IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 28 May 2023, accepted 13 June 2023, date of publication 16 June 2023, date of current version 23 June 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3286935

RESEARCH ARTICLE

Model Selection of Hybrid Feature Fusion for Coffee Leaf Disease Classification

MUHAMAD FAISAL[1], **JENQ-SHIOU LEU**[1], **(Senior Member, IEEE), AND JEREMIE T. DARMAWAN**[2]

[1]Department of Electronic and Computer Engineering (ECE), National Taiwan University of Science and Technology, Taipei 106, Taiwan
[2]Department of Bioinformatics, Indonesia International Institute for Life Sciences, Jakarta 13210, Indonesia

Corresponding author: Jenq-Shiou Leu (jsleu@mail.ntust.edu.tw)

ABSTRACT Coffee leaf diseases can significantly impact the productivity and quality of the crops. Accurate and timely identification of these diseases is crucial for effective management and control. This paper proposes a hybrid feature fusion approach for identifying coffee leaf disease, including early and late feature fusion. First, we propose several hybrid models to extract the information feature in the input images by combining MobileNetV3, Swin Transformer, and variational autoencoder (VAE). MobileNetV3, acting on the inductive bias of locality, can extract image features that are closer to one another (local features), while the Swin Transformer is able to extract feature interactions that are further apart (high-level features). These differently extracted features contain complementary information that enriches a unified feature map. Second, the extracted images from models are fused in the early fusion network. The early-fusion learner network is deployed to learn the rich information from the extracted feature. The late fusion network is implemented to comprehensively learn the fused feature before a classification network classifies coffee leaf diseases. The proposed hybrid feature fusion approach is evaluated on a challenging, real world Robusta Coffee Leaf (RoCoLe) dataset with various diseases, including red spider mite and leaf rust disease. The results show that our approach, the hybrid feature fusion of MobileNetV3 and Swin Transformer, outperforms the individual models with an accuracy of 84.29%. In conclusion, the hybrid feature fusion approach combining MobileNetV3 and Swin Transformer models is a promising approach for coffee leaf disease identification, providing accurate and timely diagnosis for effective management and control of the diseases in real-world conditions.

INDEX TERMS Coffee leaf disease classification, feature fusion, hybrid model.

I. INTRODUCTION

Agriculture forms a big portion of the world's economy, amounting to over 4% of the world's gross domestic product (GDP) [1]. Additionally, its economic benefits extend to reducing overall poverty, raising income, and increasing employment possibilities in many African and Asian countries such as Vietnam, Indonesia, and Ethiopia. Their main type of coffee production is Robusta beans which are frequently plagued with major diseases such as leaf rust and red spider mites. These plagues can impact yield losses by over 75% in severe cases and over two billion US dollars annually [2]. Typically, trained personnel manually inspect individual leaves to identify the presence of any diseases. This solution would require extensive time and effort since coffee plantations may start from 5 hectares up to several hundred hectares.

Initially, computer vision and machine learning algorithms have been proposed to solve the detection of plant leaf diseases [3], [4]. The leaf was captured using a digital camera and concurrently identified using machine learning algorithms such as k-means clustering [3], radial basis function network [4], etc. However, the machine learning algorithm requires hand-crafted feature extractions that were prepared on individual images to obtain meaningful features before the classifier performs classification [5], [6], [7]. Therefore, the deep learning algorithm is proposed to automatically perform feature extraction [6], [8], [9]. In the

The associate editor coordinating the review of this manuscript and approving it for publication was Kathiravan Srinivasan.

meantime, many plant leaf disease datasets have been generated [10], [11], [12], [13]. Eventually, deep learning algorithms have been gaining attention in classifying plant leaf diseases [14], [15], [16]. As a result, Convolutional Neural Networks (CNN), a subset of deep learning, have been intensively utilized. CNN is able to perform the necessary feature extractions automatically and has several popular architectures. CNN models such as VGG [14], ResNet [15], DenseNet [16], EffecientNet [7], [17], and MobileNet [18] that were trained on ImageNet [19] are proposed and successfully classified plant leaf diseases. In addition, the combined CNN model and attention mechanism have also been introduced [20], [21].

Although the improvements occur from the use of an advanced deep learning approach over traditional solutions, there remains a challenge for unconditioned images in real-world situations [17]. Real-world condition images are often taken using a mobile device without much modification to the background of the leaf subject. Unmodified backgrounds are usually similarly colored to the issue, including a combination of other leaves, grasses, and trees. Hence, there is not much color distinction between the image elements. Traditionally, CNNs are the popular technique for image classification tasks due to their inductive biases associated with locality and weight sharing [22]. Both of these rely heavily on the contrasts that can be drawn out from closely located pixels of an image. Therefore, it is assumed that the performance of CNN-based image classification models would significantly impact when applied to real-world condition leaf diseases [17]. Each CNN backboned model has a distinct way of extracting features from an image based on their architecture. There may be models that focus more on individual channels (depthwise convolutions) or larger spatial dimensions (spatial convolutions) of an image and thus resulting in different feature vector maps generated.

Feature fusion is a deep learning technique that combines several feature maps generated from CNN, or any hierarchical models, into an enriched integrated feature representation [23]. Combinations often employ concatenation, which may compensate for the inadequacy of features extracted from a single network [24]. Other advantages that may be drawn out from using this technique over creating a single large network are lower computational costs and improved classification capabilities. Apart from feature fusion, the possibility of combining decision-level outputs from multiple networks like that of an ensemble model exists. A hybrid of both these techniques may also be employed to reap even more benefits.

This study strives to resolve the difficulties encountered in classifying coffee leaf diseases in real-world conditions. The root problem could lie in the real-world coffee leaf disease images, which often showcase subjects that blend in or have similar colors to the background. These conditions can be observed in the Robusta Coffee Leaf (RoCoLe) dataset [11], which consists of 1,560 leaf images showing red spider mites and coffee leaf rusts equipped with detailed annotations. Thus, as mentioned earlier, the dataset is suitable for evaluating the problem in the real-world. Other studies have struggled to obtain an effective classification performance using a single network scheme, and feature fusions are yet to be investigated to solve this problem. Based on the preliminary findings that feature fusion could enrich extracted features and produce better results, we employed a series of experiments to investigate feature fusion and its various variants in this real-world coffee leaf disease problem. This study has several contributions to be highlighted:

1. We propose the use of hybrid feature fusion, including early and late feature fusion in real-world coffee leaf disease classification

2. We evaluate the performance of several hybrid feature fusions models on the RoCoLe dataset and derive a model selection criterion.

3. We investigate the effectiveness of hybrid fusion against a single conventional network based on real-world condition RoCoLe dataset [11] classification performance

Apart from this introduction, this paper is structured with four remaining main sections: the related works of deep learning in plant leaf diseases, the methodology used in this study, the experiment results, and conclusions. The methodology includes the feature extractions of different deep learning methods, including MobileNetV3 and Swin Transformer, different types of feature fusions, and fusion model selection for our proposed method. In our experiment results, a comparative report on the developed model for selection was done, followed by a comparison to other benchmarks and the proposed method. The end of this paper is concluded with a summary of the findings of this study.

II. RELATED WORKS

Agriculture plays a vital role in a country's economy by providing food and employment opportunities for a large portion of the population. However, plant diseases can lead to inferior crop yield and hinder economic growth. Several proposed techniques have been developed to address plant leaf disease identification challenges. In this section, we provide a brief overview of some relevant research techniques in this field.

A new classification technique is developed in [25] using the boosted support vector machine-based Arithmetic optimization algorithm. The input image is segmented by the vector value model, then extracted by the greyscale co-occurrence matrix. The extracted feature is classified using the proposed method, achieving an accuracy of 98.6%. In [26], a detection for local tomato leaf disease is proposed based on image processing. The gray level co-occurrence matrix is employed as feature extraction to calculate 13 statistical features. Then, a support vector machine is selected to classify those features into four classes. In [14], a new framework called AgriDet was developed by combining the inception module and VGG network to classify plant leaf disease. The Inception-VGG is used as a feature extractor, while the Kohonen layer is employed to learn multi-scale features. The framework includes the pre-processing image part, which encompasses scaling, enhancement, and contrast. The proposed framework achieves an accuracy of 93.24%. In [16], an improved DenseNet was proposed to classify potato leaf diseases. The improved DenseNet has an extra transition layer and a re-weighted cross-entropy to enhance classification in the imbalanced dataset. The proposed method achieved an accuracy of 97.2%.

In [20], a modification of GoogLeNet, rE-GoogLeNet, is developed by replacing the kernel filter from 7 × 7 to three kernel filters of 3 × 3, adding an ECA attention in the inception module, a residual network, and leaky-ReLU. The proposed network includes a simple classifier to reduce the complexity. The modified network achieved an accuracy of 99.58%. In [27], a combination of principal component analysis and a deep neural network called DeepNet is proposed. A generative adversarial network (GAN) is employed to add a mixture to the dataset. A faster-Region-based Convolutional Neural Network (F-RCNN) is used as a classifier and achieves an accuracy of 99.60%. In [28], a fine-grained disease classification that utilizes attention mechanisms is proposed. A reconstruction-generation model, as well as adversarial loss, are employed to suppress the noise. The proposed method achieves higher accuracy and less memory since the model does not increase the model complexity during inference. In [18], a transfer learning method using pre-trained MobilNetV2 is proposed for tomato leaf diseases to extract input features. The augmentation step is utilized to tackle the imbalanced dataset. The proposed method achieves a floating point of 4.87M and a size of 9.69MB with an accuracy of 99.30%. An efficient CNN is also proposed in [6] by combining the Inception module and pre-trained MobileNet called MobInc-Net. The proposed MobInc achieves an accuracy of 97.89% for the custom dataset. In [7], two pre-trained CNNS, EfficientNetB0 and DenseNet121, are utilized to extract deep features of corn images. The proposed CNNs obtain an accuracy of 98.56%, superior to other pre-trained CNNs. In [21], a combination of transformer and Inception is proposed to capture long-range and cross-channel features to improve fine-grained learning. This model outperforms previous models based on convolution and vision transformers, achieving high accuracy on multiple datasets with an accuracy of 99.94% on the Plant Village dataset, 99.22% on the ibean dataset, 86.89% on the AI2018 dataset, and 77.54% on PlantDoc. In [17], several plant leaf datasets, including RoCoLe, BRACOL, Plant Pathology, and Plant Village, were analyzed using several pre-trained models such as EfficientB0, MobileNetV2, InceptionV2, ResNet50, and VGG16. The proposed method explored the influential factors affecting models' accuracy using different conditions: laboratory and real-life conditions. In conclusion, the accuracy dropped from 92.67% to 54.41% in the worst case due to the complexity of the real-life background. In [15], an attempt to explore the visualization was presented using several approaches. The proposed method was expected to understand what the deep learning model sees when classifying images. In the end, the guided SVM was utilized to show the accuracy improvement compared to the Naïve approach. Table 1 shows the comparison of related works.

III. METHODOLOGY

A. FEATURE EXTRACTION USING DEEP LEARNING

1) MOBILENETV3

Based on the MobileNet family of CNN models, the third iteration of this family presents an improved performance over the MobileNetV2 in terms of accuracy, by 3.2%, and inference time, over 20%, on the ImageNet benchmark. This model places among the best for performance to inference time ratio. The architecture still has the inverted residual and linear bottleneck that MobileNetV2 has. However, several architectural changes were made in order to achieve such improvements, namely the addition of squeeze and excitation in the residual layer, a modified swish nonlinearity called hard-swish, a redesign of the computationally costly last stage, as well as platform-aware Neural Architecture Search (NAS) for block search and NetAdapt for per layer search of an optimal number of filters. The squeeze and excitation implemented for MobileNetV3 [29] differ from that found in MnasNet, despite being built upon the same MobileNetV2. MnasNet adopted the squeeze and excitation into the bottleneck layers, whereas the MobileNetV3 applied it on the residual layer with ReLu and hard-swish nonlinearities depending on the layer. Two developed versions of the MobileNetV3 targetted a low and high resource corresponding to the small and large suffixes. As with most CNN backbone models, the features extraction method relies heavily on inductive biases from the locality and weight sharing that thrive on opposite edges [22].

2) SWIN TRANSFORMER

In CNN architectures, the typical interaction between pixels of an image is based on the assumption of locality, which gives more importance to drastic pixel intensity changes from closeby pixels than those that are further apart. On the other hand, Swin Transformer does not have a CNN backbone for vision classification. It leverages on the transformer architecture that are commonly found in Natural Language Processing (NLP) and more recently imported into computer vision tasks in Vision Transformers [30] and its derivatives [31]. As previously identified, there are issues regarding the use of transformers in vision tasks, namely the greater requirement for training data and consequent lack of inductive biases, when compared to CNNs [30]. The Swin Transformer is developed using shifting windows self-attention to tackle this problem. By introducing a hierarchical representation of self-attention on local windows while also allowing for global cross-window relationships, the benefit of transformers in capturing long-range, global pixel interactions is not lost. Compared to the ViT that

TABLE 1. The summary of related works.

Author	Methodology	Dataset	Plant	Class	Metric
M. Prabu, et al, 2023 [25]	Boosted SVM using Arithmetic Optimization	Plant Village	14 plants	38 Diseases from 14 plant leaves	Accuracy, Precision, Recall, Specificity, F_1-Score
S. U. Rahman et al, 2023 [26]	Gray Level Co Occurrence Matrix using SVM	Tomato from Pakistan field	Tomato	Early blight, Late blight, Septoria leaf spot, Healthy	Accuracy
A. Pal et al, 2023 [14]	Inception-VGG with Kohonen Network	Plant Village, Plant Doc	14 plants	38 Diseases from 14 plant leaves	Accuracy, Sensitivity, Specificity
R. Mahum *et al*, 2023 [16]	The Improved DenseNet-201	Plant Village	Potato	Late blight, Early blight, Leaf roll, Verticillium wilt, Healthy	Accuracy, Precision, Recall, F_1-Score
L. Yang *et al.*, 2023 [20]	The Improved GoogLeNet with ECA Attention	Jiangxi Agricultural University Dataset, Kaggle	Rice	Aphelenchoides Bessyi, Bacterial leaf blight, Bacterial leaf streak, Brown spot, Leaf smut, Red blight, Rice blast, Rice sheath blight	Accuracy, Precision, Recall, F_1-Score
K. Roy *et al.*, 2023 [27]	PCA and Custom DNN	Plant Village Dataset	14 plants	38 Diseases from 14 plant leaves	Accuracy, Precision, Recall, F_1-Score
Y. Wu, et al, 2022 [28]	Attentional CNN	Plant Village Dataset	14 plants	38 Diseases from 14 plant leaves	Accuracy
S. Ahmed, et al, 2022 [18]	The Improved MobileNetV2	Plant Village Dataset	14 plants	38 Diseases from 14 plant leaves	Accuracy, Precision, Recall, F_1-Score
J. Chen, et al 2020, [6]	VGG-Inception Net	The Fujian Institute of Subtropical Botany Dataset	Rice, Maize	Stackburn, Scald, Smut, White tip (Rice); Bacterial leaf streak, Phaeosphaeria spot, Maize eyespot, Gray leaf spot, and goss's bacterial wilt (Maize)	Accuracy, Sensitivity, Specificity
H. Amin, et al 2022 [7]	EfcientNetB0, DenseNet121	Plant Village Dataset	Corn	Northern leaf blight, Cmmon rust, and Gray leaf spot, Healthy	Precision, Recall, F_1-Score
S. Yu, et al 2023 [21]	Inception Convolutional Vision Transformer	Plant Village, iBean, AI2018, PlantDoc	14 plants (plant Village); bean (iBean); 10 plants (AI2018); 13 plants (PlantDoc)	38 diseases (plant Vilage); 3 (iBean); 59 (AI2018); 27 (PlantDoc)	Accuracy, Precision, Recall, F_1-Score
G. Fenu et al, 2022 [17]	EfficientB0, MobileNetV2, InceptionV2, ResNet50, VGG16	RoCoLe , BRACOL, Plant Pathology, Plant Village	Coffee (RoCoLe); Coffee (BRACOL); Apple (Plant Pathology); Apple (Plant Village)	Healthy, Red spider mite, Rust (RoCoLe); Leaf miner, Leaf rust, Brown leaf spot and Cercospora leaf spot (BRACOL); Healthy, Cedar rust, Scab leaves (Plant Pathology); Cedar rust, Scab, Healthy (Plant Village)	Accuracy, Precision, Recall, F_1-Score
M. Yebasse, et al 2021 [15]	The Guided ResNet	RoCoLe	Coffee	Healthy, unhealthy	Accuracy

directly works on large image patches, the Swin constructs multiple layers of differently-sized patches of the same image as shown in Figure 1. By doing so, the layers could form a hierarchical representation that enables the Swin to leverage the multi-scaled information for advanced dense predictions, including for classification purposes [31]. It also presents computational benefits linear to image size and the ability to operate at various scales. These features have allowed it to achieve 87.3% accuracy on the ImageNet-1K benchmark. In contrast to CNN backbone models, the feature extraction of this Swin Transformer does not rely solely on inductive biases but also include the global self-attention mechanism embedded in the transformer architecture.

3) VARIATIONAL AUTOENCODER

Autoencoders (AE) are unsupervised learning techniques that take data inputs without any label to guide the learning process. It reduces dimensionality and is a non-linear principal component analysis (PCA). There are two main parts of an autoencoder: encoder and decoder. The encoder functions to encode input information into the latent encoded vectors. In contrast, the decoder works in the opposite direction, that is, to regenerate the features back from the latent vectors. Latent vectors are products of the encoder by compressing the input information. Outputs of the AE are expected to represent the original data better.

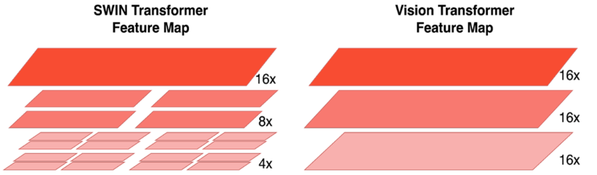

FIGURE 1. The illustration of the hierarchical feature maps generated by swin transformer compared to the single-scale feature map of the vision transformer.

A variational autoencoder (VAE) employs principles of probability theory and Bayes theorem. Instead of using a function to create the latent attributes, VAE utilizes a probability distribution to represent the latent space. Mathematically, the VAE takes in observed data x and attempts to obtain a good value of latent variables z by using $p(z|x)$, or simply $p(z)$, to generate a distribution of possible parameters [33]. As the decoder is simply reconstructing the latent variable back into data, the process can be thought of as $p(x|z)$. The overall network can be thought of as a joint probability model $p(x|z) = p(x|z)\ p(z)$. Since the latent variable is generated using a probability distribution, it is possible for the final output to be new content. Hence, it inherently has a generative property. Additionally, due to the Bayesian properties of the VAE, it is more suitable for larger datasets in terms of computation and could provide better representations [34]. Incorporating VAE into an image classification network would mean that novel image information is introduced within the neural network. As more variations are introduced to the neural network, and assuming the network learns properly, the final model's generalization ability should improve.

B. FEATURE FUSION

When the use of single networks is not sufficient, a combination of several models may be the most convenient method of developing a new high-performing network. This method of using multi-modal machine learning could be referred to as fusion. Fusion simply refers to the combination of two or more networks by means of concatenation. Rather than addition and subtraction operation that alters the weight value in a particular layer, concatenation simply creates a larger vector on one of the dimensions and due to this it can be done in varying stages of the network. However, it is a requirement for the feature maps of the two modalities to have one identical dimension values that will be concatenated. A consequence of this technique is the massive layer created immediately after the fusion is done. By leveraging multiple machine learning models to perform the task at hand, performance may experience a boost while inference time and model size would need to be compromised.

1) EARLY-FEATURE FUSION

Since most computer vision networks' early section functions extract meaningful features, early-stage fusion combines the learned feature maps of each model into a single, large feature representation. Mathematically, it could be described as $x_{concat} =$ concatenate $[x_1, x_2, \ldots, x_n]$ where x_i is the input vector of each modality. Further processing would be done on this unified feature map as x_{concat} would be the input to the first DL layer. Compared to a single network feature map, this fused feature representation would be highly enriched with features from multiple networks. As the network does not differentiate features from different modalities, there can be two contrasting scenarios when using the early-feature fusion technique. Ideally, to obtain an improved performance using this approach, each fused model must focus on different meaningful features that complement each other. As a result, it would allow later stages of the network to make decision scores based on cross-modal features. However, combining a perfectly feature map from one network with another can also be combined with another that merely extracts noise features from the same image. In this case, the fusion would create a disrupted feature map, leading to poorer performance. This early-feature fusion is the lesser-used method, as the dimension of the generated feature maps could often vary with each modality. Different kernel sizes, strides, and paddings are among the reasons each model has differently sized feature maps. It has also been reported that early-feature fusion provides more robust performances when varying noise levels are involved compared to late fusion [35].

2) LATE-FEATURE FUSION

In this method, each input feature is extracted and subsequently processed by each network individually. Fully connected layers at each modality's end are responsible for creating prediction scores. Each base model prediction is then aggregated to form a final prediction of $p(y|x_i)$, where x_i is the prediction scores from the i-th base models. Prediction error and generalization error can be reduced by using this technique as it employs the wisdom of crowds to make predictions [36]. However, the drawback of this technique is that multimodal relationships are not learned by the final model [37]. This could be an advantage when different modalities are less correlated. It could be utilized with shallow machine learning and deep learning as late fusion combines decision-level predictions rather than learned extracted features. Like bagging ensemble modeling, late-feature fusion has been employed more frequently due to its practicality [38]. An imbalance in the input dimensions does not affect the final predictions as higher dimensional modalities take precedence over lower dimensions [37].

C. PROPOSED METHOD

The early- and late-feature fusion techniques have their benefits and drawbacks. It can also be implemented in most computer vision architectures. In hopes that the benefits can be leveraged for better performance, this study sought to incorporate both techniques into a single neural network architecture and call it a hybrid fusion. Hybrid fusion

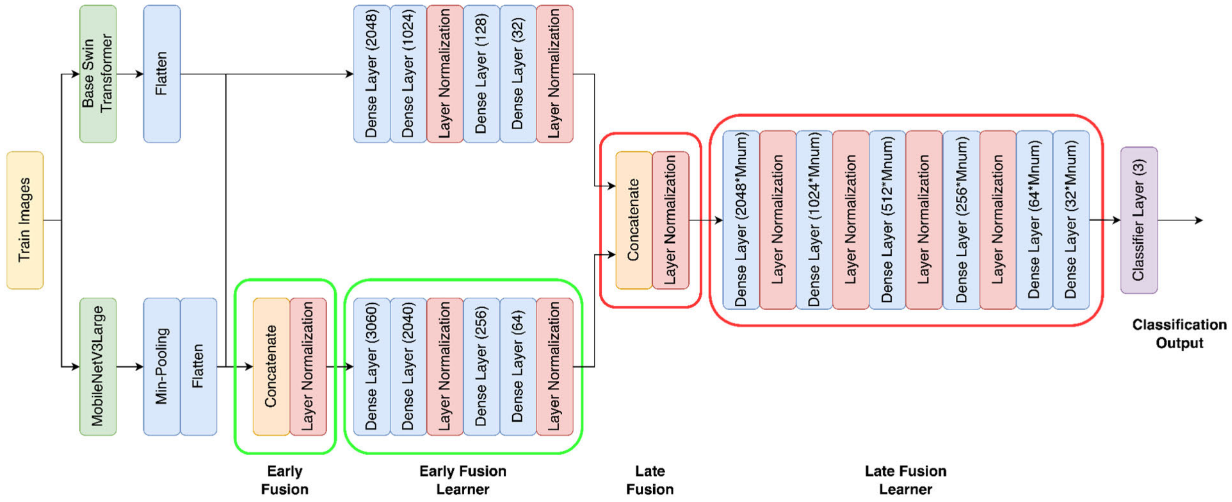

FIGURE 2. **The architecture of hybrid fusion of MobileNetV3 and swin transformer.**

would incorporate the early- feature fusion technique and the late-late feature fusion a single network. Early fusion would offer an enriched feature vector for the consequent layer of the network through its cross-modality feature correlations and inherent noise-resistant design. In contrast, late fusion minimizes the generalization and prediction errors. The most apparent drawback of this technique would be the larger final weights associated with the multiple concatenations after fusion.

1) MOBILENETV3 AND SWIN TRANSFORMER

As the final model was likely to contain large weights due to the fusion of multiple models, we employed a relatively mobile model in the MobileNetV3Large alongside the base Swin Transformer. The intuition behind the fusion of a CNN backbone with a Transformer backbone is to introduce a global self-attention that focuses on long-range dependencies into the locally-focused inductive biases. As a result, the final model would consider both global and local properties of the image and optimize its importance as it learns. For this particular architecture, a Mnum or multiplier number of 2 and 4 were experimented with. For ease of comparison, later on, the model with a multiplier of 2 will be called the 'mini' version. The choice of activation function was swish for the late fusion learner block, and all other dense layers were given ReLU non-linearity. Figure 2 shows the architecture of the hybrid feature fusion of MobileNetV3 and Swin Transformer.

2) VAE-CNN AND SWIN TRANSFORMER

This model was developed involving three distinct architectures, including VAE, Large Swin Transformer, and a custom-crafted 2D-Convolution Network. Apart from the Swin Transformer, all other modalities were trained from scratch. The custom Conv2D model architecture can be observed in Table 2. In this architecture, since more modalities are involved, the base number of units for the dense layers was raised, whereas the Mnum value was assigned a value of 3. All layers after the three modalities were given a swish non-linearity activation function. Similarly, to the previous model, we incorporated an early and late fusion with their dense learner blocks. Figure 3 shows the architecture of the Hybrid Fusion of VAE-CNN and Swin Transformer.

TABLE 2. **Units for magnetic properties parameter of custom Conv2d model.**

Layer	Parameter	Value
Conv2D	Filters, Kernel Size, Padding, Activation	32, 2, 'same', 'relu'
Conv2D	Filters, Kernel Size, Padding, Activation	32, 2, valid, 'relu'
MinPooling	Pool Size, Padding	2, 'valid'
	Layer Normalization	
Conv2D	Filters, Kernel Size, Padding, Activation	16, 3, 'same', 'relu'
Conv2D	Filters, Kernel Size, Padding, Activation	16, 3, 'valid', 'relu'
MinPooling	Pool Size, Padding	2, 'valid'
	Layer Normalization	
Conv2D	Filters, Kernel Size, Padding, Activation	32, 3, 'same', 'relu'
Conv2D	Filters, Kernel Size, Padding, Activation	32, 3, 'valid', 'relu'
	Layer Normalization	

D. DATASET

Our study investigates the effectiveness of our proposed method in real-world conditions of plant leaf disease classification. Accordingly, the RoCoLe dataset [11] with these properties was used. RoCoLe stands for Robusta Coffee Leaf, and it is a dataset collected in Ecuador designed for training,

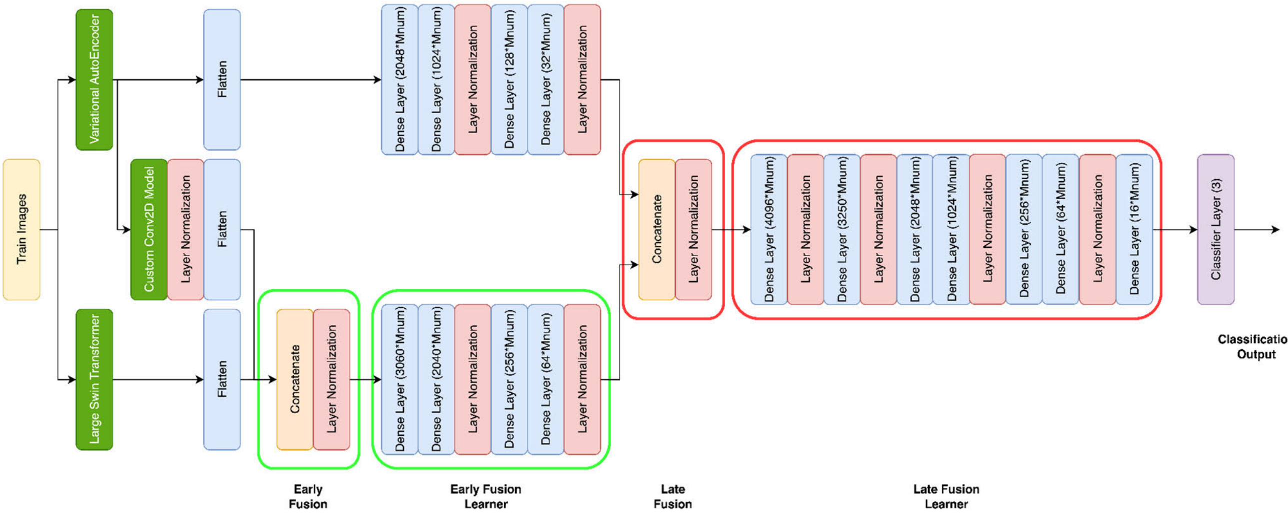

FIGURE 3. The architecture of hybrid fusion of VAE-CNN and swin-transformer.

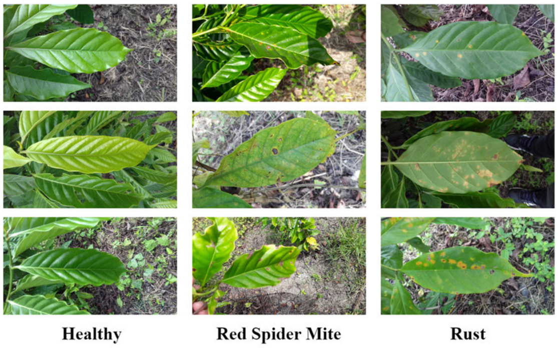

FIGURE 4. The example of RoCoLe dataset.

testing, and validating machine learning algorithms in binary or multi-class tasks. There are 1,560 images in the dataset provides three images for each 390 coffee plants. It consists of the front and back sides of the plants at different health states and with various diseases, including rust and red spider mites. The dataset also provides the severity level of the leaf infected. The images were captured using a 5-MP smartphone camera from an approximate distance of 200-300 mm from the object in various lighting and background conditions. As these were taken in real-world conditions, lighting and humidity may vary depending on the weather conditions, whereas the background may contain weeds or other plants. These conditions provide a representative sample of real-world conditions for the coffee leaf plants. Figure 4 shows the example of each class.

To maintain the real-world image conditions, no image preprocessing was applied before feeding into the model. By doing so, we could assess the model's capability on uncontrolled, real-world images containing various noises and interferences. Instead, data augmentation was applied to prevent overfitting from these noises and allow better generalization. The data augmentation in this study were pixel rescaling, horizontal and vertical flips, height and width shifting, and a maximum of 20-degree rotation. None of these alter the image quality. Afterward, the dataset is divided into train and testing sets following the k-fold cross-validation method. This method was selected instead of the three-way split (train, validation, and test), as the number of images in the dataset is limited. We opted to use 5-fold to split our dataset, where four parts are used for training the model, and one unseen part is used for testing. The dataset parts were uniquely iterated for training and testing the model. Metrics obtained from evaluating the model are based on the unseen testing dataset.

IV. EXPERIMENT AND RESULTS

In order to develop an optimized deep learning model, it needs to be trained using an optimizer and loss function. Two distinct models were developed, employing different training optimizers and loss function parameters. All model training was done for 50 epochs with the label-smoothened categorical cross-entropy loss function. We limit the epochs to 50 as our preliminary exploration discovered that 100 epochs and more did not impact performance significantly. Another acquired benefit of this is the reduced model training time. This loss function can be written as an equation in Equation 1, where y_i is the probability distribution of a prediction and $y_i{}'$ is the actual probability distribution. Furthermore, $\in$ controls the smoothing factor, while K is the number of prediction classes for a particular learning task. Except for the 'normal' MobileNetV3 and Swin Transformer model with a 0.3 smoothing factor, all other models were trained with a smoothing factor of 0.1. For both the MobileNetV3 and Swin Transformer, the Adam optimizer was utilized with a base learning rate of 0.00001. In contrast, the VAE-CNN and Swin Transformer model used a stochastic gradient descent (SGD) optimizer with a smaller base learning rate of 0.000001,

TABLE 3. The comparison of hybrid feature fusion models.

Method	Accuracy (%)	Precision (%)	Recall (%)	F_1-Score (%)
VAE-CNN + Swin-Transformer	83.97	83.14	83.97	82.96
Mini MobileNetV3 + Swin-Transformer	83.01	81.59	83.01	81.42
MobileNetV3 + Swin Transformer	**84.29**	**84.67**	**84.29**	**83.64**

a momentum of 0.99, and a decay rate based on the quotient between the base learning rate and the number of epochs. All of these hyperparameters were determined to be the best through a preliminary hyperparameter selection experiment based on several number of learning rate, SGD optimizer, and momentum. As a result, we limit our proposed methods to these hyperparameter selection results.

$$L_{y'} = -\sum_i \left(y_i'(1- \in) + \tfrac{\in}{K}\right) \log(y_i) \tag{1}$$

As deep learning models are prone to overfitting when trained with fewer data, we employed 3 (three) methods towards the data, the model, and the evaluation in our experiments to reduce the effects. Data augmentation was done on the dataset to increase the image variations prior to training. Our proposed methods used layer normalization to redistribute the weights in a particular layer across all features. It results in less overfitting and faster training time. Lastly, our evaluation utilizes the 5-fold cross-validation. It produces cross-validated metrics that are the result of averaging across all 5 runs, each trained on different dataset splits.

Evaluation is a crucial part of analyzing our developed models, and it is done based on several key metrics. These metrics include accuracy, precision, recall, and F_1-score. Accuracy is the ratio of correct prediction labels to the total number of predictions made across all labels. In the interest of discovering the performance of positively predicted labels, precision and recall are used. Precision is calculated by the number of predicted true positives divided by the total number of predicted positives. On the other hand, recall is the ratio of correct positive predicted labels to the total number of positive labels. These equations can be observed in Equations 2, 3, and 4. F_1-score is also the harmonic mean between the precision and accuracy, which could be calculated as such in Equation 5. As previously mentioned in Section III-D, all evaluation metrics were obtained on the unseen, testing part of the 5-fold cross-validation. Accordingly, comparisons done further down this section would be based on these critical metrics.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{2}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

$$F_1\text{-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{5}$$

In our experiment, several models were developed to solve the real-world condition of coffee leaf disease classification. Table 3 shows of our proposed hybrid feature fusion models. As the key to the feature fusion technique is to extract complimentary features and create a unified vector, a combination of CNN and Transformer was primarily used in the experiment. CNN and Transformers have different focuses when it comes to extracting relevant features. The former primarily extract from local pixels, whereas the latter could leverage the attention mechanism to capture more distant features. Further considerations when selecting the pre-trained models for feature extraction is the resulting large feature vector. Hence, the MobileNetV3 was selected as the CNN. Vision transformers struggle to create inductive biases with small datasets, as experienced in the ViT and DeiT. As a result, Swin Transformer was selected. Initially, a 'mini' fusion version was developed with fewer units in the dense layers. It achieved a satisfactory result with 83.01% accuracy, 81.59% precision, 83.01% recall, and an F_1-score of 81.42%.

Since real-world condition images contain backgrounds that may be considered as noise in the classification subject, the intuition to incorporate VAE is to leverage the properties of an autoencoder that could perform non-linear PCA while generating novel images to enrich the training dataset. This incorporation would prove beneficial, as seen from the performance improvements over the Mini MobileNetV3 + Swin Transformer model. This model was able to increase the accuracy of the classification by almost 1%. However, this was not explored further due to the large memory requirement. This was also the reason why the CNN after the VAE modality was not a pre-trained vision model. Instead, a simple optimized CNN model was selected. Subsequently, this also prompted an investigation into improving the later dense layers of the MobileNetV3 + Swin Transformer model. By doubling the units of the later dense layers, classification performance increased by 1.28% over the mini model. Precision also rose to 84.67% and recall to 84.29%. The F_1-score of the model was improved by 2.22% to 83.64%. With relatively similar memory usage, this model could outperform the VAE-CNN + Swin Transformer model.

Based on our experiment, Figure 5 depicts the training and validation accuracy and training and validation loss of hybrid feature fusion with MobileNetV3 and Swin Transformer.

TABLE 4. The comparison of MobileNetV3 + swin transformer to existing models.

Method	Accuracy (%)	Precision (%)	Recall (%)	F_1-Score (%)	Training Time for 50 epochs (s)
EfficientNet B7 [39]	77.56	76.68	77.56	76.63	0:43:12
MobileNetV2 [8]	67.95	65.24	67.95	65.12	0:42:15
MobileNetV3Large [29]	80.26	79.52	80.26	79.23	0:42:90
ResNet152 [42]	75.96	74.67	75.96	74.31	0:42:45
SWIN Transformer [32]	72.25	71.46	72.24	71.23	**0:42:10**
Vision Transformer [30]	57.44	53.37	57.44	53.34	0:43:44
MobileNetV3 + Swin Transformer	**84.29**	**84.67**	**84.29**	**83.64**	1:57:19

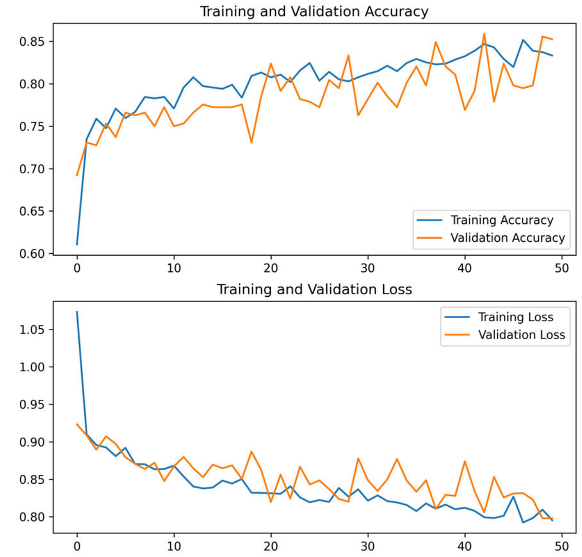

FIGURE 5. The training and validation accuracy and training and validation loss of MobileNetV3 + swin Transforme.

The x-axis represents the epoch, and the y-axis shows the accuracy. The graph shows that the training and validation accuracy performs an increased trend as the epoch increases. In the meantime, the training and validation loss tends to decline and reaching convergence. The narrow gap between training and validation accuracy represents the consistent model. We show the confusion matrix of our proposed method in Figure 6. The diagonal line, from top left to bottom right, in Figure 6 shows the correct classification of coffee leaf diseases. It can be seen that our proposed method correctly classified more of the leaf diseases rather than misclassifying them. With only four misclassified leaves in place of the healthy leaf, our hybrid feature fusion correctly classified 138 leaves. The red spider mite leaf could be predicted in 19 of 36 leaves accurately. Lastly, the proposed method could correctly classify 106 of 134 rust leaves.

As shown in Table 4, our hybrid feature fusion with MobileNetV3 and Swin Transformer achieves an accuracy of 84.29%, a precision of 84.67%, a recall of 84.29%, and an F_1-score of 83.64%. This result is better than conventional methods, such as MobileNetV3 [29] and Swin Transformer [32]. MobileNetV3Large obtains an accuracy of 80.26%, 4.41% lower than our method. Furthermore, Swin Transformer only achieves 72.25 percent accuracy, 12.04 percent less than our proposed method. These results further support our claim that the fusion of Swin Transformer together with a complimentary CNN architecture, and vice

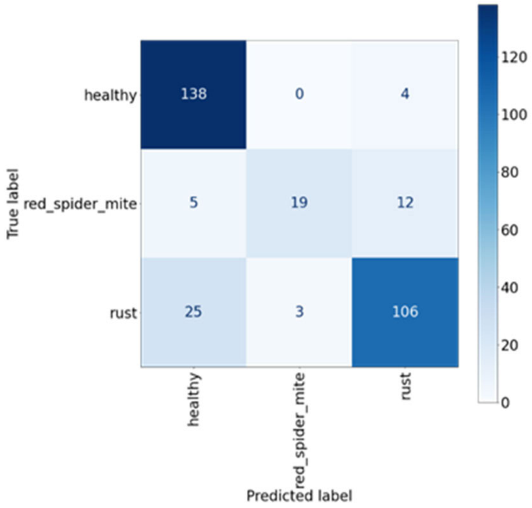

FIGURE 6. The confusion matrix of MobileNetV3 + swin transformer.

TABLE 5. The comparison of MobileNetV3 + Swin-Transformer to benchmark models.

CNN Models (Real-World Conditions)	Accuracy (%)
ResNet50 [17]	67.40
VGG16 [17]	79.83
Naive Approach [15]	75
MobileNetV3 + Swin-Transformer	**84.29**

versa, does extract differently focused features from the same image and improves the combined feature map for better classification accuracy. Unfortunately, Vision Transformer performs the worst by achieving an accuracy of 57.44%, a precision of 53.37%, a recall of 57.44%, and an F_1-score of 53.34%. With the highest performance of our proposed method, it suffers the longer training time required to train the model. It takes almost 2 hours to train the proposed method, twice longer than the conventional method, which takes around 40 minutes. It is evident since the hybrid feature model requires training two models simultaneously. Overall, the hybrid feature fusion of MobileNetV3 and Swin Transformer performs better than other conventional pre-trained models in coffee leaf disease classification in terms of accuracy, precision, recall F_1-score.

While there is a limited number of research done on real-world condition classification of Robusta Coffee Leaf Diseases, it was observed that classification accuracy suffered greatly when the same models were applied to real-world images [15], [17]. From a practical point of view, models trained on real-world conditions are desirable over laboratory-conditioned images. It allows classification without plucking out the leaves and harming the plants. However, the difficulty in producing satisfactory results when using real-world condition images may be one of the reasons very few studies strive to tackle this problem. Laboratory conditions would significantly inflate the classifier's performances and are not suitable comparators for our purpose. Past studies using real-world condition images, i.e., the RoCoLe dataset [11], have primarily used existing pre-trained models such as the ResNet50 and VGG16. The accuracy of these models was less superior to our proposed approach by around 17.5% and 4.46%, respectively.

V. CONCLUSION

We have explored a new automatic coffee leaf disease classification using hybrid feature fusion. We select model extraction from MobileNetV3, Swin Transformer, and variational autoencoder (VAE). The selected method combines the feature extraction capabilities of MobileNetV3, which is well-known for its efficient and lightweight feature extraction capabilities, and Swin Transformer, which has been reported to improve performance on many image classification tasks. Furthermore, the extracted features are learned using two-hybrid feature fusion methods: early- and late-feature fusion. The learned features are discriminated against using a fully connected layer to detect the diseases. The proposed method is evaluated using a public Robusta coffee Leaf (RoCoLe) dataset by measuring the performance metrics, including accuracy, precision, recall, and F_1-score. Compared to traditional methods, the hybrid feature fusion of MobileNetV3 and Swin Transformer is consistent. It performs better than individual conventional pre-trained models and other hybrid methods for coffee leaf diseases, with an overall testing accuracy of 84.29%.

In the future, the implementation of hybrid feature fusion using hardware accelerators such as Intel Jetson Nano is advised so that farmers in remote areas can utilize the detection system using hand-carry hardware. In addition, the application of hybrid feature fusion to other real-world datasets is necessary to generalize the hybrid feature fusion. Furthermore, the analysis of disease level for plant leaf disease is essential so that the measures are more effective.

REFERENCES

[1] E. B. Paulos and M. M. Woldeyohannis, "Detection and classification of coffee leaf disease using deep learning," in *Proc. Int. Conf. Inf. Commun. Technol. Develop. Afr. (ICT4DA)*, Nov. 2022, pp. 1–6.

[2] E. Gichuru, G. Alwora, J. Gimase, and C. Kathurima, "Coffee leaf rust *(Hemileia vastatrix)* in Kenya—A review," *Agronomy*, vol. 11, no. 12, p. 2590, Dec. 2021.

[3] H. Al Hiary, S. B. Ahmad, M. Reyalat, M. Braik, and Z. ALRahamneh, "Fast and accurate detection and classification of plant diseases," *Int. J. Comput. Appl.*, vol. 17, no. 1, pp. 31–38, Mar. 2011.

[4] E. Omrani, B. Khoshnevisan, S. Shamshirband, H. Saboohi, N. B. Anuar, and M. H. N. M. Nasir, "Potential of radial basis function-based support vector regression for apple disease detection," *Measurement*, vol. 55, pp. 512–519, Sep. 2014.

[5] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," in *The Handbook of Brain Theory and Neural Networks*, A. A. Michael, Ed. Cambridge, MA, USA: MIT Press, 1998, pp. 255–258.

[6] J. Chen, J. Chen, D. Zhang, Y. Sun, and Y. A. Nanehkaran, "Using deep transfer learning for image-based plant disease identification," *Comput. Electron. Agricult.*, vol. 173, Jun. 2020, Art. no. 105393.

[7] H. Amin, A. Darwish, A. E. Hassanien, and M. Soliman, "End-to-end deep learning model for corn leaf disease classification," *IEEE Access*, vol. 10, pp. 31103–31115, 2022.

[8] E. L. Da Rocha, L. Rodrigues, and J. F. Mari, "Maize leaf disease classification using convolutional neural networks and hyperparameter optimization," in *Proc. Anais do 16th Workshop de Visão Computacional (WVC)*, Oct. 2020, pp. 104–110.

[9] T. M. Antico, L. F. R. Moreira, and R. Moreira, "Evaluating the potential of federated learning for maize leaf disease prediction," in *Proc. Anais do 19th Encontro Nacional de Inteligência Artif Computacional (ENIAC)*, Nov. 2022, pp. 282–293.

[10] D. P. Hughes and M. Salathe, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," 2015, *arXiv:1511.08060*.

[11] J. Parraga-Alava, K. Cusme, A. Loor, and E. Santander, "RoCoLe: A robusta coffee leaf images dataset for evaluation of machine learning based methods in plant diseases recognition," *Data Brief*, vol. 25, Aug. 2019, Art. no. 104414.

[12] J. Parraga-Alava, R. Alcivar-Cevallos, J. M. Carrillo, M. Castro, S. Avellán, A. Loor, and F. Mendoza, "LeLePhid: An image dataset for aphid detection and infestation severity on lemon leaves," *Data*, vol. 6, no. 5, p. 51, May 2021.

[13] J. Jepkoech, D. M. Mugo, B. K. Kenduiywo, and E. C. Too, "Arabica coffee leaf images dataset for coffee leaf disease detection and classification," *Data Brief*, vol. 36, Jun. 2021, Art. no. 107142.

[14] A. Pal and V. Kumar, "AgriDet: Plant leaf disease severity classification using agriculture detection framework," *Eng. Appl. Artif. Intell.*, vol. 119, Mar. 2023, Art. no. 105754.

[15] M. Yebasse, B. Shimelis, H. Warku, J. Ko, and K. J. Cheoi, "Coffee disease visualization and classification," *Plants*, vol. 10, no. 6, p. 1257, Jun. 2021.

[16] R. Mahum, H. Munir, Z.-U.-N. Mughal, M. Awais, F. S. Khan, M. Saqlain, S. Mahamad, and I. Tlili, "A novel framework for potato leaf disease detection using an efficient deep learning model," *Human Ecological Risk Assessment, Int. J.*, vol. 29, no. 2, pp. 303–326, Feb. 2023.

[17] G. Fenu and F. M. Malloci, "Evaluating impacts between laboratory and field-collected datasets for plant disease classification," *Agronomy*, vol. 12, no. 10, p. 2359, Sep. 2022.

[18] S. Ahmed, M. B. Hasan, T. Ahmed, Md. R. K. Sony, and Md. H. Kabir, "Less is more: Lighter and faster deep neural architecture for tomato leaf disease classification," *IEEE Access*, vol. 10, pp. 68868–68884, 2022.

[19] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," 2014, *arXiv:1409.0575*.

[20] L. Yang, X. Yu, S. Zhang, H. Long, H. Zhang, S. Xu, and Y. Liao, "GoogLeNet based on residual network and attention mechanism identification of Rice leaf diseases," *Comput. Electron. Agricult.*, vol. 204, Jan. 2023, Art. no. 107543.

[21] S. Yu, L. Xie, and Q. Huang, "Inception convolutional vision transformers for plant disease identification," *Internet Things*, vol. 21, Apr. 2023, Art. no. 100650.

[22] C. Cheong Took and D. Mandic, "Weight sharing for LMS algorithms: Convolutional neural networks inspired multichannel adaptive filtering," *Digit. Signal Process.*, vol. 127, Jul. 2022, Art. no. 103580.

[23] M. Faisal, J. Leu, C. Avian, S. W. Prakosa, and M. Köppen, "DFNet: Dense fusion convolution neural network for plant leaf disease classification," *Agronomy J.*, Apr. 2023.

[24] X. Lu, X. Duan, X. Mao, Y. Li, and X. Zhang, "Feature extraction and fusion using deep convolutional neural networks for face detection," *Math. Problems Eng.*, vol. 2017, pp. 1–9, Jan. 2017.

[25] M. Prabu and B. J. Chelliah, "An intelligent approach using boosted support vector machine based arithmetic optimization algorithm for accurate detection of plant leaf disease," *Pattern Anal. Appl.*, vol. 26, no. 1, pp. 367–379, Feb. 2023.

[26] S. U. Rahman, F. Alam, N. Ahmad, and S. Arshad, "Image processing based system for the detection, identification and treatment of tomato leaf diseases.," *Multimedia Tools Appl.*, vol. 82, no. 6, pp. 9431–9445, Mar. 2023.

[27] K. Roy, S. S. Chaudhuri, J. Frnda, S. Bandopadhyay, I. J. Ray, S. Banerjee, and J. Nedoma, "Detection of tomato leaf diseases for agro-based industries using novel PCA DeepNet," *IEEE Access*, vol. 11, pp. 14983–15001, 2023.

[28] Y. Wu, X. Feng, and G. Chen, "Plant leaf diseases fine-grained categorization using convolutional neural networks," *IEEE Access*, vol. 10, pp. 41087–41096, 2022.

[29] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, "Searching for MobileNetV3," 2019, *arXiv:1905.02244*.

[30] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16 × 16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[31] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," 2020, *arXiv:2012.12877*.

[32] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," 2021, *arXiv:2103.14030*.

[33] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.

[34] J. Chung, K. Kastner, L. Dinh, K. Goel, A. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," 2015, *arXiv:1506.02216*.

[35] G. Barnum, S. Talukder, and Y. Yue, "On the benefits of early fusion in multimodal representation learning," 2020, *arXiv:2011.07191*.

[36] V. Kotu and B. Deshpande, "Data Mining Process," in *Predictive Analytics and Data Mining*. Amsterdam, The Netherlands: Elsevier, 2015, pp. 17–36.

[37] S. R. Stahlschmidt, B. Ulfenborg, and J. Synnergren, "Multimodal deep learning for biomedical data fusion: A review," *Briefings Bioinf.*, vol. 23, no. 2, Mar. 2022, Art. no. bbab569.

[38] H. R. Vaezi Joze, A. Shaban, M. L. Iuzzolino, and K. Koishida, "MMTM: Multimodal transfer module for CNN fusion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 13286–13296.

[39] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," 2019, *arXiv:1905.11946*.

[40] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[41] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," 2018, *arXiv:1801.04381*.

[42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015, *arXiv:1512.03385*.

MUHAMAD FAISAL received the B.E. degree in electrical engineering from Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia, in 2013, and the M.Sc. degree from the Graduate Institute of Automation and Control, National Taiwan University of Science and Technology (NTUST), Taipei, Taiwan, in 2016, where he is currently pursuing the Ph.D. degree with the Department of Electronic and Computer Engineering. His current research interests include intelligent control systems, deep learning, and model compression in deep learning. He received the NTUST Scholarship Award.

JENQ-SHIOU LEU (Senior Member, IEEE) received the B.S. degree in mathematics and the M.S. degree in computer science and information engineering from the National Taiwan University, Taipei, Taiwan, in 1991 and 1993, respectively, and the Ph.D. degree in computer science (on a part-time basis) from the National Tsing Hua University, Hsinchu, Taiwan, in 2006. He was a Research and Development Engineer with Rising Star Technology, Taiwan, from 1995 to 1997. He was an Assistant Manager with Mobitai Communications and Taiwan Mobile, from 1997 to 2007. In 2007, he joined the Department of Electronic and Computer Engineering, National Taiwan University of Science and Technology, Taipei, as an Assistant Professor, where he was an Associate Professor, from February 2011 to January 2014, and has been a Professor, since February 2014. His research interests include heterogeneous networks and mobile services over heterogeneous networks.

JEREMIE T. DARMAWAN is currently pursuing the B.Sc. degree in bioinformatics from the Indonesia International Institute for Life Sciences (i3L). He was a Research Intern with the MIT Laboratory, National Taiwan University of Science and Technology (NTUST), Taiwan, under Prof. Jenq-Shiou Leu. He received the i3L School Report Scholarship, which partially covered his bachelor's study with i3L. During the bachelor's degree, he has completed several internships and research projects. He was a recipient of the TEEP Scholarship, from Fall 2022 to Winter 2023.