) and M.Sc. degrees in Electrical Engineering from the University of Moratuwa, Sri Lanka in 2014 and 2015, respectively. He received his PhD degree in Engineering from Keio University, Japan in 2021. His research interests includes intelligent control, network function virtualization, and IoT security.



MALITHA WIJESUNDARA (Member, IEEE) earned his BEng degree with Honors in Electronic Engineering from the University of Warwick, United Kingdom in 1998. He then went on to complete his PhD in Computer Engineering at the National University of Singapore. He has extensive experience in Embedded Systems, Wireless Sensor Networks, Media Streaming for E-Learning and Energy Harvesting for Elephant Tracking. His research focuses on developing technology solutions to challenges faced by rural and underprivileged communities. He is a member of IEEE, ISACA and CS(SL).

...