A Deep Learning Approach for Coffee Leaf Disease Classification with Centroid-Based Loss Optimization

Changyu AO, Gwang-Jun KIM, Man-Sung KWAN*

Abstract: As global demand for coffee increases, coffee trees, particularly their leaves, suffer more from diseases and pests. Leaf health, being the central organ for photosynthesis, directly impacts the yield and quality of coffee. Therefore, developing a computationally efficient and scalable method for disease and pest management in coffee trees has become imperative. This study introduces a centroid-based loss adjustment method to enhance the focus on distant samples from their class centroid. It involves calculating the centroid for each class and adjusting the loss according to the distance between the sample features and their class centroid. This method is essential concerning the spatial distribution of features, which helps improve classification accuracy and model robustness when dealing with complex data. The experiments use the BRACOL and RoCoLe datasets containing images of multiple coffee leaf diseases. Multiple Deep Learning pre-trained models were tested to validate the proposed method's effectiveness. The experimental results show that this method significantly enhances the accuracy of coffee leaf disease classification and surpasses State-of-The-Art methods. This study presents a method to classify and estimate coffee leaf disease severity, improving model performance without greatly increasing the computational burden.

Keywords: coffee leaf diseases; convolutional neural network; deep learning; loss adjustment

1 INTRODUCTION

In recent years, global demand for coffee has increased, fuelled by the widespread appreciation for its unique flavor and refreshing effects [1]. Coffee growers worldwide have significantly increased their plantations to satisfy the escalating market demand. However, with the expansion of cultivation, coffee leaves face growing problems with diseases and pests, especially their leaves [2]. Coffee leaves are the principal organs of the plant [3] for photosynthesis. Once damaged, the yield and quality of coffee are directly affected. Therefore, how to effectively control coffee leaf diseases and pests has become an urgent problem.

Traditional pesticides [4] can control and prevent diseases and pests of coffee leaves. However, extensive and prolonged pesticide use causes severe environmental pollution [5], increases pest resistance, and ultimately reduces the effectiveness of treatments [6]. These ecological consequences highlight the urgent need for precise disease identification. Conventional AI methods, however, face critical limitations in real-world deployment: their inability to distinguish overlapping symptoms (e.g., Rust vs. Cercospora lesions), susceptibility to lighting variations and occlusions in field images, and degraded performance when analyzing gradual severity changes within disease classes. As a result, it is crucial to develop computationally efficient and scalable methods for controlling pests and diseases in coffee leaves. In this regard, timely and accurate identification of biotic stresses in coffee leaves, followed by targeted control measures [7], is the best option.

To address these limitations, Machine Learning (ML), a key area of AI, is now applied in various fields, including agriculture [8]. ML offers a new solution for detecting and controlling diseases and pests in coffee leaves. By analyzing a large amount of agricultural data, ML models learn various features of diseases and pests, thus enabling their fast and accurate recognition [9]. Further, Deep Learning (DL), an ML subfield, processes more complex data and provides more accurate predictions by modeling

how the human brain works [10]. DL has proven highly effective in image recognition tasks [11], a capability particularly useful for identifying diseases and pests by analysing coffee leaf images. Nevertheless, DL methods still face persistent challenges in coffee leaf disease management [12], particularly in discriminating subtle morphological variations under field conditions (e.g., early-stage Rust vs. nutrient deficiencies), maintaining robustness across heterogeneous symptom distributions, and balancing computational complexity with real-time deployment needs.

Deep features extracted from DL models have shown their irreplaceable significance in many areas of DL, especially in classifying coffee leaf diseases [13]. Improve classification accuracy by fine-grained features of coffee leaves, such as the structure of leaf veins, leaf texture, and the morphology of lesions, to guide the model to focus more on crucial information [14]. This method not only focuses on the features extracted from the whole leaf but also pays more attention to the deeper features specific to coffee leaves, ensuring the accuracy and robustness of the model when dealing with complex leaves.

The efficiency of coffee leaf disease management has been significantly improved [15] by applying DL. Not only does this method quickly and accurately identify disease-affected leaves, enabling farmers to respond promptly to reduce disease spread and losses, but it also helps reduce pesticide use [16]. Controlling coffee leaf diseases is crucial to enhancing the coffee industry's sustainability [17]. As the global demand for coffee grows, improving the health and productivity of coffee trees becomes vital, meeting the market demand and ensuring the long-term sustainability of the coffee industry [18]. Applying DL in coffee leaf disease management improves efficiency and accuracy and helps realize environmentally friendly and sustainable agricultural production methods. In addition, DL helps drive technological innovation across the agricultural sector, providing intelligent solutions to a broader range of agricultural problems.

In summary, recognizing the crucial role of deep features in coffee leaf classification, this paper proposes an

innovative solution: a centroid-based loss adjustment method. Without changing the backbone, this method calculates the centroid for each class. It adjusts the loss based on the distance between coffee leaf features and their respective centroids, ensuring that the model gives more attention to samples further from their category centroid. Using validation trials on the BRACOL and RoCoLe datasets, the centroid-based loss adjustment method was applied to various baseline models, enhancing their performance.

The structure of this paper is outlined as follows: Section 2 reviews existing research. Section 3 describes the dataset and research methodology, including the proposed method. Section 4 presents experimental results and analyses. Section 5 discusses and compares this method with existing ones. Finally, Section 6 concludes the study and suggests future research directions.

2 LITERATURE REVIEW

Recent studies have applied various DL methods to classify coffee leaf disease and estimate severity. The study by Kumar et al. [19] used Convolutional Neural Networks (CNN) [20] to detect coffee plant diseases. They added a flattened layer and a dense layer at the back of the model end to enhance its ability to handle image recognition tasks. Transfer learning and data augmentation techniques were used to improve generalization and recognition accuracy, achieving up to 97.61%. However, the methodology of this study requires refinement in assessing disease severity, likely due to the limited capability of the model to differentiate subtle disease variations. The study by Lisboa et al. [21] modified the ResNet50 [22] model by adding three dense layers, allowing it to identify issues in Brazilian Arabica coffee leaves and categorize them into diseases and pests. Their method highlights deep neural networks' ability to recognize complex agricultural diseases. Binney and Ren [23] trained DL models (including ResNet50, Densenet-121 [24], and VGG19 [25]), fine-tuned these models to evaluate specific leaf diseases and added dense layers to the models' back end. The studies by Kumar, Lisboa, and Binney added dense layers to the back end of their respective network structures to improve model accuracy. However, adding extra layers implies increased model complexity and computational requirements. Extra layer stacking leads to model overfitting, especially when the data is insufficient or low quality. At the same time, as the model becomes complex, it becomes more challenging to adjust and optimize the model parameters, which limits the flexibility and applicability of the model. These studies have significantly enhanced coffee leaf disease recognition. They face computational resources and model management in real-world applications.

The study by Karthik et al. [26] developed global contextual attention based on the Inception architecture, which combined the Inception module [27], the global context block [28], and multi-head attention [29] to improve performance in classifying coffee leaf diseases. Okada et al. [30], on the other hand, proposed a DL structure for coffee leaf disease severity classification, focusing on capturing the distribution of discoloration and emphasizing features like color by introducing an attentional mechanism in the neural network's shallow layer. This method effectively supports the discrimination of disease severity and improves recognition accuracy. These two studies added standard attention mechanisms to enhance the accuracy of the model. However, these intentional mechanisms, focusing mainly on immediate, surface-level features, need help comprehensively analysing the image. This lead the model to overemphasize essential elements like color and texture while neglecting more complex, abstract features crucial for accurate classification. The deep features obtained from feature extraction contain abstract information about the sample, such as specific shapes, parts of the object, or the whole object. Research in image recognition [31] and natural language processing [32] has focused on transforming deep features into training parameters, optimization conditions, or decision criteria to enhance model focus and adaptability. The use of deep features to adjust the loss function not only focuses on global error minimization but also makes detailed adjustments to the specific features of the samples.

With the proven effectiveness of DL for coffee leaf disease detection, several studies have developed smartphone applications for various diseases. These applications leverage the power of deep learning to provide real-time, accurate detection and classification, making it easier for farmers to manage and treat coffee leaf diseases promptly. The study by Manso et al. [33] created an app for detecting and categorizing coffee leaf miners and rust. This app incorporates DL techniques to highlight mobile device portability and ease of use in agricultural disease management. Their work focuses on providing a cost-effective solution to help farmers promptly identify specific coffee leaf diseases. In a separate study, Pinto et al. [34] expanded this area further by developing an application capable of recognizing five different coffee leaf disease traits, including brown eye spots and coffee leaf blight. This study demonstrates further innovation and development in coffee leaf disease identification by increasing the range of recognizable traits and providing users with a more comprehensive disease management tool. Although these applications offer portability and user-friendly interfaces for managing coffee leaf diseases, their development has primarily relied on widely used baseline models like CNNs, which still require optimization for improved accuracy and other factors such as processing speed and adaptability.

Given the challenges and limitations encountered in real-world application scenarios, several studies have focused on optimizing various aspects of DL models, including model complexity and generalization capabilities. Ramamurthy et al. [35] proposed a unique architecture by combining EfficientNetB0 [36] and Ghost modules [37], effectively reducing the model's computational demands while maintaining a relatively high classification accuracy. Although there are limitations in the recognition accuracy of specific categories, this method provides a cost-effective solution for automated disease detection. In contrast, the study by Wang et al. [38] designed a lightweight model utilizing transfer learning and knowledge distillation [39]. This "student network" learns critical information from a complex "teacher network", enabling efficient disease identification and classification. A common challenge in both studies is maintaining or improving the recognition

ability and accuracy while reducing computational requirements. Strategies to reduce computational complexity often compromise the model's capacity to recognize subtle features, especially in coffee leaf disease images with rich details and variability.
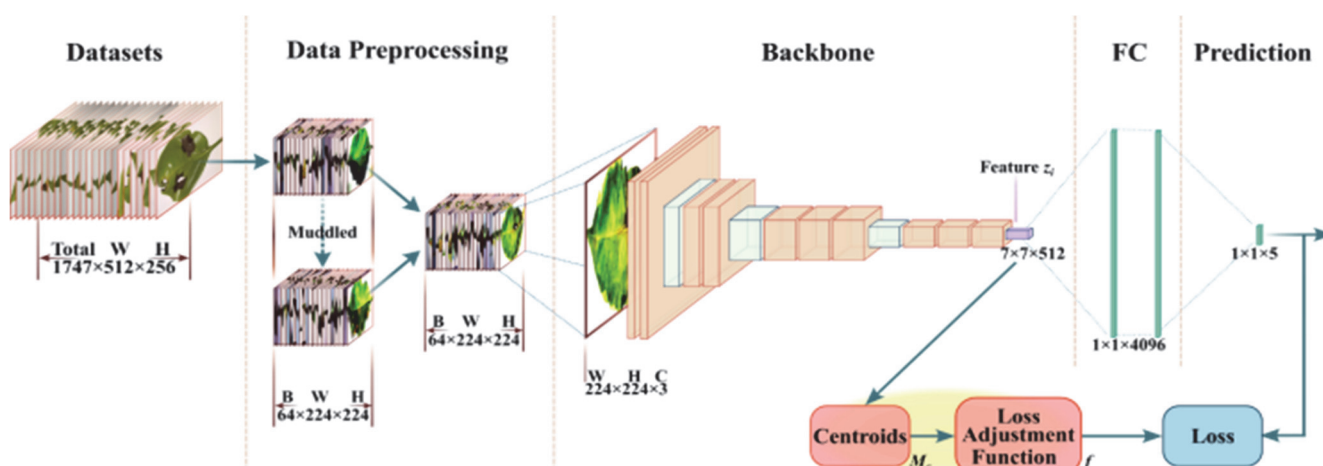

Figure 1 Proposed structure

Esgario et al. [40] investigated the use of Multi-task Learning (MTL) [41, 42] to simultaneously process symptom recognition and severity estimation in coffee leaves. This method attempted to solve multiple tasks in one DL model but did not significantly outperform traditional methods focused on a single task. Although MTL theoretically enhances the model's generalization, the complexity of feature extraction limits its effectiveness under a shared architecture. This work reveals that traditional methods focusing on specific problems still have advantages in complex agricultural disease classification tasks. Tassis and Krohling [43] employed TripletNet [44], which enforces inter-class separation using anchor-positive-negative triplets, but this approach struggles with overlapping symptoms in coffee leaves. ProtoNet [45] avoids explicit triplet curation by comparing features to fixed centroids, but it forces discrete severity assignments, failing to represent the continuous nature of disease progression. These challenges are also evident in other loss functions: Center Loss [46] compresses intra-class variations effectively but relies on static centroids that cannot adapt to evolving symptoms, and contrastive learning [47] suffers from computational bottlenecks due to exhaustive pairwise comparisons, especially with high-resolution leaf images. In contrast, our approach addresses these issues with dynamic centroid adaptation and nonlinear severity-sensitive scaling. Unlike TripletNet, which depends on manually selected anchors, our method automatically adjusts feature distances via evolving centroid distributions. Moreover, our dynamic centroids track lesion progression patterns more effectively than ProtoNet's rigid prototypes or Center Loss's static targets. Additionally, our sinusoidal loss modulation emphasizes borderline cases by amplifying their gradient contributions, offering a more biologically accurate representation of symptom severity than contrastive learning's uniform weighting.

3 METHODOLOGY

This chapter outlines the research framework and critical techniques used for implementation. Fig. 1 displays a flowchart outlining the proposed research methodology, including the sequence of steps in the DL method. First, the original coffee leaf disease images were subjected to a series of basic preprocessing steps, including resizing, flipping, rotating, and color dithering, to ensure they were suitable for subsequent DL processing. Next, the basic preprocessed image set was copied and muddled to implement more advanced data enhancement techniques. This step mixes this muddled set with the basic processed set to generate new training samples. Using the Between-ClassPlus (BC+) [48] method, this mixing enhances the dataset's diversity and the model's generalization by combining images from different classes. Then, the mixed images enter the model's backbone network for feature extraction. Multiple DL models, pre-trained on the ImageNet dataset, are compared for their efficiency in extracting key features from coffee leaf images. These features are utilized both in the fully connected layer (FC) for disease classification and in the proposed centroid-based loss adjustment method. The loss function was adjusted by calculating the centroid for each class and focusing on the distance between sample features and their class centroid. This adjustment made the model pay more attention to samples that were farther from their class centroid. The subsections delve into each key component of the research methodology separately.

3.1 Datasets

This study utilized two well-recognized and representative datasets, the BRACOL [40] and RoCoLe [49] datasets in Fig. 2, to bolster the credibility of the findings. These datasets are widely recognized as standard benchmarks in coffee leaf disease research due to their rich samples and detailed classification. Below is a detailed description of these two datasets.

BRACOL dataset consists of 1,747 images of Arabica coffee leaves; 1,685 were retained after removing exception labels. These images have white backgrounds and were taken by different smartphones. The leaf dataset and the symptom dataset are the two sub-datasets contained in the BRACOL dataset.

The leaf dataset consists of images of entire leaves, categorized into five classes: Healthy, Rust, Leaf Miner, Brown, and Cercospora Leaf Spot, with corresponding quantities of 272, 531, 387, 348, and 147. It is also classified into five severity levels: Very High, High, Low, Very Low, and Healthy, with corresponding quantities of 56, 101, 332, 924, and 272. Therefore, Biotic Stress and Severity are the two classification tasks contained in the Leaf dataset.

The symptom dataset consists of 2,722 images of independent symptoms cropped from the original images. These images are categorized into the same five classes as the leaf dataset, with corresponding quantities of 256, 593, 991, 504, and 378.

RoCoLe Dataset comprises 1,560 images of Robusta coffee leaves, all captured using a 5-MP smartphone in the same field as natural coffee plants. It labels both symptoms and severity, including Healthy, Red Spider Mite, and Rust Levels 1 to 4, with corresponding quantities of 791, 167, 344, 166, 62, and 30.
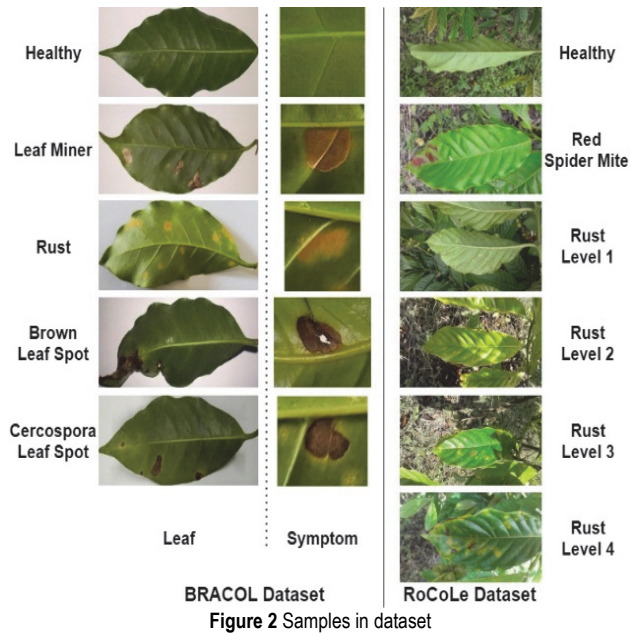

Figure 2 Samples in dataset

3.2 Data Preprocessing

This study employs various primary data preprocessing methods to improve the generalization and adaptability of the model for real application scenarios. These techniques include random flips (horizontal and vertical) with a 50% probability of simulating different viewing angles and orientations. Then, rotations of up to 10 degrees with 25% probability were randomly applied, which was done to increase the visual diversity of the samples. To further enrich the image data, a color dithering process was introduced, adjusting the brightness, contrast, and saturation of the image with a 25% probability to increase data variability. These preprocessing steps help the model to better understand and process coffee leaf images from different angles and lighting conditions. To further enhance the generalization of the model, we apply a data augmentation method: BC+. BC+ creates new training samples by randomly mixing samples from different classes. The formula for its image mixing is as

$$X_{\text{mixed}} = \frac{p_i(X_i - \mu_i) + (1 - p_i)(X_i' - \mu_i')}{\sqrt{p_i^2 + (1 - p_i^2)}},$$

$$\text{where } p_i = \frac{1}{1 + \frac{\sigma_i}{\sigma_i'} \cdot \frac{1 - r_i}{r_i}}. \tag{1}$$

Input batch X is used to calculate the mean μ_i and standard deviation σ_i for each image, with X' randomized as a target for mixing with the original sequence. The mixing ratio r is a random float in the interval $[0, 1)$ used to compute the mixing parameter p_i. Similarly, the mixing ratio r is used to compute the mixing label, and the formula is as

$$Y_{\text{mixed}} = r_i \cdot Y_i + (1 - r_i) \cdot Y_i' \tag{2}$$

Image mixing is more than simply fusing two images; instead, it focuses on maintaining the recognisability of lesion features during the generation of samples. In addition, coffee leaf disease often manifests itself in diverse symptoms. With BC+, researchers can generate a set of samples containing multiple combinations of symptoms, providing a rich context for model learning and extracting disease features. BC+ enables models to better adapt to the variability and complexity of real-world leaf diseases.
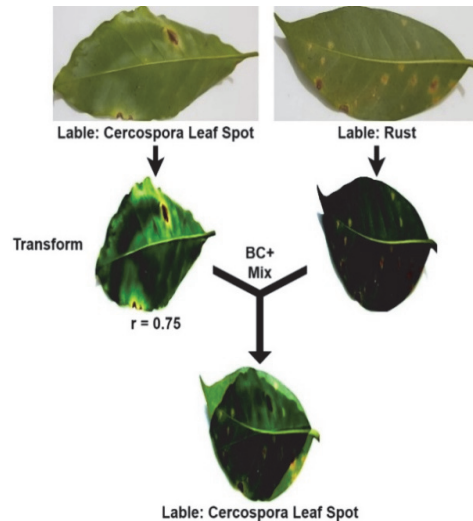

Figure 3 Example of image preprocessing

In combination with these augmentation techniques, BC+ helps to address the inherent variability of coffee leaf disease symptoms by generating diverse new samples that enrich the training dataset. While traditional data augmentation techniques increase the diversity of the dataset by introducing visual variations, BC+ enhances this by generating synthetic samples that combine features from different classes. This fusion of augmented data and BC+ allows the model to better generalize across a wider range of symptoms and helps mitigate class imbalance by providing more balanced representations of underrepresented classes. Together, these methods

improve the model's robustness and ability to classify various stages of disease manifestations. Fig. 3 provides an example of image preprocessing that demonstrates the effectiveness of the BC+ when augmenting the dataset.

3.3 Backbone

CNNs perform well in image recognition and classification tasks, especially in coffee leaf disease classification. To accurately recognize and classify diseases on coffee leaves, this study compares various popular CNN models, including AlexNet [50], GoogLeNet [27], VGG16 [25], ResNet50 [22], MobileNetV2 [51], InceptionV3 [52] EfficientNetV2_s [53], and MobileNetV4_s [54]. The parameters and number of layers for each network are listed in Tab. 1.

Table 1 Overview of CNN model properties in this study

Backbone	Layers	Parameters (M)
AlexNet	8	61.09
GoogLeNet	22	6.99
VGG16	16	138.34
ResNet50	50	25.56
MobileNetV2	53	3.40
InceptionV3	48	23.90
EfficientNetV2_s	42	20.18
MobileNetV4_s	47	1.27

These models represent a significant development in DL for image recognition and have been used on multiple benchmark datasets to demonstrate their effectiveness. These models are pre-trained on the ImageNet dataset to extract rich information from photos containing millions of images across numerous categories. Therefore, this study utilizes these pre-trained models as a starting state to train directly on the coffee leaf disease datasets. These models learn disease-specific features in images at a finer granularity.

3.4 Loss Function

This study selected Cross Entropy Loss as the primary loss function for training. Cross entropy is a standard method for evaluating inconsistency between the model's predicted probability distribution and the actual distribution in multi-class classification problems. The Cross Entropy Loss L_i for a single sample is defined as

$$L_i = -\sum_{c=0}^{M} y_{i,c} \cdot \log\left(\hat{y}_{i,c}\right) \tag{3}$$

where, M represents the total number of classes. y is a one-hot encoded vector representing the ground truth (GT). $\hat{y}$ is the probability distribution output by the model after transformation by the softmax function. In this expression, $y_{i,c}$ is the element corresponding to class c in the GT vector, while $\hat{y}_{i,c}$ represents the probability of the model predicting that sample o belongs to class c. For each batch of training, the sum of the loss functions is the average of batch sample losses, as follows

$$Loss = \frac{1}{N}\sum_{i=0}^{N} L_i \tag{4}$$

where N is the batch size; this form of loss function drives the optimization process to raise the predicted probability of the actual label while lowering the predicted probability of incorrect labels, thereby enhancing overall classification performance. Minimizing this loss through Stochastic Gradient Descent (SGD) adjusts model parameters to reduce prediction error. This method helps to ensure that the model accurately predicts when exposed to new data, which is a core objective in classifying coffee leaf disease.

3.5 Centroid-Based Loss Adjustment

This research proposes a centroid-based loss adjustment method to improve the accuracy of coffee leaf disease classification, illustrated in Fig. 4. After feature extraction, the method computes the class centroids and adjusts the loss based on the distance to these centroids. This method accounts for sample distribution in feature space and assigns greater weight to those farther from the class centroid. Therefore, the model pays more attention to those samples that are more challenging to classify.
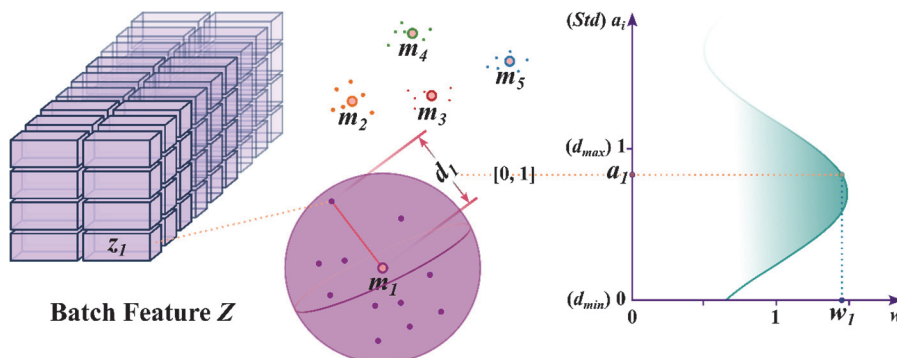

Figure 4 Centroid-based loss adjustment method

Let $Z = \{z_1, z_2, ..., z_N\}$ be the set of features corresponding to the batch size N. Let $L = \{l_1, l_2, ..., l_N\}$ be the set of labels associated with these features. Define the centroid function as:

$$m_c = \frac{1}{|N_c|}\sum_{i=0}^{N_c} z_i \tag{5}$$

The centroid of class within a batch is denoted m_c. N_c is the index set corresponding to class c, i.e., $N_c = \{ i \mid l_i = c \}$. $|N_c|$ is the number of features related to class c. The feature vector z_i is extracted from the sample x_i.

After computing the centroid m_c for each class, the Euclidean distance d_i between each feature vector z_i and the centroid of the corresponding class is calculated. This step is intended to assess the localization of each sample for its class. For each feature vector z_i, its distance from the corresponding category centroid m_c is defined by

$$a_i = \frac{d_i - d_{\min}}{d_{\max} - d_{\min}}, \text{ where } d_i = z_i - m_c \tag{6}$$

In this process, the distance d_i is calculated for all feature vector z_i in the batch, and each distance d_i is normalized to $[0,1]$, i.e., the normalized distance a_i, by the maximum value $d_{\max}$ and minimum value $d_{\min}$. Although using the normalized distance a_i as a weight adjustment factor is a direct method, it requires refinement to avoid issues like gradient explosion or disappearance, which adversely impact model performance during training. When the normalized distance a_i is directly used as weights, the loss function is too sensitive to the sample position, especially when the sample is very close or far from the centroid. Therefore, the normalized distance a_i is mapped into the sine function to calculate the weights w_i as

$$w_i = A \cdot \sin\left((a_i - p)\pi \right) + b \tag{7}$$

The hyperparameters amplitude A, phase p, and baseline b adjust the sine function to ensure that the calculated weights w_i are in a stable range [0.6464, 1.5]. Suitable values to achieve this are $A = 0.5$, $p = 0.25$, $b = 1$. This setup avoids situations where the sample weights are scaled up or down too much, especially for samples close to or far from the class centroids. The output of the sine function is stable when $a_i > 0.5$, which makes the weight change of the boundary sample more stable. This stationary reduces the gradient fluctuations during training. Models to better learn the features of the category boundary regions. The selection of these parameters is based on weight adjustment stability considerations, not specific to the BRACOL and RoCoLe datasets. For other datasets, these parameters can be used as initial values and further tuned. Ultimately, these calculated weights w_i are applied to the loss function for the corresponding samples. For the final loss function, we let

$$f(w_i, L_i) = \frac{1}{N} \sum_{i=0}^{N} w_i \cdot L_i \tag{8}$$

This method allows samples far away from the centroid of their class (i.e., samples located at the category boundary) to receive greater attention during the training process, which helps optimize the overall classification performance of the model.

4 EXPERIMENTS AND RESULTS

4.1 Experimental Setup

In each experiment of this study, the raw image data was resized to ensure consistency of input features and computational efficiency. The images were resized to 224 × 224 pixels. For the InceptionV3 experiments, they were scaled to 299 × 299 pixels to fit the model's input. The dataset was split into training, validation, and test sets, with respective proportions of 70%, 15%, and 15%. This proportion allocation is designed to ensure that the model can learn from a sufficiently large training set while allowing for the fair assessment of model performance through separate validation and test sets. All models utilize networks pre-trained on the ImageNet dataset to enhance training efficiency. The final FC layer was adapted to align with specific classification objectives and kept all network layers trainable (unfrozen) throughout the training process. During network training, due to class imbalance in the dataset, the model state with the highest F1-score (fs) on the validation set was preserved to improve generalization. Then, the preserved model was tested on the test dataset, and its performance was measured in terms of accuracy, precision, recall, and fs. The evaluation metrics were calculated as macro averages for all categories. The model was trained for 80 epochs using an SGD optimizer, with a 0.01 learning rate, 0.9 momentum, and 0.0005 weight decay. The experiments were run on a hardware environment with i7-13700KF CPU, 64G RAM, and NVIDIA GeForce RTX 4090 GPU with CUDA version 11.8. All models are implemented using PyTorch 2.0.1.

4.2 Baseline Model Performance

Tab. 2 details the performance of different models on the Leaf dataset's biotic stress task. The outcomes demonstrate how effectively the proposed method enhances model recognition capacity. The GoogLeNet model achieves the loftiest improvement, increasing accuracy from 91.67% to 95.41%. Efficient NetV2_s achieves the highest accuracy of 96.89%, making it the best model for the biotic stress task. It demonstrates the applicability of the proposed method in improving model performance, especially in the diverse and complex symptom representations of the dataset. This complexity is because the images in the Leaf dataset depict the entire leaf, whereas multiple symptoms are present in some images. With the addition of BC+, this phenomenon becomes more pronounced. During training, the proposed method guides the model to focus on the most severe disease (i.e., GT). Specifically, the loss function of the model is adjusted to recognize features that contribute most to the final decision. This method significantly enhances the robustness of the model in the face of images with complex symptom representations.

The results for the severity task show the same positive trend. The best-performing model, VGG16, has an original

accuracy of 86.51%, which improves to 90.90% with the proposed method. The rest of the models showed a slight improvement. The improvement indicates that the proposed method effectively reduces the overlap between different severity classes in the feature space and promotes clear boundary delineation. Notably, after applying this method, all models show higher precision and recall than the original results, which indicates improved accuracy in identifying positive class samples and reduced false positives and negatives.

Tab. 3 shows the test results for various models on the Symptom dataset. Using the proposed method, the

InceptionV3 model performs exceptionally well, achieving 97.80% accuracy and 97.52% fs. Since the images are cropped to contain only a single symptom, the model focuses on symptom-specific detailed features with explicit learning regions. This method enhances the model's ability to utilize feature details for symptom differentiation by facilitating rational aggregation of samples in the feature space.

Table 4 further demonstrates the performance of various models on the RoCoLe dataset. The images of this dataset are characterized by the complexity of its factual field background and the image labels that incorporate both disease class and severity. The accuracy of the VGG16 model is enhanced from 77.40% to 82.42% after using the proposed method, indicating that the proposed method is quite effective in improving model recognition. In addition, the proposed method motivates the model to improve the aggregation of similar samples within the feature space, thus improving the ability to distinguish between different symptoms and degrees. This effect is reflected in the fs improvement, where the VGG16 enhancement grows the most, from 45.50% to 62.21%, illustrating the positive effect of the method in refining the model for complex feature recognition.

Although all models showed performance improvement after applying the method proposed in this study, the performance on the RoCoLe dataset was lower overall than the BRACOL dataset. The main challenges faced by the models on the RoCoLe dataset include recognizing symptoms from complex backgrounds, which requires a higher degree of ability to distinguish complex patterns. In addition, the multidimensional nature of the image labels requires the model to consider both class discrimination and severity assessment during the learning process, increasing the difficulty of the learning task. Specifically, the background of the field images is highly variable due to factors such as light, leaf density, and soil, which interfere with the model's feature extraction. Moreover, the different manifestations of disease severity require the model to be able to capture more fine-grained feature differences. Given the constraints of the model's parameters, effectively distinguishing disease type and severity levels presents a significant challenge.

The results of this study show that although the natural complexity of the RoCoLe dataset and the multidimensional nature of the labels pose challenges to model learning, the proposed method improves the model's performance on this challenging dataset to some extent.

4.3 Effectiveness of Centroid-Based Loss Adjustment

To thoroughly elucidate the impact of the proposed loss adjustment method, Fig. 5 presents a detailed comparison of loss values before and after adjustment relative to the distance of sample features from their respective class centroids, specifically on the ResNet50 model. The standard quarter line, determined after data normalization, is a reference point in the graph. To the right of the standard quarter line, the adjusted loss value increases as the distance from the feature to the centroid grows, compared to the original loss value, and vice versa on the left side. This indicates that samples located further from their class centroid in the feature space are assigned greater significance in the loss function.

To gain insight into the details of the proposed impact of the method on the feature space, this section presents the spatial density variation of each class in the feature space at each epoch of the training process for ResNet50. Figure 6 depicts the standard deviation of features within each class, indicating the clustering extent within the feature distribution of similar samples.
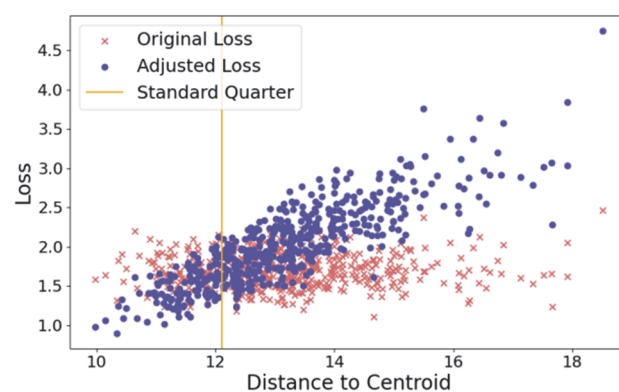

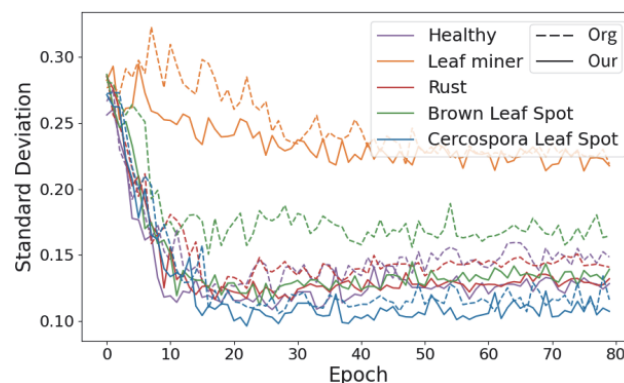**Figure 5** Comparison of adjusted losses

Figure 6 The standard deviation of class for all epochs

As shown in Fig. 6, with the epoch increase, the standard deviation of each class shows decline trends. The standard deviation of healthy leaves (purple) declined faster at the beginning and then gradually stabilized, showing a trend of gradual concentration in the distribution of features. The standard deviations of rust (red), brown leaf spot (green), and Cercospora leaf spot (blue) showed a similar decreasing trend throughout the training process. The overall decreasing trend for leaf miners (orange) was slow. All classes showed fluctuations in some epochs, which reflect dynamic adjustments in the feature space distribution.

It is worth noting that compared to the original method (dashed line), the proposed method (solid line) in this study shows advantages for standard deviation control of

different disease features while maintaining the stability of features. Throughout the training epoch, the proposed method maintains a lower standard deviation, which suggests a more significant sample aggregation in the feature space, enhancing the capacity of the classifier for discrimination.

Overall, the standard deviation of each class gradually decreases as the training proceeds, indicating that the aggregation of the sample distribution increases in the feature space. This result proves that the proposed method enhances the model to learn the similarity between samples of the same class and promotes the discrepancy between each class, further strengthening the model's classification performance.

Table 2 Comparisons with different models on the Leaf dataset

Task	Models	Original [40]				With the Method			
		acc / %	prec / %	rec / %	fs / %	acc / %	prec / %	rec / %	fs / %
BioticStress	AlexNet	92.46	89.37	90.02	-	95.10±0.51	94.63±0.73	91.35±0.45	92.71±0.60
	GoogLeNet	91.67	88.33	89.38	-	95.41±1.03	94.47±0.94	90.70±1.90	92.10±1.45
	VGG16	95.47	96.73	91.94	-	96.40±0.63	96.16±0.58	94.58±0.65	95.10±0.54
	resnet50	95.63	94.12	92.70	-	96.80±0.42	95.77±0.65	95.96±0.55	95.71±0.61
	MobileNetV2	94.84	93.24	91.79	-	96.10±0.58	95.80±0.66	93.56±0.74	94.51±0.68
	InceptionV3*	95.60 ±0.39	94.50±0.46	93.40±0.52	94.25±0.43	96.89±0.34	96.45±0.48	93.68±0.55	95.08±0.45
	EfficientNetV2_s*	96.12±0.55	96.40±0.53	93.70±0.50	94.37±0.45	97.26±0.36	97.98±0.32	94.21±0.57	95.68±0.31
	MobileNetV4_s*	93.10±0.75	91.48±0.70	88.98±0.78	89.50±0.74	94.44±0.69	92.33±0.76	89.24±0.84	90.24±0.70
Severity	AlexNet	84.13	74.23	72.86	-	86.71±0.59	82.98±0.46	77.06±0.48	79.45±0.60
	GoogLeNet	82.94	74.27	73.76	-	86.21±0.70	85.72±0.56	81.05±0.54	82.86±0.47
	VGG16	86.51	82.49	80.89	-	90.90±0.37	88.27±0.46	82.97±0.62	84.85±0.44
	resnet50	84.13	81.66	78.90	-	89.05±0.83	84.12±0.58	79.79±0.54	81.25±0.65
	MobileNetV2	84.52	73.29	77.36	-	88.30±0.51	84.28±1.16	78.41±2.86	80.87±1.52
	InceptionV3*	85.20±0.87	78.05±0.57	76.94±0.62	77.52±0.89	89.77±0.69	82.92±0.68	82.69±0.45	82.51±0.52
	EfficientNetV2_s*	86.75±0.58	80.06±0.52	77.98±0.38	79.72±0.46	89.03±0.26	85.55±0.31	85.52±0.13	85.08±0.22
	MobileNetV4_s*	87.04±0.66	79.44±0.70	78.13±0.74	78.97±0.68	88.76±0.92	87.12±0.68	80.92±0.86	83.01±0.94

*Original model trained by the researcher

Table 3 Comparisons with different models for the Symptom dataset

Models	Original [40]				With the Method			
	acc / %	prec / %	rec / %	fs / %	acc / %	prec / %	rec / %	fs / %
AlexNet	96.58	96.12	96.59	-	96.70±0.23	96.17±0.21	96.83±0.14	96.46±0.21
GoogLeNet	96.82	96.56	96.64	-	96.41±1.03	95.57±0.94	96.45±1.90	96.12±1.45
VGG16	95.6	95.02	95.31	-	96.08±0.25	96.05±0.22	95.80±0.33	95.91±0.36
resnet50	97.07	96.85	96.69	-	97.12±0.19	96.89±0.27	96.81±0.23	96.94±0.16
MobileNetV2	96.33	95.65	96.41	-	96.27±0.55	96.18±0.42	96.03±0.52	96.15±0.42
InceptionV3 [19]	97.61	-	-	-	97.80±0.24	96.82±0.17	98.20±0.41	97.52±0.23
EfficientNetV2_s*	96.25±0.72	96.18±0.60	96.68±0.53	96.30±0.61	97.11±0.69	96.79±0.80	97.22±0.74	97.10±0.66
MobileNetV4_s*	94.55±0.80	93.87±0.72	94.20±0.73	94.02±0.72	95.23±0.85	95.08±0.94	95.02±0.82	95.14±0.85

*Original model trained by the researcher

Table 4 Comparisons with different models for the RoCoLe dataset

Models	Original [30]				With the Method			
	acc / %	prec / %	rec / %	fs / %	acc / %	prec / %	Rec / %	fs / %
AlexNet	73.20	-	-	51.00	75.75±1.60	52.72±1.81	47.99±1.24	47.53±1.54
GoogLeNet	71.90	-	-	45.50	77.28±2.63	55.89±2.48	51.25±2.55	52.59±2.54
VGG16	77.40	-	-	45.50	82.42±1.77	62.07±1.85	72.19±1.75	62.61±1.60
resnet50	75.80	-	-	51.80	80.07±1.55	60.72±1.63	55.89±1.51	58.81±0.51
MobileNetV2	78.70	-	-	50.50	79.66±2.39	55.43±2.67	52.31± 2.54	50.67±2.38
InceptionV3*	77.11±1.07	47.45±1.17	46.12±0.92	46.60±0.96	81.20±0.85	58.88±0.98	53.98±0.84	55.91±0.83
EfficientNetV2_s*	75.98±1.81	53.89±1.44	50.48±1.29	51.95±1.38	79.58±1.62	58.29±1.68	66.77±1.56	58.95±1.60
MobileNetV4_s*	72.59±2.33	43.99±2.47	45.58±2.15	43.08±2.48	76.87±2.62	43.94±2.51	46.17±2.55	44.68±2.52

*Original model trained by the researcher

4.4 Comparison with State-of-the-Art Approaches

In this section, we compare the proposed centroid-based loss adjustment method with several state-of-the-art (SoTA) methods on the BRACOL dataset. Tab. 5 summarizes the performance of these models.

On the Leaf Biotic Stress dataset, our method achieves 97.26% accuracy, 97.98% precision, 94.21% recall and 95.68% F1 score. This outperforms the MTL (ResNet50) method with 95.24% accuracy and the ProtoNet (MobileNetV2, 5-way 5-shot) method with 96.03% accuracy and 93.70% F1 score. While the MTL approach provides a broad framework for handling multiple tasks, it may not fully capture the nuances of complex classification tasks, as its joint learning approach sometimes dilates task-specific optimizations. ProtoNet, on the other hand, is effective for a small amount of learning but may not be able to take full advantage of the available data for finer parameter tuning. Our center-based approach achieves a more precise optimization and improves the overall performance by tuning the internal parameters of the model to pay more attention to the center points of the class.

On the Leaf Severity dataset, our method applied to VGG16 achieves 90.90% accuracy, 88.27% precision, 82.97% recall and 84.85% F1 score. This outperforms Karthik's method (EfficientNetB0 + Ghost), which achieves 84.00% accuracy, and Takuhiro's method (VGG19), which achieves 87.90% F1 score. Karthik's

method has fewer parameters and is computationally efficient, but its performance may be limited by the trade-off between efficiency and model expressiveness. Although Takuhiro's method shows strong feature aggregation ability, it does not necessarily capture task-specific details effectively, especially in more complex datasets. However, our center-based tuning provides a more targeted approach that optimizes model performance for the specific task at hand, leading to higher accuracy and a more balanced trade-off between different evaluation metrics.

On the Symptom dataset, our method applied to InceptionV3 achieves the highest performance with 97.80% accuracy, 96.82% precision, 98.20% recall, and 97.52% F1 score. This outperforms other methods such as TripletNet (MobileNetV2) with an F1 score of 96.38%, and ProtoNet (EfficientNet-B4, 5-way 1-shot) with an F1 score of 96.70%. Although TripletNet performs well in distinguishing between different classes, its performance suffers when dealing with samples with multiple symptoms. ProtoNet, as a small amount learning method, performs well when data is scarce, but may not be able to take full advantage of the abundant data in this task. Our center-based method, which focuses on the center point during training, can deal with the variations in the data more efficiently, leading to better results.

Table 5 Performance comparison of SoTA methods on the BRACOL dataset

Dataset		Model	acc / %	prec / %	rec / %	fs / %
Leaf	Biotic Stress	MTL (Resnet50) [39]	95.24	95.29	91.14	-
		ProtoNet (MobileNetv2, 5-way 5-shot) [41]	96.03	96.12	92.21	93.70
		TripletNet (MobileNetV2) [41]	95.24	94.62	91.45	92.71
		Centroid-Based Loss Adjustment (EfficientNetV2_s)	97.26	97.98	94.21	95.68
	Severity	Karthik (EfficientNetB0 + Ghost) [34]	84.00	76.00	95.33	84.00
		MTL (Resnet50) [39]	86.51	79.50	76.31	-
		Takuhiro (VGG19) [29]	88.10	-	-	87.90
		Centroid-Based Loss Adjustment (VGG16)	90.90	88.27	82.97	84.85
Symptom		TripletNet (MobileNetV2) [41]	96.42	96.24	96.54	96.38
		ProtoNet (EfficientNet-B4, 5-way 1-shot) [41]	96.72	96.66	96.75	96.70
		Centroid-Based Loss Adjustment (InceptionV3)	97.80	96.82	98.20	97.52

4.5 Computational Efficiency and Training Time

Table 6 Comparisons of average training time on Leaf dataset

Backbone	Original / sec	With the Method / sec
AlexNet	4.57	4.69
GoogLeNet	4.64	4.75
VGG16	6.92	7.06
ResNet50	4.90	5.03
MobileNetV2	6.33	6.47
InceptionV3	6.15	6.23
EfficientNetV2_s	5.66	5.82
MobileNetV4_s	5.98	6.15

Along with the increase in effectiveness, this study also examines the change in model training time. Table 6 compares the average training time when using the original model versus applying the proposed method on the Leaf dataset. Even though the proposed method introduces an additional computational step to guide the aggregation of samples in the feature space, a minimal increase in training time is observed. The average increase is 2.24% in training time for models. Although the proposed method slightly increases the training time, it does not affect the inference speed of the model. The proposed method adjusts the loss value during training and does not compute the Euclidean distance again at inference time after deployment, so the model size and inference speed remain constant, ensuring that the method does not impose any additional computational burden on real-time applications. These results show that the proposed method effectively improves the model performance without significantly increasing the computational burden.

5 DISCUSSION

The centroid-based loss adjustment method has obvious advantages in improving the internal parameters according to the distance of the center point to improve the accuracy of the model. However, there are some limitations that need to be addressed.

The estimation of the center point incurs additional computational overhead, which delays the convergence of the model. In our experiments, the training process takes 5 to 10 more epochs to converge compared to the baseline model. While the extra computation is not high, it does affect the overall training time, especially if the dataset is large and more complex.

The performance of the proposed method is sensitive to hyperparameter tuning, especially those that affect the calculation of the distance between the center points. For example, although we used a smoother sine function for optimization if the function value range is too wide within the set interval, it will lead to gradient problems such as vanishing or exploding gradients. On the other hand, if the range of values is too narrow, the method cannot effectively adjust the feature space. Fine-tuning these hyperparameters is crucial to obtain the best results.

It should also be stated that although the center-based loss adjustment method increases the training time, it does not affect the deployment phase. Our method acts like a plug-in that adjusts the loss function during training without changing the model size or inference speed. Therefore, once the model is trained, real-time applications or systems with limited computational resources will not be affected by the proposed method. The centroid-based loss adjustment method has shown strong potential to improve model performance for complex classification tasks, but care must be taken for training time and hyperparameter tuning.

6 CONCLUSION

This study introduces a ground-breaking centroid-based loss adjustment method, addressing the critical challenge of classifying and assessing the severity of coffee

leaf diseases, a growing concern in global coffee production. This method aims to enhance model performance in categorizing and evaluating the seriousness of coffee leaf diseases, addressing agricultural challenges. The technique pays more attention to samples farther away from the class centroid by calculating the centroid for each class and adjusting the loss. This method targets global error minimization and optimizes the distribution of samples in the feature space. Through experimental validation on both the BRACOL and RoCoLe datasets, the proposed method improved classification accuracy by an average of 2.28% compared to standard cross-entropy loss, demonstrating its effectiveness in handling imbalanced data. Notably, it leads to an average f1 increase of 4.06% for the pre-trained deep learning model, highlighting its robustness in real-world classification tasks. The usefulness of the method is further demonstrated by results on numerous pre-trained DL models, particularly in the case of unevenly distributed data. While the method introduces additional computational overhead, experiments show that the increase in training time is limited to 5-10 additional epochs, which is a manageable trade-off given the improved performance without compromising inference speed.

This research aims to refine the method further to manage increased computational requirements potentially arising from a more extensive set of classification labels. Future improvements may include optimizing centroid estimation to reduce computational overhead while maintaining accuracy gains. Additionally, integrating adaptive hyperparameter tuning mechanisms could mitigate sensitivity issues observed during training. Also, to apply the method to newer plans and more efficient DL models, aligning our research with the latest advancements in the field. Expanding and enriching the coffee leaf dataset will be crucial, improving the model's capacity to identify and classify a broader range of disease types. This will provide a more comprehensive solution for managing agricultural diseases and enhance the model's generalization capability. Furthermore, given its modular nature, this method could be extended beyond agricultural applications to other domains such as medical image analysis, remote sensing, and document classification, where feature space optimization is crucial in classification accuracy.

Acknowledgment

7 REFERENCES

[1] Sunarharum, W. B., Williams, D. J., & Smyth, H. E. (2014). Complexity of coffee flavor: A compositional and sensory perspective. *Food Research International*, *62*, 315-325. https://doi.org/10.1016/j.foodres.2014.02.030

[2] Rhiney, K., Guido, Z., Knudson, C., Avelino, J., Bacon, C. M., Leclerc, G., Aime, M. C., & Bebber, D. P. (2021). Epidemics and the future of coffee production. *Proceedings of the National Academy of Sciences*, *118*(27), e2023212118. https://doi.org/10.1073/pnas.2023212118

[3] Araujo, W. L., Dias, P. C., Moraes, G. A., Celin, E. F., Cunha, R. L., Barros, R. S., & DaMatta, F. M. (2008). Limitations to photosynthesis in coffee leaves from different canopy positions. *Plant Physiology and Biochemistry*, *46*(10), 884-890. https://doi.org/10.1016/j.plaphy.2008.05.005

[4] Venzon, M., Rosado, M. C., Fadini, M. A. M., Ciociola Jr, A. I., & Pallini, A. (2005). The potential of NeemAzal for the control of coffee leaf pests. *Crop Protection*, *24*(3), 213-219. https://doi.org/10.1016/j.cropro.2004.07.008

[5] Azmi, M. A. & Naqvi, S. (2011). Pesticide pollution, resistance, and health hazards. *Pesticides-The Impacts of Pesticides Exposure*, *26*. https://doi.org/10.5772/13758

[6] Hawkins, N. J., Bass, C., Dixon, A., & Neve, P. (2019). The evolutionary origins of pesticide resistance. *Biological Reviews*, *94*(1), 135-155. https://doi.org/10.1111/brv.12440

[7] Ul Haq, I. & Ijaz, S. (2020). History and recent trends in plant disease control: An overview. *Plant Disease Management Strategies for Sustainable Agriculture through Traditional and Modern Approaches*, 1-13. https://doi.org/10.1007/978-3-030-35955-3_1

[8] Benos, L., Tagarakis, A. C., Dolias, G., Berruto, R., Kateris, D., & Bochtis, D. (2021). Machine learning in agriculture: A comprehensive updated review. *Sensors*, *21*(11), 3758. https://doi.org/10.3390/s21113758

[9] Mengistu, A. D., Alemayehu, D. M., & Mengistu, S. G. (2016). Ethiopian coffee plant diseases recognition based on imaging and machine learning techniques. *International Journal of Database Theory and Application*, *9*(4), 79-88. https://doi.org/10.14257/ijdta.2016.9.4.07

[10] Chauhan, N. K. & Singh, K. (2018). A review on conventional machine learning vs deep learning. *2018 International Conference on Computing, Power and Communication Technologies (GUCON)*, 347-352. https://doi.org/10.1109/GUCON.2018.8675097

[11] Pak, M. & Kim, S. (2017). A review of deep learning in image recognition. *2017 4th International Conference on Computer Applications and Information Processing Technology (CAIPT)*, 1-3. https://doi.org/10.1109/CAIPT.2017.8320684

[12] Noon, S. K., Amjad, M., Qureshi, M. A., & Mannan, A. (2020). Use of deep learning techniques for identification of plant leaf stresses: A review. *Sustainable Computing: Informatics and Systems*, *28*, 100443. https://doi.org/10.1016/j.suscom.2020.100443

[13] Raghavendra, B. (2023). An efficient approach for coffee leaf disease classification and severity prediction. *International Journal of Intelligent Engineering & Systems*, *16*(5). https://doi.org/10.1016/j.compag.2019.105162

[14] Ushadevi, G. (2020). A survey on plant disease prediction using machine learning and deep learning techniques. *Inteligencia Artificial*, *23*(65), 136-154. https://doi.org/10.4114/intartif.vol23iss65pp136-154

[15] Sorte, L. X. B., Ferraz, C. T., Fambrini, F., dos Reis Goulart, R., & Saito, J. H. (2019). Coffee leaf disease recognition based on deep learning and texture attributes. *Procedia Computer Science*, *159*, 135-144. https://doi.org/10.1016/j.procs.2019.09.168

[16] Zambolim, L. (2016). Current status and management of coffee leaf rust in Brazil. *Tropical Plant Pathology*, *41*(1), 1-8.

[17] de Resende, M. L., Pozza, E. A., Reichel, T., & Botelho, D. M. (2021). Strategies for coffee leaf rust management in organic crop systems. *Agronomy*, *11*(9), 1865. https://doi.org/10.3390/agronomy11091865

[18] Krishnan, S. (2017). *Sustainable coffee production*. Oxford Research Encyclopedia of Environmental Science. https://doi.org/10.1093/acrefore/9780199389414.013.224

[19] Kumar, M., Gupta, P., & Madhav, P. (2020). Disease detection in coffee plants using convolutional neural network. *2020 5th International Conference on*

Communication and Electronics Systems (ICCES), 755-760. https://doi.org/10.1109/ICCES48766.2020.9138000

[20] Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*. https://doi.org/10.1109/TNNLS.2021.3084827

[21] Lisboa, E., Lima, G., & Queiroz, F. (2021). Coffee Leaf Diseases Identification and Severity Classification using Deep Learning. *Anais Estendidos do XXXIV Conference on Graphics, Patterns and Images*, 201-205. https://doi.org/10.5753/sibgrapi.est.2021.20039

[22] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770-778. https://doi.org/10.48550/arXiv.1512.03385

[23] Binney, E. & Ren, D. (2023). Coffee Leaf Diseases Classification and the Effect of Fine-tuning on Deep Convolutional Neural Networks. *IJFMR-International Journal for Multidisciplinary Research*, 4(5).

[24] Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4700-4708. https://doi.org/10.1109/CVPR.2017.243

[25] Simonyan, K. & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556. https://doi.org/10.48550/arXiv.1409.1556

[26] Karthik, R., Alfred, J. J., & Kennedy, J. J. (2023). Inception-based global context attention network for the classification of coffee leaf diseases. *Ecological Informatics*, 77, 102213. https://doi.org/10.1016/j.ecoinf.2023.102213

[27] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1-9. https://doi.org/10.48550/arXiv.1409.4842

[28] Cao, Y., Xu, J., Lin, S., Wei, F., & Hu, H. (2019). Gcnet: Non-local networks meet squeeze-excitation networks and beyond. *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 0-0. https://doi.org/10.48550/arXiv.1904.11492

[29] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30. https://doi.org/10.48550/arXiv.1706.03762

[30] Okada, T., Huang, Y., Hao, G., Iizuka, S., & Fukui, K. (2023). Low-Level Feature Aggregation Networks for Disease Severity Estimation of Coffee Leaves. *2023 18th International Conference on Machine Vision and Applications (MVA)*, 1-5. https://doi.org/10.23919/MVA57639.2023.10215626

[31] Kim, M., Jain, A. K., & Liu, X. (2022). Adaface: Quality adaptive margin for face recognition. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18750-18759. https://doi.org/10.48550/arXiv.2204.00964

[32] Li, H. (2018). Deep learning for natural language processing: advantages and challenges. *National Science Review*, 5(1), 24-26. https://doi.org/10.1093/nsr/nwx110

[33] Manso, G. L., Knidel, H., Krohling, R. A., & Ventura, J. A. (2019). A smartphone application to detect and classify coffee leaf miner and coffee leaf rust. arXiv preprint arXiv:1904.00742. https://doi.org/10.48550/arXiv.1904.00742

[34] Pinto, L. A., Mary, L., & Dass, S. (2021). The Real-Time Mobile Application for Identification of Diseases in Coffee Leaves using the CNN Model. *2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, 1694-1700. https://doi.org/10.1016/j.inpa.2021.01.004

[35] Ramamurthy, K., Thekkath, R. D., Batra, S., & Chattopadhyay, S. (2023). A novel deep learning architecture for disease classification in Arabica coffee plants. *Concurrency and Computation: Practice and Experience*, 35(8), e7625. https://doi.org/10.1002/cpe.7625

[36] Tan, M. & Le, Q. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. *International Conference on Machine Learning*, 6105-6114. https://doi.org/10.48550/arXiv.1905.11946

[37] Han, K., Wang, Y., Tian, Q., Guo, J., Xu, C., & Xu, C. (2020). GhostNet: More features from cheap operations. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1580-1589. https://doi.org/10.48550/arXiv.1911.11907

[38] Wang, Z., Ren, Z., & Li, X. (2023). Identification of Coffee Leaf Pests and Diseases based on Transfer Learning and Knowledge Distillation. *Frontiers in Computing and Intelligent Systems*, 5(1), 15-18. https://doi.org/10.54097/fcis.v5i1.11537

[39] Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531. https://doi.org/10.48550/arXiv.1503.02531

[40] Esgario, J. G., Krohling, R. A., & Ventura, J. A. (2020). Deep learning for classification and severity estimation of coffee leaf biotic stress. *Computers and Electronics in Agriculture*, 169, 105162. https://doi.org/10.1016/j.compag.2019.105162

[41] Zhang, Y. & Yang, Q. (2021). A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*, 34(12), 5586-5609. https://doi.org/10.48550/arXiv.1707.08114

[42] Cen, S. & Lim, C. G. (2024). Multi-task Learning of the PatchTCN-TST Model for Short-term Multi-load Energy Forecasting Considering Indoor Environments in a Smart Building. *IEEE Access*. https://doi.org/10.1109/ACCESS.2024.3355448

[43] 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

[44] Schroff, F., Kalenichenko, D., & Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 815-823. https://doi.org/10.48550/arXiv.1503.03832

[45] Snell, J., Swersky, K., & Zemel, R. (2017). Prototypical networks for few-shot learning. *Advances in Neural Information Processing Systems*, 30. https://doi.org/10.48550/arXiv.1703.05175

[46] Wen, Y., Zhang, K., Li, Z., & Qiao, Y. (2016). A discriminative feature learning approach for deep face recognition. *Computer vision - ECCV 2016: 14th European conference*, part VII, 499-515. https://doi.org/10.1007/978-3-319-46478-7_31

[47] Wang, Z., Wang, R., Wang, M., Lai, T., & Zhang, M. (2024). Self-supervised transformer-based pre-training method with General Plant Infection dataset. *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, 189-202. https://doi.org/10.1007/978-981-97-8490-5_14

[48] Tokozume, Y., Ushiku, Y., & Harada, T. (2018). Between-class learning for image classification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5486-5494. https://doi.org/10.48550/arXiv.1711.10284

[49] Parraga-Alava, J., Cusme, K., Loor, A., & Santander, E. (2019). RoCoLe: A robusta coffee leaf images dataset for

evaluation of machine learning-based methods in plant diseases recognition. *Data in Brief*, *25*, 104414. https://doi.org/10.1016/j.dib.2019.104414

[50] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, *25*. https://doi.org/10.1145/3065386

[51] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4510-4520. https://doi.org/10.48550/arXiv.1801.04381

[52] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the Inception architecture for computer vision. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2818-2826. https://doi.org/10.48550/arXiv.1512.00567

[53] Tan, M. & Le, Q. V. (2021). EfficientNetV2: Smaller models and faster training. arXiv preprint. https://doi.org/10.48550/arXiv.2104.00298

[54] Qin, D., Leichner, C., Delakis, M., Fornoni, M., Luo, S., Yang, F., Wang, W., Banbury, C., Ye, C., Akin, B. et al. (2024). Mobilenetv4-universal models for the mobile ecosystem. arXiv preprint. https://doi.org/10.1007/978-3-031-73661-2_5

Contact information:

Changyu AO
Department of Computer Engineering,
Chonnam National University,
Yeosu 59626, South Korea
E-mail: aochangyu@jnu.ac.kr

Gwang-Jun KIM
Department of Computer Engineering,
Chonnam National University,
Yeosu 59626, South Korea
E-mail: kgj@chonnam.ac.kr

Man-Sung KWAN
(Corresponding author)
Department of Computer Engineering,
Chonnam National University,
Yeosu 59626, South Korea
E-mail: evergreen_ms@hanmail.net