Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.7729102

W-GFPO YOLOV3 AND J-FCM-BASED MULTI-DISEASE PREDICTION IN COFFEE LEAVES

Santhosh Kumar S [1], B K Raghavendra [2]

[1]Research Scholar, Department of Information Science & Engineering, Don Bosco Institute of Technology, Bengaluru, Karnataka, Affiliated to Visvesvaraya Technological University, Karnataka, India. Email: santhoshkumars.cs@gmail.com

[2]Research Supervisor, Professor and Head, Department of Information Science & Engineering, Don Bosco Institute of Technology, Bengaluru, Karnataka, Affiliated to Visvesvaraya Technilogical University, Karnataka, India. Email: raghavendra.bk69@gmail.com

Abstract: Organic acids, amino acids, proteins and carbohydrates are enclosed in coffee leaves. In various applications like coffee leaf tea, therapeutic agents, tobacco substitutes, packaging material, personal hygienic products, ethno medicine and animal feed, coffee leaves were wielded. For preventing disease in coffee plants, research has been done for high potential benefits. Generally, for coffee leaf disease classification, the utilized Deep Learning (DL) models were effectively applied. But, to predict a tiny portion of the disease in the overall leaf area, the prevailing techniques were not effective. Thus, this work proposes a novel technique of Weibull distributed-Golden Flower Pollination Optimization (W-GFPO) YoloV3 and Jaccard Index Fuzzy C-means Clustering (J-FCM)-centric Multi-disease Prediction in Coffee Leaf (WJMPCL). Primarily, to enhance the edge, the images are pre-processed. Then, the image is given to the color spacing model, which enhanced the image and is further given to feature extraction. For training purposes, the features are given to W-GFPO YoloV3 and segmentation is done with labeling. After that, for clustering the images into healthy leaf clusters, coffee leaf miner, phoma tarda, coffee leaf rust, and Iron spot, the labeled image is given to J-FCM. Utilizing coffee leaf images, the proposed mechanism was assessed, which attained superior performance in all experiments when analogized with the prevailing approaches.

Keywords: *coffee leaf miner and Iron spot, Jaccard Index Fuzzy C-means Clustering (J-FCM), phoma tarda, Weibull distributed- Golden Flower Pollination Optimization (W-GFPO),*

Introduction

Since coffee was estimated to be consumed in nearly 200 million kilogram bags in the future, it has been recorded as one of the most widely consumed beverages worldwide. Thus, in almost all countries that are tropical, it is no wonder that coffee cultivation has been widespread [1]. Owing to the climatic conditions and multiple coffee diseases that have persisted for several years, this cultivation might get affected. Seeds, beans, stem, and roots are major parts while considering the multi-diseases of coffee plants [2].Since the photosynthesis

of the plants is carried out by the leaves, they are considered the most important part; thus, the health of the fruit and beans is grounded on the leaves.

Utilizing various mathematical models, numerous research was done for coffee leaf disease prediction. Hemileiavastatrix fungus and Cercosporacoffeicola fungus cause Coffee leaf rust and brown eye spot, respectively, which are coffee leaf diseases[3]. Black rot, which was caused by CorticiumKoleroga, is a threat to coffee plants; also, this was a devasting fungal disease. These fungi are terrible diseases caused by fungal infection; also, for avoiding complete damage to plants, they must be detected quickly [4][5]. Moreover, the coffee plantations are affected by these kinds of diseases; also, the plant will be affected by the higher occurrence of fungus, causing damage exceeding 50%.

To overcome these issues, an automatic coffee leaf disease recognition system development has been proposed previously for detecting Cercospora and rust detection in coffee leaves [6]. These works process in the principle of a texture attribute extraction owing to shapeless forms followed by disease recognition. Here, the affected parts were segmented; then, the diseases were predicted utilizing the recognition algorithms has been [3]. Likewise, extending the texture attributes to two to three types and then recognizing features for disease recognition has also been proposed. Still, these procedures were similar for other plant leaf diseases also and not specified for coffee leaf; thus, they failed in attaining popularity [7].

Naïve Bayes, Random Forest, Decision Tree, and Regression models are the Machine learning (ML) models that have been concentrated on the coffee plants' disease prediction. External support is required for these models, which also takes a long time to make a precious decision. Moreover, in the accurate as well as timely detection of coffee leaf disease prediction, Soft computing played a significant role. However, they lack efficiency and have a lower success rate [8]. Currently, DL models are dominating leaf disease prediction. However, these models have limitations like requiring a large number of training images and overfitting problems, which remain unsolved. For finding the spots of the disease, optimization algorithms like Genetic Algorithm (GA), which has been proposed previously, were employed [9]. But, these algorithms failed to consider the image's color textures. Thus, to overcome these issues, this paper proposes a novel algorithm model for the auto-recognition of coffee leaf diseases.

Problem Definition

For coffee leaf disease prediction, the prevailing models suffered from certain issues. From that, most unsolvable issues were taken, which are enlisted further,

- Previous works mainly concentrated on coffee leaf rust disease (CLF) owing to its fine spot prediction on image processing; also, they did not focus on other diseases owing to color transformation problems in preprocessing and cropping in the infected area.
- Segmentation algorithms wielded in prevailing coffee leaf approaches failed to consider multi-disease in a single leaf and also perceptual completion causes high-level vision issues.
- Conventional histogram features, textural features and shape features are not sufficient for predicting the disease in the tip of the leaf and starting area owing to its inorganic quality.

To overcome these issues, this paper proposes a novel disease prediction model and the major contributions are,

- To develop a coffee leaf disease prediction model for rust, miner, phomatarda, and iron spot utilizing G-CLAHE and PURF pre-processing.
- A novel segmentation algorithm with the W-GFPO-YoloV3 model to consider multi-disease in a single leaf and overcome perceptual completion problems.
- To include features like GIST and PHOG; also, conventional features enhanced the prediction accuracy.

The paper's structure is systemized as: the associated works are discussed in section 2; the proposed methodology is explicated in section 3; the experimental outcomes are elucidated in section 4; lastly, the paper is winded up in section 5.

Literature Survey

The authors [10] propounded a practical model that could segment and classify various sorts of leaf lesions, and estimate the stress severity in coffee leaves utilizing a Convolution Neural Network (CNN). It endured the phases of segmentation as well as severity classification. But, this system required continuous internet for its operation. [11] introduced combined rust management for an organic coffee crop in several coffee-producing countries. For this purpose, the model wielded cultural management, plant extracts usage, resistant cultivars, biological control, along with chemical rust control. Yet, this system was executed only in the field that cannot be emphasized all time. The proffered a strategy[12] that generated masks for deep leaves classification. After that, utilizing Semi-Automated Segmentation (SAS) of leaves along with disordered regions, the outcomes were labeled and are able to mount segmentation accuracy. Nevertheless, the author cited anomalies regarding several misclassified cases of illuminated regions that were further recognized as infections. Examined biotic stresses[13] in coffee leaves for evaluating the few-shot learning's performance in classification along with severity estimation tasks. To analyze the biotic stress, Triplet Networks, as well as Prototypical Networks, were utilized, which attained an accuracy of 96%. However, the training time required by this model was higher when analogized to others. The authors [3] established a coffee leaf disease classification together with the fine-tuning effect on Deep CNNs (DCNNs). Utilizing transfer learning along with fine-tuning, they were trained that attained 95% accuracy. But, it was observed that it becomes more difficult for training them since more layers are added to a network; in addition, their accuracy begins to saturate and then decline. The authors [14] presented a strategy that employed transfer learning and the number of pre-trained CNN models that extracts deep channel characteristics as of the coffee plant leaves' images. Then, the extracted deep features were accumulated by various DL models that attained higher validation. Yet, the DL required large amounts of data for training the network. Examined the vegetation indexes [15] of healthy coffee leaves owing to necrosis, which caused premature leaf falling and decreased the plant's yield and lifespan. The images were obtained from the farm directly and computed for spectral bands utilizing Raster Calculator, which was efficient. Still, it was not applicable to a broad range of datasets. The authors [16] presented a DCNN and optimized it. Further, a difficult task was performed that classified diseases and then an appropriate treatment was performed. This model predicted

disease with better accuracy; however, the time taken by this model to predict was not satisfiable. The authors [17] introduced a DL classification model that was applied with fine-tuning, data augmentation, transfer learning, preprocessing techniques, along with selected hyper-parameters. It exhibited a highly accurate feature extraction but still, the diagnosis of coffee leaf disease for the different diseases was not done. [18] presented DCNNs in the detection as well as classification of healthy and unhealthy coffee plant leaf diseases. For disease classification, this technique attained lower time consumption. However, it was unable to predict any new diseases that occur in the coffee leaf plant. [19] detected the coffee leaf disease, namely Hemileiavastatrix, which was owing to climate change that mounted the cases of this disease. Utilizing an Artificial Intelligence-centric application, this disease was found; also, it performed classification grounded on the spots attained. Yet, the accuracy was much lower for this model and also it concentrated on one disease only. The authors [20] employed GA for addressing the problem of identifying rust in coffee leaves and also suggested the use of fungus and less use of pesticides. Along with GA, it wielded a convolution kernel mask for feature extraction. It exhibited a high precision value but still, its dice coefficient error was higher. The authors [21] propounded an application that was contributed for automatically determining the sort of coffee leaf disease and also pest with the percentage of the injured area. It underwent segmentation, clustering and feature extraction given to extreme learning. It accurately segmented the severity of the diseased leaf. However, the background in the image was neglected, which caused non-resistant results. The authors [22] proffered a DL-centric object detection approach that performed k-means clustering and RGB value quantification. It attempted to detect the coffee leaf disease's severity and then the infection stages were also predicted. But, with fewer symptoms of the disease, five clusters attained a more favorable outcome since the signs of the illnesses were smaller and required a more significant number of clusters to be detected. [23] expounded transfer learning along with various pre-trained CNN models that are modeled for extracting deep characteristics as of coffee leaves' images. Deep features were extracted deeply; also, the ensemble architectures were trained for detecting the classification. But, these models required a large number of training data.

Multidisease In Coffee Leaf Prediction Approach Using Gfpo-Yolov3

Predicting coffee leaf disease utilizing DL has been trendy research; however, it could not predict disease even with small spots. Thus, this paper proposes a model for multi-disease prediction in the coffee leaf utilizing the GFPO-YOLOv3 model as displayed in Figure 1.

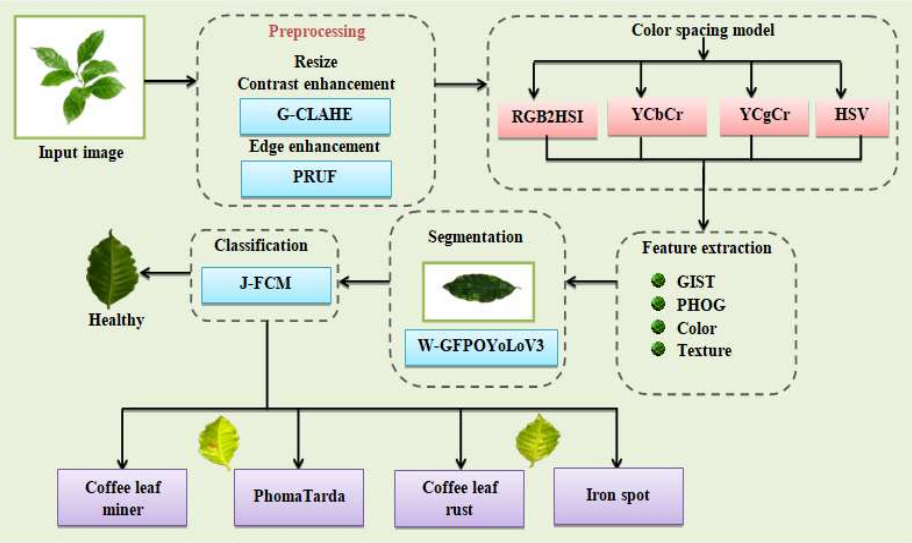

Figure 1. Proposed Framework for coffee leaf multi-disease prediction model.

Pre-Processing

As pre-processing was done to enhance an image, the obtained raw images are given for pre-processing and their distortions are suppressed for finding deep features for further processing. Here, pre-processing was done in three steps: Image resizing, Contrast enhancement, and Edge enhancement.

- ***Image resizing***: Image resizing was done for mounting the total number of pixels for further processing. Let the raw input image be E and the resized image is specified as E_z.
- ***Contrast Enhancement:*** Gaussian-Contrast Limited Adaptive Histogram Equalization (G-CLAHE) has been utilized for contrast enhancement purposes. Here, for enhancement, conventional *CLAHE* was taken owing to its operation even in small regions of the image. Still, it has been enhanced by utilizing the overall Gaussian value of the image to select a threshold. Gaussian value is the weighted average of each pixel's neighborhood. Resized image (E_z) has been given to *G-CLAHE* for contrast enhancement image as (E_{zc})

$$CLAHE = G * \frac{V_{cr-xp} * V_{cr-yp}}{V_{gray}} \quad (1)$$ where,

$V_{cr\text{-}xp}$ signifies the number of pixels in the contextual region *(m)*, $V_{cr\text{-}yp}$ is the number of pixels in the contextual region *(n)* and the number of gray levels in the contextual region is specified as V_{gray}. This *CLAHE* was then clipped utilizing the Gaussian function *(G)* that was computed by the fraction of mean *(mean)* and standard deviation *(std deviation)* values of an image.

$$G = y - mean / std\ deviation \quad (2)$$

$$E_{zc} = GCLAHE\ (E_z) \quad (3)$$

- ***Edge enhancement:*** Pansharpening Rational Unsharp Mask Filter (PRUF) has been proposed for edge enhancement purposes. For extracting and amplifying the target image's high-frequency components, then adding them back to the target image, a

conventional Rational Unsharp Mask Filter has been selected. However, it has the disadvantage of subpixel mapping as it eliminated small values. Thus, it is modified to pansharpening mapping of image values. Primarily, the filter (Ed_i) of PRUF works as,

$$Ed_i = E_{zc} + \vartheta(E_{zc} - z) \quad (4)$$

Where, smoothed image is symbolized as Z that was the low pass filter's output, and the factor of gain is signified as ϑ. After that, by passing through a high-pass filter, the outcomes were modified as,

$$Ed_i = E_{zc} + \vartheta a \quad (5)$$

where, a indicates the input image's high-frequency component. To find a equation (4) is utilized.

$$a = F_y(p,q)C_y(p,q) + F_z(p,q)C_z(p,q) \quad (6)$$

Where,

$$F_y(p,q) = 2y(p,q) - y(p,q-1) - y(p,q+1) \quad (7)$$

$$F_z(p,q) = 2y(p,q) - y(p-1,q) - y(p+1,q) \quad (8)$$

$$C_y(p,q) = \frac{2\kappa[y(p,q+1) - y(p,q-1]^2}{2[y(p,q+1) - y(p,q-1)]^4 + \kappa^2} \quad (9)$$

$$C_z(p,q) = \frac{2\kappa[y(p+1,q) - y(p-1,q)]^2}{2[y(p+1,q) - y(p-1,q)]^4 + \kappa^2} \quad (10)$$

where, K is a constant parameter and $y(p,q)$ represents the pixel intensity value at a location (p,q) in the input image; hence, these rational operations (F_y,F_z) could amplify low and medium-intensity gradient details C_y and C_z. Afterward, applying this rational filter to the original image, its enhanced image is expressed as,

$$E_p = y \oplus \tilde{a} \quad (11)$$

Here, (E_p) symbolizes the enhanced image, $\tilde{a}$ is the mean of the high-frequency component and this image was then mapped utilizing pansharpening mapping. The mapping indexes were obtained by determining the threshold of these vectors. These indexes assist in finding whether the disease spots were highly thresholded at the leaf tip or edge area. It is represented as,

$$E_{pm} = \frac{Perimeter}{2 \times \sqrt{(Area \times \Pi)}} \quad (12)$$

Here, *Perimeter* and *Area* indicate the perimeter and area of the patches correspondingly.

Color Spacing Model

After pre-processing, the image was further given to color spacing models to enhance the luminance and intensity of the image. RGB2HSI, YCbCr, YCgCr, HIS and HSV are the color spacing models utilized here. Primarily, the input image was taken and converted to a grayscale, which was then plotted in a histogram. It might further undergo extraction of color components that produce an enhanced image as depicted in Figure 2.

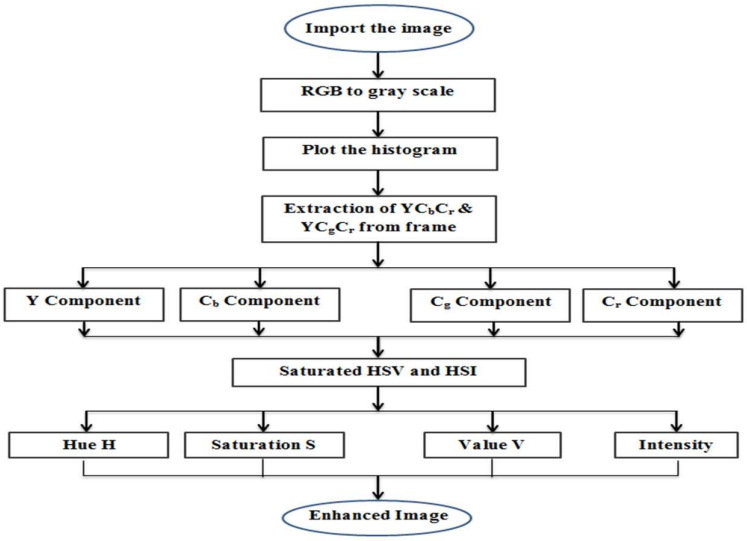

Figure 2. Proposed color space model working procedure

RGB2HSI color spacing

Edge-enhanced images were attained in this space. Images were initially at chrominance Red (α), Green (β) and Blue (δ) values, which were then converted to (Hue Saturation Intensity) HSI as follows,

$$H = 360 - \cos^{-1}\left[\frac{\left(\alpha - \frac{1}{2}\beta - \frac{1}{2}\delta\right)}{\sqrt{\alpha^2 + \beta^2 + \delta^2 - \alpha\beta - \alpha\delta - \beta\delta}}\right] \quad (13)$$

$$S = 1 - \frac{H}{I} \quad (14)$$

where, $I = (\alpha + \beta + \delta)/3$ (15)

HSV color spacing

Same as the HSI model, Hue and saturation were computed for (E_p) with formulation and V is the value or brightness of the color.

$$H = ar\cos\frac{\frac{1}{2}(2\alpha - \beta - \delta)}{\sqrt{(\alpha - \beta)^2 - (\alpha - \delta)(\beta - \delta)}} \quad (16)$$

$$S = \frac{\max(\alpha, \beta, \delta) - \min(\alpha, \beta, \delta)}{\max(\alpha, \beta, \delta)} \quad (17)$$

$$V = \max(\alpha, \beta, \delta) \quad (18)$$

YCbCr Color spacing

In this space, the obtained edge-enhanced image (E_p) is transformed to *YCbCr*, as *Y* is luminance, Blue chrominance is C_b and red is C_r.

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \frac{1}{256}\begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112 \\ 112 & -93.768 & -18.214 \end{bmatrix}\begin{bmatrix} \alpha \\ \beta \\ \delta \end{bmatrix} \quad (19)$$

YCgCr color spacing

Similar to YCbCr, the green chrominance was signified as C_g and this also gets input as edge enhanced image (E_p) and generates *YCgCr* as,

$$\begin{bmatrix} Y \\ Cg \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \frac{1}{256} \begin{bmatrix} 65.481 & 128.553 & 24.996 \\ -81.085 & 112 & -30.915 \\ 112 & -93.768 & -18.214 \end{bmatrix} . \begin{bmatrix} \alpha \\ \beta \\ \delta \end{bmatrix} \quad (20)$$

The overall enhanced image obtained out of color spacing is exemplified as (E_{oe}), and this image was given to the next stage.

Feature extraction

The images obtained (E_{oe}) were then given to the feature extraction to identify multi-diseases in coffee leaves. Features extracted here are both conventional (*Color* and *Texture*) and additional features (*Gist* and *PHOG*). To enhance the classifier's accuracy, additional features were added in feature extraction. The features extracted are explicated further,

$$\Gamma_F = \{\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4\} \quad (21)$$

Texture Features (Γ_1)

These features are extracted since they have important details about the structural arrangement of surfaces with their relationships. Texture features extracted here are,

- *Contrast(Con)*

$$Con = \sum_{p,q=0}^{L-1} y(p,q)(p-q)^2 \quad (22)$$

where, L is the final limit of the intensity value of the image.

- *Homogeneity(Hom)*

$$Hom = \sum_{p,q=0}^{L-1} \frac{y(p,q)}{1+(p-q)^2} \quad (23)$$

- Dissimilarity *(Disim)*

$$Disim = \sum_{p,q=0}^{L-1} y(p,q)|p-q| \quad (24)$$

- Correlation *(corr)*

$$Corr = \sum_{p,q=0}^{L-1} \frac{pqy(p,q) - \eta_p \eta_q}{\vartheta_p \vartheta_q} \quad (25)$$

where,

$$\eta_p = \sum_{y}^{L-1} yp(y,*) \quad (26)$$

$$\eta_q = \sum_{y}^{L-1} yq(*,y) \quad (27)$$

$$\vartheta_p = \sqrt{\sum_{p}^{L-1} (p-\eta_p)^2 p(y,*)} \quad (28)$$

$$\vartheta_q = \sqrt{\sum_{q}^{L-1} (q-\eta_q)^2 q(*,y)} \quad (29)$$

- Variance

$$Var = \sum_{y}^{L-1} |y-M|y(p,q) \quad (30)$$

Color Features (Γ_2)

Color features were extracted for finding the representation of the image's pixel values at position *(p,q).* Let the input image has dimensions $f \times g$. Color features extracted here are,

- Mean (N)

$$N = \frac{\sum_{p}^{f-1} \sum_{q}^{g-1} y(p,q)}{f \times g} \quad (31)$$

- Deviation (φ)

$$\varphi = \frac{\sum_{p}^{f-1} \sum_{q}^{g-1} (y(p,q) - N)^2}{f \times g} \quad (32)$$

Gist Feature (Γ_3)

The Gist feature was selected as it could describe the infected area even with a lower dimensional matrix and further provided a spatial structure of the infected area. This Gist feature has a collection of perceptual dimensions; thus, it can overcome the perceptual completion problem. At various dimensional feature vectors, this feature extraction was done. Primarily, this feature extraction technique filters input images with several low-level visual feature channels like flicker, motion, orientation, color, along with intensity. These filter output maps were computed by,

$$M_y(p,q) = \left| O_y(p) \Theta O_y(q) \right| \quad (33)$$

Where, the mapping filter output is specified as M_y, the pixels were mapped at varying scales, which were denoted as O_y, and scale difference was indicated as Θ . Likewise, these results of feature map when given through orientation channels resulted in,

$$M_y(p) = Gabor(\theta_y, p) \quad (34)$$

Here, θ_y symbolizes the image's different spatial angles. From this computation, a low level of center features was attained.

Pyramid of Histogram of Oriented Gradients (PHOG) features (Γ_4***)***

PHOG features, which describe an image by its local shape along with spatial layout, were spatial descriptors. These features enclosed HOG over every single sub-region of the image at every resolution stage. Primarily, the Histogram of Oriented Gradients (HOG) was computed from an image for varying bin count *(B_c).* PHOG features attained for an image is a weighted combination of the above HOG features *(H_B),* which is expressed as,

$$\Gamma_4 = B_c \in H_B \quad (35)$$

W-GFPO YoloV3 Segmentation

By utilizing the W-GFPO YoloV3 model, the segmentation of multi-diseased areas in leaves was done by taking input image E_{oe}. Here, the conventional YoloV3 model was selected since it has an efficient selection of multi-disease in a single subject. However, utilizing leaky Relu or Relu as the activation function failed to undergo training when a non-linear input pixel neighborhood occurs. Thus, to surpass this issue, Learnable Leaky ReLU is activated, which has the advantage of learning deeply with features and obtaining the outcome. Figure 3 elucidates the overall framework of W-GFPO Yolo V3.

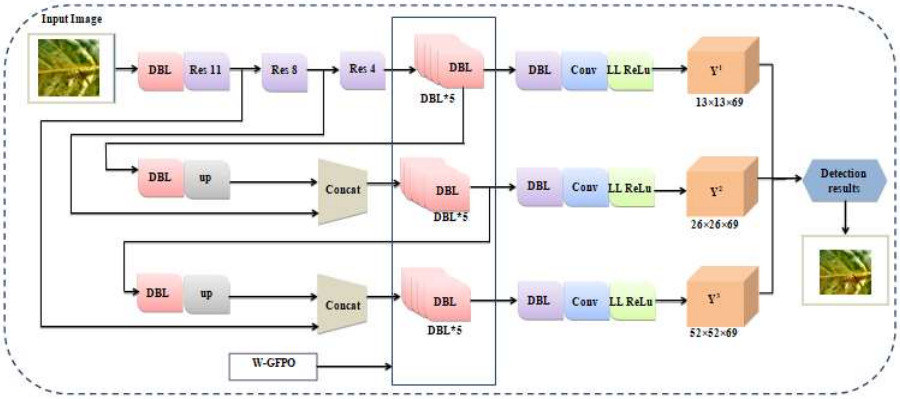

Figure 3. W-GFPO Yolo V3 Architecture

Yolov3 framework utilized DarkNet-53 as the basic network. This model contains a 106-layer network and a significant and familiar structure is made by the deeper network level. A hop layer connection along with a residual module with multi-dimensional detection and upsampling feature fusion technique is enclosed in the framework. Input images given were sampled; also, the target was predicted at the feature map scales' last time included in downsampling and it was trained using the features (Γ). Semantic info was provided by a small feature map, whereas location information was provided by a large feature map. Lastly, these maps get fused and given for detecting both small and large-scale targets.

Weibull distributed- Golden Flower Pollination Optimization algorithm (W-GFPO)

The target score of each bounding box on different scales is predicted by the Yolo-V3 and the values are chosen using W-GPFO. Here, the conventional Golden Flower Pollination Optimization (GFPO) algorithm was selected. Although it has the advantage of finding the infected area of disease with less convergence, still it has a maximum value constraint on the updating stage. Thus, to overcome this issue, it was selected using weigh bull distribution strategy. W-GPFO creates population and solution space. For the searching purpose, Weibull distribution has been adopted and given as follows,

$$Q(x) = \begin{cases} 1 - \left(\frac{w}{fi}\right)^{j} & fi > w \\ 0 & fi < w \end{cases} \quad (36)$$

Here, the value w signifies the scale parameter, j epitomizes the shape parameter and the fitness function of the maximum threshold is exemplified as f_i. After that, this fitness function was selected grounded on the obtained features that are less than the threshold and is given as,

$$fi = Max(\Gamma) \quad (37)$$

At this evaluation stage, a heavy-tailed probability distribution algorithm has been utilized that undergoes a global explorative process and a random selection was done with Weibull distribution given as,

$$fi_{t+1} = fi_t + stp * Q(fi) * (fi_t - g^*) \quad (38)$$

wherein, fi_{t+1} specifies the solution vector at $t+1$ iteration. Likewise, fi_t is the solution vector at t iteration. $Q(fi)$ indicates the Weibull distribution and g^* symbolizes the global solution. Further, the step size is given as stp . Utilizing the switching probability, global and local search

processes were controlled. This probability permits pollens to investigate the global as well as local pollination functions. Pollens wielded here are for exploring the key space to detect the best technique for global pollination that maintained a diversity of approaches. The golden selection was done to divide the search space into line segments as follows,

$$\nu = \frac{1}{\exists} \quad (39)$$

wherein, ν indicates the inverse golden ratio and the golden ratio $(\exists)$ was a constant value. After that, two separate search equations were obtained as,

$$\hat{fi}_1 = j + 0.618\,(w - j) \quad (40)$$

$$\hat{fi}_2 = w - 0.618\,(w - j) \quad (41)$$

In 40 and 41 equations, $\hat{fi}_1$ and $\hat{fi}_2$ are two solution vectors. These new intervals were applied for the subsequent iteration. In different evolution, the step size is computed by the difference between individuals in the current population that is given as,

$$\hat{fi}_0 = low + (up - low) * rand\ (R) \quad (42)$$

Here, the solution vector attained via the intensification process is represented as $\hat{fi}_0$. Limits are set up as lower limit *(low)* and upper limit *(up)* with a random value from the optimization problem *rand(R)*. Finally, the initial solution obtained was distributed evenly over the search space utilizing the following equation,

$$\hat{fi}_{t+1} = \hat{fi}_t + stp * \tan\theta * (\hat{fi}_t - g^*) \quad (43)$$

From this optimized solution *(Q(x))*, only one box gets selected for identifying the infected area. Further, the YoloV3 model processes the image as follows,

Step 1: Let the pre-processed image pixel *K(p, q)* be divided into *U*U* grids. The position information of (B_b) boundary boxes and their corresponding objectness score is computed by the grid and is represented as,

$$O_S^T = \mathrm{P}_{l,m}(Object) * IOU_{\mathrm{Pr}ed}^{Truth},\ P_{l.m}(Object) \in \{0,1\} \quad (44)$$

where, O_S^T signifies the objectness score of the T^{th} grid, $P_{l,m}(Object)$ denotes the object function of the m^{th} bounding box in the i^{th} grid and $IOU_{Truth\ Pred}$ signifies the intersection over the union between the predicted box and the ground truth box. Correspondingly, $P_{l,m}(Object)=1$ illustrates the presence of the object in the grid and otherwise 0. Intersection Over Union *(IOU)* is formulated as,

$$IOU = \frac{A_1 \cap A_2}{A_1 \cup A_2} \quad (45)$$

where, A_1 and A_2 denote the area of box 1 and box 2, correspondingly. Therefore, the objectness score not only signifies the presence of objects in the grid but also predicts the accuracy of the bounding box in the presence of the object.

Step 2: After that**,** the binary cross entropy (E_{BC}) is determined for the predicted objectness score and the ground truth objectness score. It is expressed as,

$$E_{BC} = \sum_{S=0}^{U^2} \sum_{T=0}^{G} V_{st}^{obj} \left[O_s^T \log \hat{(O_S^T)} - \left(1 - \hat{O_S^T}\right) \log \left(1 - O_S^T\right) \right] \quad (46)$$

where, U^2 illustrates the image's number of grids, G signifies the number of bounding boxes and V_{St}^{obj} is the entropy loss function. Moreover, O_S^T and $\hat{O}_s^T$ represent the predicted objectness score and truth objectness score, correspondingly.

Step 3: As this step was enhanced utilizing the W-GFPO optimization algorithm for selecting multiple targets at the same location and thereby the best bounding boxes using *(Q(x))* is selected. Then, $K(C_i, C_j)$ is modeled as,

$$K(C_i, C_j) = \tanh(\beta \cdot C_i^T C_j + Z) \quad (47)$$

where, β and Z show kernel parameter. Therefore, the objects detected in the image are signified as $K(p,q)$ where $T=1,2,3,..,L$ which signifies the number of objects detected. These segmented images were then labeled (L_i) grounded on their features. The Learning Leaky Relu function wielded as activation is given as,

$$LLReLu = f(l) = \begin{cases} l, & if\ l > 0 \\ \exp(l) * l^{-1}, & if\ l \le 0 - 1.22 \end{cases} \quad (48)$$

Here, l symbolizes the input's control learning point and $f(l)$ is the activation function. The term *exp* symbolizes exponential function. Grounded on features obtained, the images were labeled to diseases, namely coffee leaf miner, phoma tarda, coffee leaf rust, and Iron spot.

Pseudocode 1 for W-GPFO

Input: Input image and features
Output: Labeled bounding box of the infected area

Begin
Initialize GF population, $low, up, rand(R)$ maximum iteration t_{max}
Set Iteration $t = 1$
while $(t \le t_{max})$ **do**
calculate fitness fi
obtain best global solution (g^*)
if $(rand < \Gamma)$ {
Distribute flowers with Weibull distribution $(Q(x))$
Apply tangent flight algorithm
}**else** {
Perform Uniform distribution
Apply $\nu = \frac{1}{\exists}$
}
end If
Evaluate new flower fitness fi_{t+1}
if $(fi_{t+1} > fi_t)$ {
Update position by $fi_t + stp * \tan\theta * (fi_t - g^*)$
}**else** {
Discard flower
}
end If
Update current best solution
end While
$t = t + 1$

Return optimal solution

End

Clustering using J-FCM

Labeled data obtained *(L_i)* was further given for clustering with similar diseases utilizing J-FCM. Since the conventional Fuzzy C-Means Clustering (FCM) has low constraint properties, it has been selected here. Yet, it has a limitation when utilizing Euclidean distance of slow convergence. Thus, it is modified to follow the Jaccard Index-centric distance measurement. The J-FCM working process is described further,

- Here, *(L_i)* are given as input and the number of cluster centroids *(z_k)* is selected randomly. It is expressed as,

$$z_k = z_1, z_2, \ldots \ldots, z_y \tag{49}$$

- After that, the Jaccard distance $\left(d\langle T, z_k \rangle\right)$ between the initial pixels T and the cluster centroids *(z_k)* is computed using equation (49),

$$d = \sum_{k=1}^{T} \left\langle \sum_{k, x_e \in G} \|T - z_k\|^2 \right\rangle \tag{50}$$

where, G signifies the cluster and $G \geq 1$.

- The J-FCM's objective function (ϖ) is computed as,

$$\varpi = \sum_{k=1}^{T} \sum_{k=1}^{A} \left(m^{ek}\right)^A \cdot [d(T, z_k)]^2 \tag{51}$$

where, $1 \leq A \leq \infty$ specifies the real number greater than 1 that controls the degree of fuzziness, m^{jk} displays the membership function $m^{jk} \in [0,1]$. Further, it was modified as,

$$\left\| m^{jk}\left(I^t\right) - m^{jk}\left(I^t + 1\right) \right\| \leq \varepsilon \tag{52}$$

Here, ε is a constant in the range of 0 to 1, I^t indicates the number of iterations. Lastly, grounded on the distance, the objective function is given for further identification and analysis.

- Thus, the finally obtained clusters are given as,

$$\mathrm{H} = \{h_0, h_1, h_2, h_3, h_4\} \tag{53}$$

Here, H signifies the total clusters, h_0 is healthy leaf cluster, coffee leaf miner, phoma tarda, coffee leaf rust, and Iron spot are signified as h_1 h_2 h_3 and h_4, correspondingly. The clustering process finally clusters the images according to the label and if not found any label, then the image was clustered as a healthy leaf cluster.

Pseudocode 2 for G-YOLOv3

Input: Labelled image (L_i)
Output: clusters H

Begin

Initialize pixels T, I^t
For each labeled feature **do**
Determine (z_k)

Perform $\sum_{k=1}^{T} \left\langle \sum_{k, x_e \in G} \| T - z_k \|^2 \right\rangle$ with Jaccard distance

Update G until all pixels are grouped

Compute ϖ , m^{jk}

end For

Return clusters H

End

Results and Discussion

Here, the proposed approach's performance is experimentally verified in comparison with the conventional techniques. In the working platform of Python, the experiments are performed. The dataset utilized for experimental analysis is synthetically created. Figure 4 depicts the sample image results after the proposed preprocessing and segmentation.

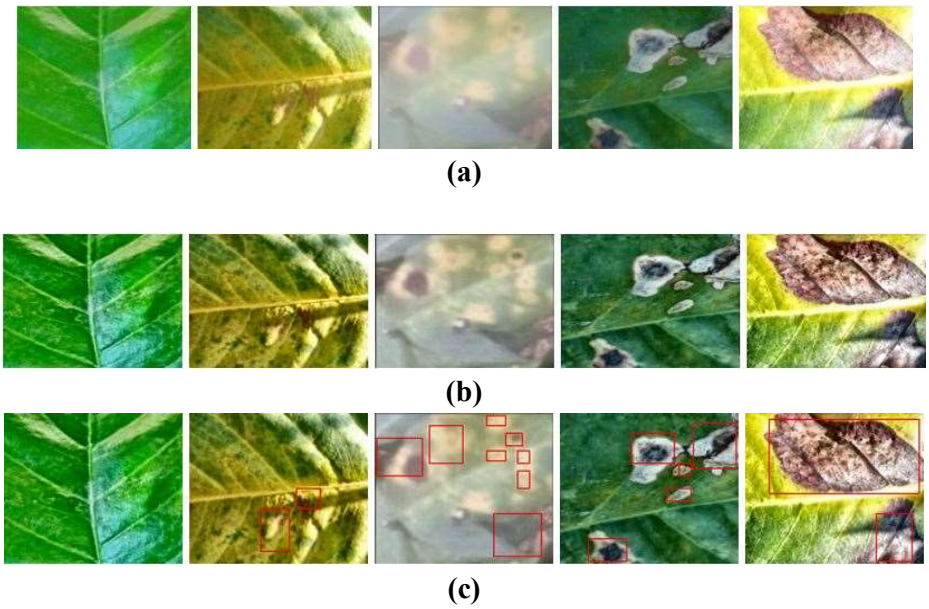

Figure 4. Sample results of (a) Input images (b) Preprocessed Images and (c) segmented images using the proposed approaches.

Performance analysis

Here, the outcomes are evaluated in three segments, namely edge enhancement, segmentation, and clustering.

Performance evaluation of edge enhancement

Here, the proposed edge enhancement technique PRUF's performance is analyzed in comparison with the RUF, UF, Wavelet Transform (WT) and Sobel approach regarding False Positive (FP), False Negative (FN) and error.

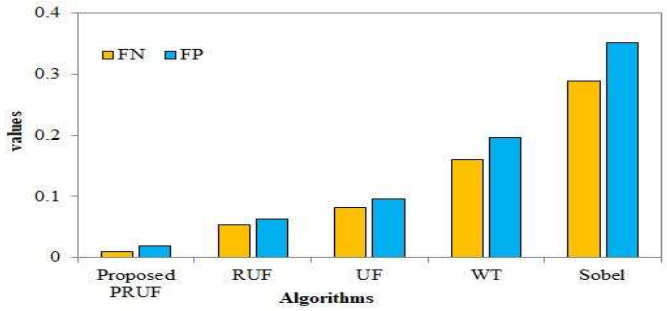

Figure 5. FP and FN results of enhanced images

Figure 5 represents that FP and FN value obtained by the RUF (0.0533, 0.0625) was better than the other prevailing algorithms during the experimental analysis. Yet, utilizing the Pansharpening in the RUF, the FN and FP are reduced to 0.0094 and 0.0186. Hence, for efficient edge enhancement in the proposed framework, the PRUF technique was utilized, which enhanced the edges by perfect detection of edges and non-edges.

Table 1: Error analysis

Algorithm	Error value
Proposed PRUF	0.1125
RUF	0.4691
UF	0.5664
WT	0.6842
Sobel	0.8150

The experimental error outcomes of the proposed PRUF and the conventional edge detection algorithms are displayed in Table 1. The error value is the error (false edges, missing edges, etcetera) detected during the edge enhancement. From the table, it is evident that the less error value of the PRUF (0.1125) technique makes it more suitable for reliable edge enhancement of leaves in the proposed technique.

Performance analysis of segmentation

This section discusses the Intersection Of Union (IOU), accuracy, sensitivity, specificity, Dice score, and Jaccard Index results of the proposed G-YOLOv3 segmentation in comparison with the conventional Grabcut-YOLOv3, YOLOv3, and watershed segmentation. Moreover, the proposed W-GPFO is analogized with the existing Ant Colony Optimization (ACO), GPFO, Particle Swarm Optimization (PSO), and GA concerning the Fitness vs iteration.

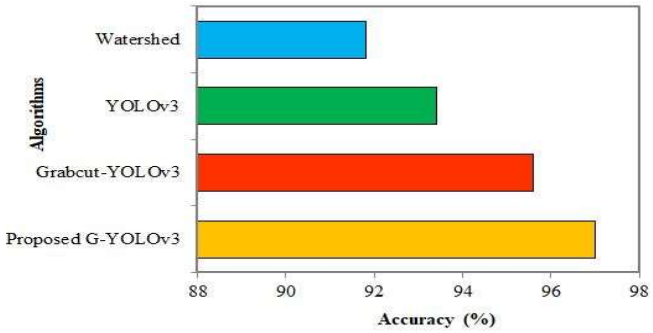

Figure 6. Accuracy level of the segmented image

Utilizing the accuracy metric, the accuracy of the segmented foreground and background pixels is assessed. The Grabcut-YOLOv3 segmented image is 2.35% and 4.13% more accurate than the YOLOv3 and watershed segmentation. Still, the proposed G-YOLOv3 overthrow the Grabcut-YOLOv3 by 1.46%, which is apparent from Figure 6 Thus, it is concluded that with G-YOLOv3 segmentation can be done more accurately than other techniques.

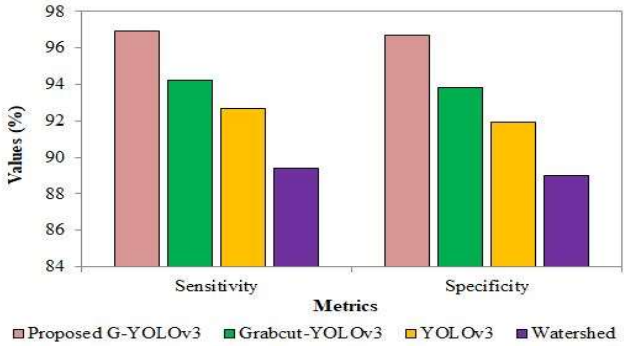

Figure 7. Performance analysis based on sensitivity and specificity

The experimental sensitivity and specificity results of the proposed and prevailing segmentation techniques are depicted in Figure 7. The perfectness of a system in segmenting foreground and background from the segmented image is determined by sensitivity and specificity. It is transparent that the watershed algorithm gives a poor performance of 89.4% and 89% for sensitivity and specificity, followed by the YOLOv3. However, the G-YOLOv3 obtained better outcomes during experimental analysis, which shows that with G-YOLOv3, the foreground as well as the background of the diseased coffee leaves are segmented more perfectly, which makes it more suitable for the proposed framework.

Table 2: Dice score and Jaccard index results

Algorithms	**Dice score (%)**	**Jaccard Index (%)**
Proposed G-YOLOv3	88.13	79.54

Grabcut-YOLOv3	86.81	77.27
YOLOv3	84.47	75.98
Watershed	81.9	72.84

The Jaccard index and dice score are the metrics utilized for evaluating the similarity level between predicted and ground truth segmented images. From Table 2, it is transparent that the YOLOv3 gives superior outcomes when analogized with the watershed algorithm, followed by Grabcut-YOLOv3 and G-YOLOv3. However, the G-YOLOv3 segmented the image more similarly to the ground truth, which is confirmed by achieving higher dice score (88.13%) and Jaccard index (79.54%) values than other segmentation techniques.

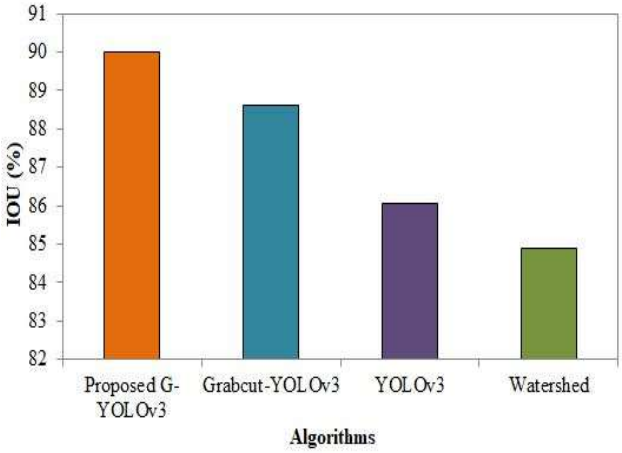

Figure 8. IOU analysis

IOU determines the degree of overlap between the segmented image by a system and ground truth. Figure 8 illustrates that during experimental analysis, Grabcut-YOLOv3 attained a better IOU degree (88.62%) in contrast to the other prevailing segmentations. However, the proposed G-YOLOv3 attained a 1.55% higher IOU degree than the Grabcut-YOLOv3, which shows the superiority of the G-YOLOv3 during segmentation.

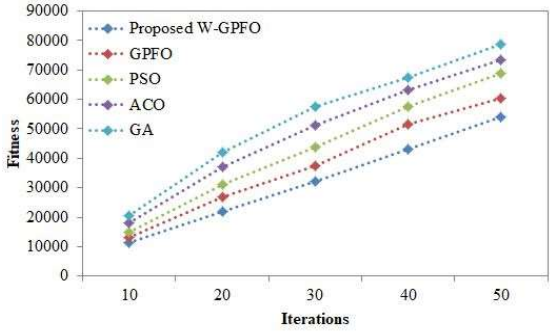

Figure 9. Fitness vs iteration

The fitness analysis determines how close the selected features are closer to the features of coffee leaf diseases. The fitness values for 10-50 iterations for the proposed W-GPFO and the conventional optimization algorithms are depicted graphically in Figure 9. Here, the

optimization algorithms give the fittest features at the 50th iteration. Still, the fitness of W-GPFO (54165) is higher than conventional optimization algorithms, which makes the W-GPFO more reliable to choose features of coffee leaf disease in the G-YOLOv3 segmentation.

Performance analysis of clustering

In this phase, the multi-disease classification of the coffee leaf utilizing the proposed JFCM clustering model experimental results is analyzed grounded on the accuracy, F1-macro, and F1-micro metrics.

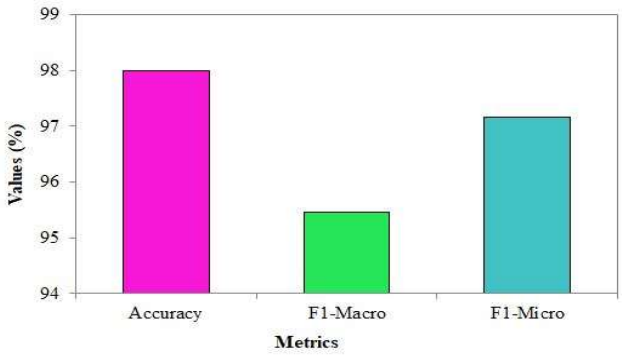

Figure 10. Analysis of proposed JFCM

From Figure 10, it is evident that the proposed JFCM technique attained a 98% accuracy level, 95.46% F1-Macro level, and 97.17% F1-Micro level during the coffee leaf multi-disease classification. The accuracy, F1-Macro, and F1-Micro are the metrics wielded for evaluating the classified outcomes' quality. The proposed framework achieved a higher accuracy level, which is nearly close to 100%, which depicts the efficacy of the JFCM for classification in the proposed framework. Hence, it is proved that the multiple diseases in the coffee leaf are classified more accurately in the proposed mechanism.

Comparative analysis

Here, the proposed WGPFO-JFCM-based Multi-disease Prediction of Coffee Leaves (WJMPCL) framework is analogized with the related works Gaussian Mixture Model (GMM) [12], DenseNet121 [3] and Ensemble [14] concerning accuracy.

Table 3: Comparative results based on the accuracy

Algorithms	**Accuracy (%)**
Proposed WJMPCL	98
GMM (Hasan et al., 2022)	90

DenseNet121 (Binney & Ren, 2022)	95.44
Ensemble (Novtahaning et al., 2022)	97.3

The comparative assessment of the proposed along with the prevailing works whose objective is the multi-disease prediction in coffee leaves is illustrated in Table 3, Although the objective of the proposed and the GMM, DenseNet121, ensemble model are the same, still the WJMPCL predicts the multi-disease in coffee leaves 8.88%, 2.68%, and 0.71% more accurately than the prevailing GMM, DenseNet121, and ensemble models. This comparative analysis proves the dominance of the WJMPCL framework over other models for multi-disease prediction in coffee leaves.

Conclusion

This paper proposed a JFCM-centric multi-disease prediction in coffee leaves with a reliable G-YOLOv3 segmentation. Here, for image enhancement, the PRUF technique is utilized; also, for the coffee leaf disease prediction, JFCM is wielded. A synthetically generated dataset is utilized for the analysis of the proposed approaches' performance. In the experimental analysis, the proposed PRUF attained a less error of 0.1125 during image enhancement. Moreover, the images are segmented with a higher accuracy level of 97%, dice score of 86.81%, Jaccard index of 79.54%, and IOU degree of 90%. In the comparative analysis, the prediction of multiple diseases in the coffee leaves attained a 98% accuracy level, which is higher than the prevailing works. These evaluations proved the proposed WJMPCL model's superiority over other models for multi-disease prediction in coffee leaves. This system concentrated on a multi-disease prediction, but the fungus, which causes the disease in coffee leaves, is not considered. Thus, fungus prediction with other diseases in the coffee leaves will be predicted in the future.

References

[1] Bordin, Y., & Leite, P. R. R. (2023). Smart Agricultural Technology Coffee disease classification at the edge using deep learning. *Smart Agricultural Technology*, *4*, 1–9. https://doi.org/10.1016/j.atech.2023.100183

[2] Abd Algani, Y. M., Marquez Caro, O. J., Robladillo Bravo, L. M., Kaur, C., Al Ansari, M. S., & Kiran Bala, B. (2023). Leaf disease identification and classification using optimized deep learning. *Measurement: Sensors*, *25*, 1–6. https://doi.org/10.1016/j.measen.2022.100643

[3] Binney, E., & Ren, D. (2022). Coffee Leaf Diseases Classification And The Effect Of Fine-Tuning On Deep Convolutional Neural Networks. *International Journal for Multidispliinary Research*, *4*(5), 1–13.

[4] Divyanth, L. G., Ahmad, A., & Saraswat, D. (2023). A two-stage deep-learning based segmentation model for crop disease quantification based on corn field imagery. *Smart*

Agricultural Technology, *3*, 1–12. https://doi.org/10.1016/j.atech.2022.100108

[5] Saeed, A., Mossad, A., Abdelhamid, M. A., Alkhaled, A. Y., & Mayhoub, M. (2023). Smart Detection of Tomato Leaf Diseases Using Transfer Learning-Based Convolutional Neural Networks. *Agriculture*, *13*, 1–14.

[6] Tritsch, N., Steger, M. C., Segatz, V., Blumenthal, P., Rigling, M., Schwarz, S., Zhang, Y., Franke, H., & Lachenmeier, D. W. (2022). Risk Assessment of Caffeine and Epigallocatechin Gallate in Coffee Leaf Tea. Foods, 11, 1–20.

[7] Koshariya, A., Rout, S., Venu, N., & Gwalior, S. (2022). An Enhanced Machine Learning Approach For Identifying Paddy Crop Blast Disease Management Using Fuzzy Logic. Ifans International Journal of Food and Nutritional Sciences, 1152–1163.

[8] Mao, R., Wang, Z., Li, F., Zhou, J., Chen, Y., & Hu, X. (2023). GSEYOLOX-s : An Improved Lightweight Network for Identifying the Severity of Wheat Fusarium Head Blight. Agronomy, 1–14.

[9] Gurung, K., Pandey, A., Bamaniya, B. S., Dasila, K., Sharma, L., & Bag, N. (2021). Leaf Blight Caused by Colletotrichum Fructicola of Large Cardamom (Amomum Subulatam Roxb .) an Important Cash Crop Grown in Sikkim , India. Europe PMC, 1–21.

[10] Esgario, J. G. M., de Castro, P. B. C., Tassis, L. M., & Krohling, R. A. (2022). An app to assist farmers in the identification of diseases and pests of coffee leaves using deep learning. Information Processing in Agriculture, 9(1), 38–47. https://doi.org/10.1016/j.inpa.2021.01.004

[11] De Resende, M. L. V., Pozza, E. A., Reichel, T., & Botelho, D. M. S. (2021). Strategies for coffee leaf rust management in organic crop systems. Agronomy, 11(9), 1–14. https://doi.org/10.3390/agronomy11091865

[12] Hasan, R. I., Yusuf, S. M., Mohd Rahim, M. S., & Alzubaidi, L. (2022). Automated masks generation for coffee and apple leaf infected with single or multiple diseases-based color analysis approaches. Informatics in Medicine Unlocked, 28, 1–8. https://doi.org/10.1016/j.imu.2021.100837

[13] Tassis, L. M., & Krohling, R. A. (2022). Few-shot learning for biotic stress classification of coffee leaves. Artificial Intelligence in Agriculture, 6, 55–67. https://doi.org/10.1016/j.aiia.2022.04.001

[14] Novtahaning, D., Shah, H. A., & Kang, J.-M. (2022). Deep Learning Ensemble-Based Automated and High-Performing Recognition of Coffee Leaf Disease. Agriculture, 12(11), 1909. https://doi.org/10.3390/agriculture12111909

[15] Santos, L. M. Dos, Ferraz, G. A. E. S., Marin, D. B., Carvalho, M. A. de F., Dias, J. E. L., Alecrim, A. de O., & Silva, M. de L. O. e. (2022). Vegetation Indices Applied to Suborbital Multispectral Images of Healthy Coffee and Coffee Infested with Coffee Leaf Miner. AgriEngineering, 4(1), 311–319. https://doi.org/10.3390/agriengineering4010021

[16] Montalbo, F. J. P., & Hernandez, A. A. (2020a). An Optimized Classification Model for Coffea Liberica Disease using Deep Convolutional Neural Networks. Proceedings - 2020 16th IEEE International Colloquium on Signal Processing and Its Applications, CSPA 2020, 213–218. https://doi.org/10.1109/CSPA48992.2020.9068683

[17] Montalbo, F. J. P., & Hernandez, A. A. (2020b). Classifying barako coffee leaf diseases using deep convolutional models. International Journal of Advances in Intelligent

Informatics, 6(2), 197–209. https://doi.org/10.26555/ijain.v6i2.495

[18] Kumar, M., Gupta, P., Madhav, P., & Sachin. (2020). Disease detection in coffee plants using convolutional neural network. Proceedings of the 5th International Conference on Communication and Electronics Systems, ICCES 2020, Icces, 755–760. https://doi.org/10.1109/ICCES48766.2020.09138000

[19] Suparyanto, T., Firmansyah, E., Wawan Cenggoro, T., Sudigyo, D., & Pardamean, B. (2022). Detecting Hemileia vastatrix using Vision AI as Supporting to Food Security for Smallholder Coffee Commodities. IOP Conference Series: Earth and Environmental Science, 998(1), 1–9. https://doi.org/10.1088/1755-1315/998/1/012044

[20] Marcos, A. P., Rodovalho, N. L. S., & Backes, A. R. (2019). Coffee Leaf Rust Detection Using Genetic Algorithm. Proceedings - 15th Workshop of Computer Vision, WVC 2019, 16–20. https://doi.org/10.1109/WVC.2019.8876934

[21] Manso, G. L., Knidel, H., Krohling, R. A., & Ventura, J. A. (2019). A smartphone application to detection and classification of coffee leaf miner and coffee leaf rust. Journal of Latex Templates, 1–36.

[22] Junior, J. S. Z., Costa, H., Dorzenoni, R. R., Guarçoni, R. C., Fornazier, M. L., Sossai, S. R., Botacim, L. A., Souza, E. M. R., Ferrão, M. A. G., Martins, D. dos S., Favarato, L. F., & Fornazier, M. J. (2022). Modeling the brown eye spot sampling in Arabica coffee. International Journal of Advanced Engineering Research and Science, 9(4), 226–231. https://doi.org/10.22161/ijaers.94.26

[23] Ravi, V., Acharya, V., & Pham, T. D. (2022). Attention deep learning-based large-scale learning classifier for Cassava leaf disease classification. Expert Systems, 39(2), 1–24. https://doi.org/10.1111/exsy.12862

[24] Saveedra, L. M., Caixeta, E. T., Barka, G. D., Borem, A., Zambolim, L., Nascimento, M., Cruz, D. C., Oliveria, A. carlos B. de, & Pereria, A. antonia. (2023). Marker-Assisted Recurrent Selection for Pyramiding Leaf Rust and Coffee Berry Disease Resistance Alleles in Coffea arabica L. Genes, 14, 1–18.

[25] Santhosh Kumar S, Raghavendra BK (2021) A Study on Core Challenges in Coffee Plant Leave Disease Segmentation and Identification on Various Factors. Proceedings 3rd International Conference on Data Science, Machine Learning and Applications (ICDSMLA), Springer 433–446. https://doi.org/10.1007/978-981-19-5936-3_41

[26] Santhosh Kumar S, Raghavendra BK (2019) diseases detection of various plant leaf using image processing techniques: a review. In: 2019 5th International conference on advanced computing and communication systems (ICACCS) . Doi: 10.1109/Icaccs.2019.8728325