IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 10 April 2023, accepted 8 May 2023, date of publication 16 May 2023, date of current version 26 May 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3276763

RESEARCH ARTICLE

Detection of Basal Stem Rot Disease Using Deep Learning

YU HONG HAW[1], YAN CHAI HUM[2], (Senior Member, IEEE), JOON HUANG CHUAH[3], (Senior Member, IEEE), WINGATES VOON[2], (Member, IEEE), SITI KHAIRUNNIZA-BEJO[4], NUR AZUAN HUSIN[4], POR LIP YEE[5], (Senior Member, IEEE), AND KHIN WEE LAI[1], (Senior Member, IEEE)

[1]Department of Biomedical Engineering, Universiti Malaya, Kuala Lumpur 50603, Malaysia
[2]Department of Mechatronics and Biomedical Engineering, Lee Kong Chian Faculty of Engineering and Science, Universiti Tunku Abdul Rahman, Kajang, Selangor 43000, Malaysia
[3]Department of Electrical Engineering, Faculty of Engineering, Universiti Malaya, Kuala Lumpur 50603, Malaysia
[4]Department of Biological and Agricultural Engineering, Faculty of Engineering, Universiti Putra Malaysia, Serdang, Selangor 43400, Malaysia
[5]Department of Computer System and Technology, Faculty of Computer Science and Information Technology, Universiti Malaya, Kuala Lumpur 50603, Malaysia

Corresponding author: Khin Wee Lai (lai.khinwee@um.edu.my)

This work was supported by the Fundamental Research Grant Scheme (FRGS), Ministry of Higher Education, Malaysia and Universiti Malaya under Project FRGS/1/2019/TK04/UM/01/2.

ABSTRACT Palm oil industry is an important economic resource for Malaysia. However, an oil palm tree disease called Basal Stem Rot has impeded the production of palm oil, which caused significant economic loss at the same time. The oil palm tree disease is caused by a fungus known as *Ganoderma Boninense*. Infected trees often have little to no symptoms during early stage of infection, which made early detection difficult. Early disease detection is necessary to allow early sanitization and disease control efforts. Using Terrestrial Laser Scanning technology, 88 grey-distribution canopy images of oil palm tree were obtained. The images were pre-processed and augmented before being used for training and testing of the deep learning models. The capabilities of the Convolution Neural Network deep learning models in the classification of dataset into healthy and non-healthy class were tested and the best performing model was identified based on the Macro-F1 score. Fine-tuned DenseNet121 model was the best performing model, recorded a Macro F1-score of 0.798. It was also noted that Baseline model showed a relatively remarkable macro-F1 score of 0.747, which was better than all the feature extractor models and some of the fine-tuned models. However, fine-tuned models suffered from model overfitting due to dataset limitations. For future work, it is recommended to increase the sample size and utilize other CNN architectures to improve the model performance and progress towards detecting Basal Stem Rot at the early stage of infection by classifying sample images into multiple classes.

INDEX TERMS Basal stem rot, convolutional neural network, deep learning, Ganoderma boninense, oil palm, terrestrial laser scanning.

I. INTRODUCTION

Oil palm (*Elaeis guineensis Jacq.*) is an important agricultural produce in several developing countries in South East Asia region [1], [2]. Globally, the palm oil which is produced from oil palm tree makes up about one-third of the world's vegetable oil and fat supply, with 85% of the palm oil is produced in Malaysia and Indonesia [2], [3]. Oil palm is commonly used as cooking oil. About three-quarters of the global palm oil supply is used for food while almost all of the rest is used for industrial purpose [4].

For Malaysia, palm oil is one of the largest contributors to the country's economy, and the industry plays a crucial role in elevating the socio-economic development of the country.

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

As of 2020, the country as a whole has more than 5 million hectares of oil palm plantations [4], [5]. However, since year 2009, a yield decline was observed in the oil palm industry in Malaysia. Several factors played a role in the decline of production, including plant disease, biodiversity, the age of palm tree and etc [4]. Infectious disease such as the basal stem rot (BSR) have caused a significant reduction in the yearly yield of palm oil [1].

BSR is an oil palm tree disease that can be found in Malaysia and Indonesia [6]. BSR is caused by the white rot fungus, also commonly known as *Ganoderma* [7]. There are different species of *Ganoderma* that can cause the BSR disease in oil palm trees, but the *Ganoderma boninense (G. boninense)* species is by far the most common *Ganoderma* fungus. *G. boninense* spreads and colonizes other oil palm trees through the roots, shoots, stems and rotting stem tissues [7], making the fungus a soil-borne pathogen. The fungus is also Necrotrophic, which means that the fungus is unnoticeable and not easily detectable during the early stages of infection [1].

In the early stage of infection, the fungus begins to infect and break down the cell wall of the infected oil palm tree while taking in key nutrients from the infected tree. At this stage, the tree shows little to no symptoms of infection. As infection stage progress the tree leaves will show symptoms of chlorosis, and the symptoms of BSR will show at the shoot of the infected oil palm tree. At this stage, new tree leaves will form the spears, remain unopened while being fully elongated. Complete colonization of *G. boninense* causes a restriction in water intake which leads to the drooping effect at the lower leaves and the tree gives a skirt like appearance. At full colonization, the color of the leaves begin to turn yellow. The leaves will begin to dieback from the tip of the leaves. During later stages of BSR infection the color of the tree base turns black since there is a severe degradation that occurs within the stem. Basidiocarps is often seen at the base of the plant during late stage of infection. At the end of the infection progression, the crown of the oil palm falls off and the foundation of the tree becomes weakened and eventually led to the fracture, collapse and death of the oil palm tree [1], [7].

The impact of *G. boninense* is serious on an infected tree. The entire economic life of an oil palm tree is around 25 years. However, once infected, the oil palm tree will typically die within 2 to 3 years. An infected tree is also expected to have 50 to 80% yield reduction over the tree's lifetime since BSR infection causes weight loss at the fruit branches [1], [3], [8].

Nationally, *G. boninense* is also spreading rapidly. From an estimated infection rate of 1.5% in year 1995, *Ganoderma boninense* has infected about 7.4% of the oil palm tree population in year 2017 [3]. In Malaysia, the spread of BSR caused an estimated economic loss of RM 225 million per annum [9]. Considering the speed of infection and the potential economic loss, it is vitally important to control the spread of BSR especially at the early stages of infection.

There are several methods that can be used to control the spread of BSR, namely physical control (such as cleaning, trenching and burning), chemical control (such as the use of fungicides and modified fertilizers), biocontrol methods, rapid degradation, and the use of genetic resistant materials. Physical control can be expensive due to the dependence on human labor. Burning is also outlawed by the Malaysian Government, under the Environmental Quality Act (EQA 1974) [1], [9], [10]. Chemical control may be effective but may be dangerous to the environment as the phenolic compounds involved are not well studied [1], [7]. As for biocontrol method, both chemical fungicides and biocontrol agents showed little effectiveness on field [8]. Rapid degradation method is a method that uses other fungi to increase the rate of deterioration of dead oil palm trees that has collapsed at the end of the infection progression. However, the environmental impact of this method is still not well researched [1]. Lastly, the use of genetic resistant material has the potential however the genes that are responsible for BSR resistance are still not isolated [1].

Considering the multiple options mentioned above, the most effective way to control the spread of this disease is still the physical method. However, this method requires human labor to physically clean and remove the infected plant [9]. Therefore, it is crucially important to develop a cost effective BSR detection method that can accurately detect BSR at the early stage of infection, to ensure sufficient sanitization efforts can be done to prevent the spread of BSR infection. The method should also be stable and work under any environmental conditions, while not requiring specialized tools or training [11].

The purpose of this research is to determine the possibility of *G. boninense* infection detection from the terrestrial laser scanning (TLS) cropped point cloud crown section data using deep learning, namely convolutional neural network (CNN).

II. LITERATURE REVIEW

In order to curb the spread of BSR infection, it is vitally important to accurately detect BSR infection early [12]. The default method of detecting BSR is by conducting visual inspection on the oil palm tree. However, this method is heavily dependent on human experience that is not transferable. This method is also susceptible to human error, at the same time technically challenging since oil palm tree that is at the early stage of infection do not present any visible symptoms [1], [13]. Manual detection method is also expensive since it is heavily dependent on human labor [4].

The laboratory methods were often cited to be reliable, especially Polymerase Chain Reaction technology (PCR). PCR has a very high detection accuracy and sensitivity; however the tests are very expensive as PCR requires expensive infrastructure and highly skilled professionals to conduct the PCR test following complicated protocols. Besides the cost, PCR lacks the flexibility needed for large scale testing as it does not allow on-site testing, since the DNA samples need to be transported to the laboratory for analysis. Furthermore, PCR is susceptible to sample contamination, making the test not suitable for BSR infection detection [1], [9].

Other laboratory tests available include Ganoderma Selection Medium (GSM) and Enzyme-linked Immunosorbent Assay Polyclonal Antibodies (ELISA-PABS). GSM has the potential for large scale use, however the testing reagent may not be safe for the environment. Reliability of GSM test is also yet to be proven [13]. ELISA-PABS showed greater promises in terms of accuracy and sensitivity when the test was compared against GSM. ELISA-PABS is also less technically demanding since the test is simpler while being compared to PCR. However, the test is still sophisticated and not suitable for large scale infection detection [13].

In summary, manual infection detection method is too dependent on human labor and the laboratory methods are too costly and technically demanding. Both manual and laboratory methods are not suitable for large scale BSR infection detection. Thus, digital detection methods may be better compared to both the manual and laboratory methods, since remote sensors perform better compared to the naked eyes, at the same time digital detection method may be operationally simpler compared to the laboratory methods [13].

Multiple methods of digital image and data collection were discussed extensively by Tee, C.A.T et al. (2021) [13] and Mohd Hilmi Tan, M.I.S et al. (2021) [12]. The image and data collected need to be pre-processed and analyzed using various methods in order to produce meaningful results that can be used to detect and classify BSR infection. The below section discusses the previous research involving the use of TLS and CNN in the detection and classification of BSR infection.

A. TERRESTRIAL LASER SCANNING (TLS)

Terrestrial Laser Scanning (TLS) is a ground based light detection and ranging (LiDAR) technology. The technology first shoots a pulsed laser light at an object, and the refection of the pulsed laser light will then be captured by a sensor. The reflected pulsed laser light is used to profile an object and capture the external shape of the object [13]. TLS offers a variety of advantages, which include easy setup, easy data acquisition, ease of operation for both professional and layperson, fast data acquisition, high image accuracy and high resolution. TLS also offers imaging from different perspectives and users are allowed to have repeatable views. TLS only requires equipment purchase therefore it does not require expensive laboratory set up, which shows that TLS can be considered more cost effective in a long run [14].

For application on oil palm TLS imaging, TLS laser energy is able to penetrate through canopy gaps and measure the tree canopy structure. Regular optical imagery is limited since the lower part of tree's canopy is often not visible [14]. It was hypothesized that healthy tree tends to have larger crown size and the canopies are often better developed as compared to infected trees [15]. This is true for oil palm tree as well, since *G. boninense* infection causes a restriction in water and nutrient intake of the oil palm tree. This leads to the underdevelopment of the leaves of the trees and the entire tree as a whole. The impact of stunted leaf growth causes a delay in new leaf development which leads to a reduction in front production, thus lead to a smaller crown size. The more severe an infection, the more pronounce the impact on the canopy of an oil palm. By studying the physical appearance of oil palm tree such as the canopy drooping effect, crown coverage area, and front features using TLS technology, an infected oil palm tree can be detected, and the infection severity of an oil palm tree can potentially be classified as well [14].

The below section listed in chronological orders a few research that were done that is related to BSR detection using TLS technology. The purpose of the below review is to identify which aspects of the research were done in order to identify the knowledge gap.

N.A. Husin et al. (2020) utilized TLS technology and analyzed the oil palm tree based on the observation that an infected tree has a skirt like appearance due to dehydration and nutrient deficiency that are caused by *G. boninense*. The TLS technology used for this research was by FARO technologies. The purpose of this research was to classify BSR disease using canopy stratification [14]. During this research, N.A. Husin et al. constructed 3D images of the oil palm tree using laser point data, the tree canopy was then stratified horizontally. The data was classified using six classification models, namely linear, two-factorial, quadratic, cubic, quartic and fifth models. The models were used in combinations, with less combinations developed for higher degree of polynomial function as the models were too complex. A total of 118 classification models were developed using a combination of parameters which include frond angle, frond number, crown pixel, canopy at 200cm from the top (S200) and canopy at 850cm from the top (S850). In conclusion, linear model with a combination of frond number, frond angle and S200 showed the greatest accuracy for healthy tree classification (100%) while showing a classification accuracy of 86.67% for healthy-unhealthy trees [14].

To emphasize on the analysis on the crown of tree, in a further study, N.A. Husin et al. (2020) first obtained the significant strata using t-test, which it was then used to develop a detection model to detect infected tree and healthy tree [16]. N.A. Husin et al. utilized multiple linear regression equations with the significant strata as the strata parameters to develop the coefficients model. Through the analysis of crown strata, it was concluded that healthy trees have higher crown densities than unhealthy tress starting from 240cm from the top (strata no.5) to the bottom. Prediction models using the strata parameters C650, C700, C800, and C850 (where the number next to C is the height from top measured in cm) were 92.5% accurate in the classification of healthy-unhealthy trees [16].

In the following further study, N.A. Husin et al. (2020) first utilized principal component analysis (PCA) to reduce the dimensionality of the dataset in order to improve the interpretability of the dataset while minimizing information loss [17]. This research utilized several machine learning techniques to classify the images. The models were evaluated to determine which model performed the best in image classification and BSR detection. The machine learning techniques used in this research were Decision Tree (DT), Discriminant Analysis (DA), Naïve Bayes (NB), Support Vector Machine (SVM), Nearest Neighbor (NN) and Ensemble Modelling (EM). The results showed that kernel naïve Bayes (KNB) model developed using the input parameters of the principal components (PCs) 1 and 2 had the best performance among 90 other models with a multiple level accuracy of 85% and a Kappa coefficient of 0.80. Furthermore, the combination of the two highest PC variance with the most weighted to frond number, frond angle, crown area, and C200 significantly contributed to the classification success. The model also could classify healthy and mildly infected trees with 100% accuracy. Therefore, it was concluded that using TLS data, machine learning approach was highly accurate in the prediction of BSR during early stage of infection [17].

To determine the suitable timeframe in monitoring the progress of BSR disease, N.A. Husin et al. (2021) continued the research using TLS [18]. The researchers observed the change of crown and frond metrics of oil palm tree with two different scan durations, namely 2 and 4 months after the first scan. The purpose of this research was to help develop long-term solutions and improve response speed in the treatment of infected oil palm tree. The results showed that crown strata that was 850cm from the top and the crown area (both collected over four months and second two months period) were the most suitable metrics to be used to distinguish infected and non-infected oil palm trees. At the point of this research, infection inspection practice occurred every 6 or 12 months. A faster cycle was recommended since BSR may cause the death of young oil palm trees within 12 months. A more frequent disease census may reduce and prevent crop destruction since detection could be done earlier and allow earlier intervention and treatment efforts [18].

From the review on the research papers related to TLS technology, in the detection of BSR infection, it was shown that TLS remains a reliable technology for oil palm tree data collection. It was found that all the studies used various classification methods, which include machine learning methods such as DT, DA, NB, SVM, NN, and EM. However, deep learning model such as Convolution Neural Network (CNN) was never implemented on TLS data in the detection of BSR infection.

B. CONVOLUTIONAL NEURAL NETWORK (CNN) RESEARCH IN BSR DETECTION

CNN is a type of neural network that is very effective in the field of computer vision, pattern recognition and classification. CNN is often regarded as the upgraded version of Artificial Neural Network (ANN). ANN is a neural network that imitates the human brain function in complex task completion and decision making. ANN consists of multilayer and back-propagation to enable the learning ability of computer in determining the nonlinear combinations. CNN has recently gained popularity in the field of computer vision and plant disease detection. The reason why CNN has shown increase in popularity is because of the automatic feature extraction from any given dataset. CNN contains an output layer, a hidden layer, multiple convolutional layers, pooling layers, fully connected layers, and normalization layers to automatically extract abstracted shallow and deep features of the input [13], [19]. To review the possibility of utilizing deep learning models on oil palm data, the section below explores the research papers that utilized neural network or deep learning in the detection of BSR or other plant disease through dataset generated by various sensors. The research below were listed in chronological order.

P. Ahmadi et al. (2017) researched on the possibility of using ANN on raw, first and second derivative spectroradiometer datasets. The purpose of this research was to discriminate and classify BSR infection especially during early stage of infection [20]. The research utilized visible and near-infrared (VIS-NIR) spectral reflectance data from tree trunk samples that were sent to lab after trunk drilling. The research found that using ANN, early detection of BSR was possible with a high level of accuracy, ranging from 83.35 to 100% [20].

Instead of relying on hyperspectral sensors, A.Y. Khaled et al. (2018) identified that the electrical properties of oil palm tree leaves could be used in the infection severity detection of BSR. SVM and ANN classifiers were used to determine the classification accuracies on BSR infection by analyzing the significant frequencies that were selected using generic algorithm (GA), random forest (RF) and SVM-feature selection (SVM-FS). It was observed that SVM classifier showed better performance in terms of accuracy as compared to ANN, with accuracies over 80% [21].

In a further study, P. Ahmadi et al. (2022) utilized an unmanned aerial vehicle (UAV) together with a near infrared sensor (NIR) to collect images of oil palm tree canopy. The images captured were analyzed and classified using ANN model. Training accuracy of 97.52% was achieved for healthy palm trees, however the testing accuracy dropped to 72.73%. For early infected palm trees, the training accuracy achieved was 88.50%, however the testing accuracy dropped to 57.14% [22].

C.C Lee et al. (2022) also utilized a hyperspectral sensor. The sensor was used in combination with an UAV. Multilayer Perceptron (MLP) were used to study the spectral features from oil palm tree of different infection severity. The images collected were red, green and blue (RGB) image and hyperspectral images. It was discovered that using MLP, the accuracy achieved was 86.67% whereas SVM and 1-dimensional CNN only achieved 66.67% and 73.33% accuracy respectively [23].

L.Z Yong et al. (2023) detected BSR in oil palm seedlings at an early stage of infection using NIR-hyperspectral images and three distinct types of deep learning architecture. Three models were used, including a 16-layer convolutional neural network (VGG16) model that was trained on the segmented images, as well as Mask RCNN and VGG16 models that were both trained on the original images. The best performance was achieved by VGG16 model, which had an F1 score of 91.72% when trained with the original images at 938 nm wavelength [24].

Based on the research reviewed, it was shown that most studies reviewed utilized ANN instead of CNN. CNN was only attempted on spectral data such as VIS-NIR and hyperspectral data. The potentials of using CNN on TLS dataset should be studied to observe if the detection accuracy of BSR infection can be improved as compared to other machine learning methods that have been explored.

As a reference, there are other plant disease research that utilized CNN. J.R. Xiao et al. (2020) showed the possibilities of detecting strawberry diseases with 98% accuracy, while applying CNN models on 1306 feature images [25]. P. Sharma et al. (2020) compared the application of CNN on full images (F-CNN) or segmented image (S-CNN) on around 17,000 tomato leaves images, and found that S-CNN showed a very remarkable 98.6% accuracy as compared to only 42.3% accuracy for F-CNN [26].

When presented with large data, CNN has the potential to provide high disease detection accuracy. With less data available, the accuracy of CNN tend to be much lower. Nguyen et al. (2021) showed that applying 2-dimensional CNN (2D-CNN) and 3-dimensional CNN (3D-CNN) on 40 hyperspectral images of grapevine only gave 71 to 75% training accuracy and around 50% testing accuracy [27].

To overcome the issue of limited dataset, data augmentation can be used to increase the number of labelled images artificially within the dataset. Example of data augmentation includes adjustment of rotation, brightness, contrast and sharpness of the images. Data augmentation has the potential to increase the accuracy of deep learning models, example given by G. Hu et al. (2018) that showed 7% accuracy improvement can be achieved by using data augmentation while applying CNN on a small dataset [28].

III. METHODOLOGY

A. OVERVIEW

This section is to provide an overview on how the dataset was implemented in the study, followed by the details of data pre-processing steps, model implementation, and experiment setups. All experimentations were conducted with the TensorFlow framework with Keras as the backend in the Google Colaboratory platform (2.30 GHz Intel®Xeon®CPU, up to 32 GB RAM, and NVIDIA P100 or T4 GPU).

Figure 1 depicts the overall methodology of the study. Stage 1 was the data collection step, which this study used secondary data obtained in [17] (refer Section B). After stage 1, the dataset was separated into train set $\mathrm{D_{tr}}$ and test set $\mathrm{D_{ts}}$ to ensure the test dataset was never seen by the deep learning model (refer Section C). At stage 2, the $\mathrm{D_{tr}}$ was augmented to increase the number of images artificially to form $\mathrm{D_{tr,aug}}$ (refer Section D). Next, from stage 3 to 4, $\mathrm{D_{tr,aug}}$ was subjected to stratified five-fold cross-validation (SFFCV) to evaluate the stability of the baseline model and the five selected ImageNet pre-trained CNNs (refer Section E to G). Depending on if the model was stable, finally at stage 5, the model was trained with the whole training dataset $\mathrm{D_{tr,aug}}$ and tested on the whole testing dataset $\mathrm{D_{ts}}$ (refer to Quantitative Evaluation, Section G).

B. DATA COLLECTION

The study used secondary data obtained in [17]. The images were collected using a TLS system called Faro Laser Scanning Focus 3D (FARO Technologies, Inc., Florida, USA). For this research, canopy was the area of interest. Upon completing the scans, all the scans were imported into SCENE software for processing and analysis. SCENE software harmonized the multiple TLS scans and synchronized the laser point data to create a cluster of point clouds and a complete 3D view of the oil palm tree. After the point cloud registration process, the scanned images went through single tree extraction and crown section image extraction to single out each tree per image.

After singling out each tree, the images were pre-processed using grey-level distribution. This method was a global thresholding method that relied on the grey value of the image with the aim of separating objects of interest in an image from the background. This method segregated the object from the background using the object's grey-level distribution.

This algorithm used the assumption that the image contains two classes of pixels following a bi-modal histogram, namely the foreground pixels and background pixels. The algorithm calculated the optimum threshold to discriminate the two classes, so their combined spread (intra-class variance) was minimal. This method was considered as simple and effective [14].

A total of 88 grey-level distribution images of the oil palm tree canopy were collected. The collected images comprised two major classes, namely healthy (H) and non-healthy (NH). The healthy class was also labelled as T0. The NH class could be further divided into three subclasses, namely mildly infected (T1), moderately infected (T2), and severely infected (T3). Table 1 illustrates the dataset composition.

The objective of this research was to detect the *G. boninense*, i.e. classifying the dataset to 2 separate classes, namely H class and NH class. For this objective, H class had 26 images, where NH class had 62 images (combination of T1, T2 and T3 subclasses, with 20, 21 and 21 images respectively). The measure to tackle the imbalance in the H and NH dataset was mentioned in section F of this chapter.

C. DATA PRE-PROCESSING

Data pre-processing is a process where the raw inputs are prepared into batches of processed data that will be made compatible with the classifier. First, the dataset, D_s was defined as $D_s = \{X_1, X_2, \ldots, X_N\}$ of N images, where an image, $X =$

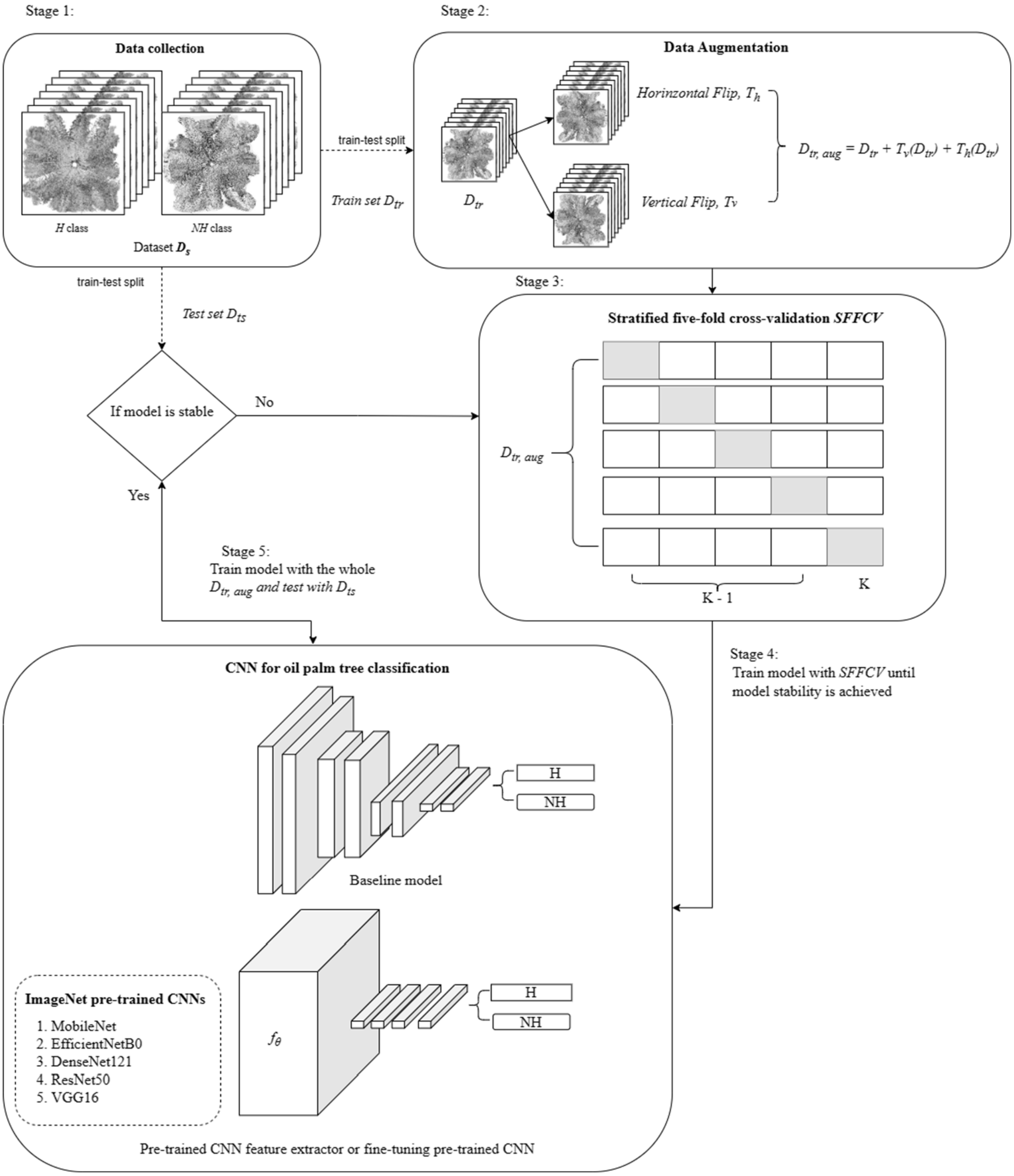

FIGURE 1. The overall flow of the research methodology.

(width, height, number of channels), was a two-dimensional set of pixels with three channels that indicate RGB color space. Each X was associated with the label space, where $Y(X) \in \{H, NH\}$. Subsequently, train-test split method was done on $\mathrm{D_s}$, where train set, $D_{tr} \approx 0.7(D_s)$, test set, $D_{ts} \approx$ (1 - 0.7)($\mathrm{D_s}$), and $D_{tr} \cap D_{ts} = \emptyset$. Table 2 illustrates the composition of D_s after the train-test split. The dataset split was done to ensure the test set was independent and was not used in hyperparameters fine-tuning. The test set was used for testing only.

D. DATA AUGMENTATION

It was observed that the D_{tr} induced model overfitting easily during model training during the preliminary experiments. was severely limited which caused the deep learning models to not perform well due to data limitations [29]. As a result, artificial transformation, $T \in \{T_v, T_h, T_i\}$ was introduced to D_{tr} to increase the number of training images, where T_v = vertical flip, and T_h = horizontal flip, forming the augmented train set, $D_{tr,aug} = D_{tr} + T_v(D_{tr}) + T_h(D_{tr})$. Table 3 illustrates the composition

TABLE 1. The composition distribution of the dataset.

Main classes	Healthy *H*	Non healthy *NH*			Total
Total	26	62			88
Sub-classes	***T0***	***T1***	***T2***	***T3***	**Total**
Total	26	20	21	21	88

TABLE 2. The distribution of D_s after the train-test split.

	H	*NH*	Total
D_{tr}	20	44	64
D_{ts}	6	18	24
Total	26	62	88

TABLE 3. The distribution of D_s after transformation T.

	H	*NH*	Total
$D_{tr,\ aug}$	60	132	192
D_{ts}	6	18	24
Total	66	150	216

of $D_{tr,aug}$ after $T(D_{tr})$. Figure 2 depicts T of a sample image.

During model training, real-time augmentation, $T_i \in \{T_{ir}, T_{ib}, T_{ir}\}$ was applied using the ImageDataGenerator class from TensorFlow Keras to the $D_{tr,aug}$, where T_{ir} = random rotation, T_{ib} = random brightness, and T_{iz} = random zoom. T_i augments X in $D_{tr,aug}$. The augmentation was done randomly and in real-time (with *seed* = 42 to ensure reproducibility) during the model training.

E. STRATIFIED FIVE-FOLD CROSS VALIDATION

After $D_{tr,aug}$ formation, stratified five-fold cross-validation (SFFCV) was performed on the $D_{tr,aug}$. K-fold stratified cross-validation technique divided $D_{tr,aug}$ into K subsets ($K = 5$), which the first K fold was taken as the validation set, while the remaining $(K - 1)$ folds were used as training set, at the same time ensured all classes had the same proportion akin to the original dataset. The average validation results were computed from results generated from each fold. Performing stratified cross-validation reduced result variability, ensured model stability, and provided a more comprehensive performance evaluation across the whole dataset.

Before model training, batches of preprocessed tensor image data were generated from $D_{tr,aug}$ with operations such as rescaling, resizing, batching, and shuffling. It was observed that image intensity values in D_s range between 0 and 255. Thus, these intensity values were normalized to values between 0 and 1. Furthermore, it was found that images in D_s vary in resolutions. Hence, the images in D_s were resized to 224 × 224. Table 4 depicts the operations details of the pre-processing steps.

TABLE 4. The operation details of the pre-processing steps.

Operation	Details
Rescale	1./255 *not performed on experiments with pre-trained models, used custom pre-processing function (see Appendix A)
Resize	(224, 224, 3)
Batch size	32
Shuffle	True

F. WEIGHTING TECHNIQUE

Referring to Table 3, it was observed that the dataset D_s was imbalanced. For an imbalanced dataset, classification model tends to be biased in inferencing the majority class. Therefore, class weighting technique was implemented to give the minority class a more significant weight in the model cost function while imposing a higher penalty on the minority class. With this penalty, the model was able to converge to minimize loss for the minority class [30]. The equation used to determine the class weight is defined as:

$$\textit{Class Weighting} = \frac{N}{N_c \times N_{sc}} \tag{1}$$

where N = total number of samples, N_C = number of classes, and N_{SC} = number of samples in each class.

G. MODEL IMPLEMENTATION

1) BASELINE MODEL

This section describes the baseline *CNN* model for the oil palm tree classification into H or NH class. Given an X from D_s as an input, the primary goal was to classify X into $Y(X) \in \{H, NH\}$. The baseline model consisted of three convolutional blocks and one fully connected layer for classification. The result generated by the baseline model were used as the standard for the study. Table 5 shows the structure of the baseline model.

2) TRANSFER LEARNING

Transfer learning is the process of improving a target predictive function $f_T(\cdot)$ by using associated information from a source domain, D_{source}, with its corresponding source task, T_{source}, where $D_{source} \neq D_{target}$ (target domain) or $T_{source} \neq T_{target}$ (target task) [31]. The first blocks of a CNN learn generic features while the final blocks of a CNN learn task-specific features. Therefore, transfer learning leverages the learned generic features from D_{source} and then relearns the

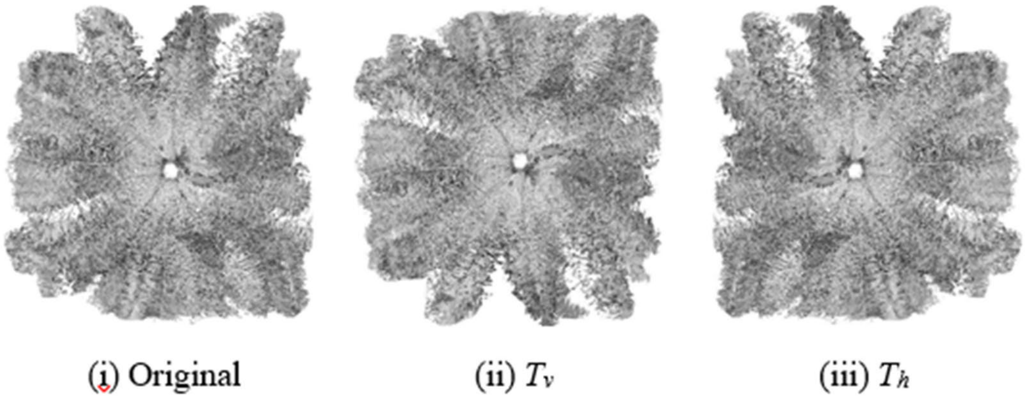

FIGURE 2. Examples of the original image (i) and the respective transformed image Tv (ii) and Th(iii).

TABLE 5. The structure of the proposed baseline model.

Block	Layer	Details
0	0	Input layer, shape = (224, 224, 3)
1	1	Convolutional layer [3,3], 32, strides = 1, padding = same
2		Batch normalisation layer
3		ReLU activation function
2	4	Convolutional layer [3,3], 64, strides = 2, padding = same
5		Batch normalisation layer
6		ReLU activation function
3	7	Convolutional layer [3,3], 64, strides = 2, padding = same
8		Batch normalisation layer
9		ReLU activation function
4	10	Global average pooling layer
Layer 11		Fully connected layer [1], sigmoid activation function

specific features of D_{target}. Transfer learning takes in what a model has already learned while only tuning the final classifier in order to classify the given custom dataset thus solving the problem involving the given dataset. Transfer learning allows the reduction of the time needed and the amount of data required for model training [32]. Essentially, transfer learning allows small datasets to be trained on CNNs with a lower risk of model overfitting.

TABLE 6. The selected five ImageNet pre-trained CNNs used for the study.

Architectures	Main contributions	Parameters (M)
MobileNet [33]	Depth-wise separable convolutions	4.2
EfficientNet B0 [34]	Compound scaling	5.3
DenseNet121 [35]	Dense connectivity	8.1
ResNet50 [36]	Residual learning	25.6
VGG16 [37]	Increasing depth convolutions	138.4

In this study, five pre-trained CNNs (see Table 6) in the ImageNet dataset (a dataset that comprises more than 1.2 million natural images with 1000 classes) were transferred to the palm tree binary classification task (H and NH classes). The reason why the 5 CNN's were selected was to test the dataset using diverse CNN architectures from different CNN architecture family, for example, the EfficientNets were comprised of B0-B7 models, where the ResNets were comprised of models from 18-152 layers and etc.. The purpose of testing multiple CNN architectures was to provide a general sense to which model architecture may perform better in such a specific application.

For the classification model, the model consists of a feature extractor, f_θ (parametrized by the model parameter θ) and a classifier, $C(\cdot|W)$ (parametrized by weight matrix $W \in \mathbb{R}^{d\times c}$). The classifier $C(\cdot|W)$ consists of two dropout layers, fully connected layers, with the last layer equipped with a

TABLE 7. Structures of the proposed model, a feature extractor, f_θ with a linear classifier $C(\cdot|W)$.

Layer Name	Block	Details
Input Layer	0	Shape = (224, 224, 3)
f_θ	1	Feature extractor, f_θ 2-dimensional global average pooling layer
$C(\cdot|W_b)$	2	Dropout layer with rate = 0.4 Fully connected layer [256], ReLU activation function Dropout layer with rate = 0.4 Fully connected layer [1], sigmoid activation function

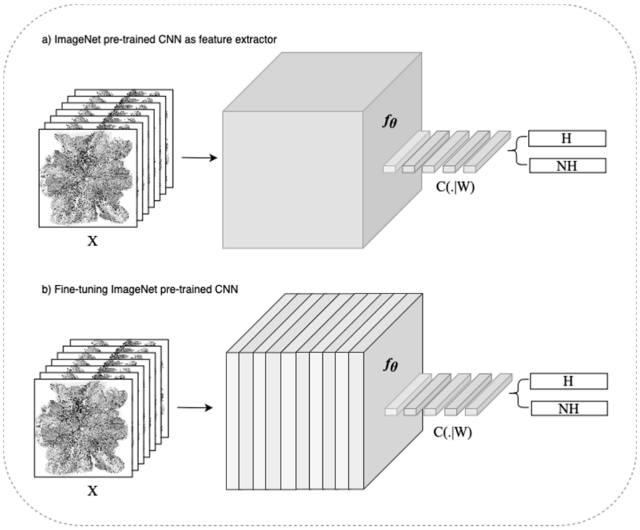
a) ImageNet pre-trained CNN as feature extractor
X
fθ
C(.|W)
H
NH
b) Fine-tuning ImageNet pre-trained CNN
X
fθ
C(.|W)
H
NH

FIGURE 3. The two configurations for the ImageNet pre-trained CNNs. For the first mode, fixed parameter θ in the f_θ and trained a new $C(\cdot|W)$ on Dtr, aug,. For the second mode, fine-tuned the parameter θ in the f_θ and trained a new $CC(\cdot|W)$ on Dtr, aug. (Cube shape as fixed parameter θ while layers of rectangular cuboids as trainable parameter).

sigmoid function σ (see Table 7). Two modes of experiments were conducted (see Figure 3 below):

(1) Fixed the parameter θ in the f_θ and trained a new $C(\cdot|W)$ on $D_{tr,aug}$, i.e. frozen weights.

(2) Fine-tuned the parameter θ in the f_θ and trained a new $C(\cdot|W)$ on $D_{tr,aug}$, i.e. fine-tuned weights.

The models were trained with $D_{tr,aug}$ by minimizing the binary cross-entropy loss, BCE_{loss} (see Equation 2), using Adam Optimizer to optimize the model parameters [38]. Furthermore, grid search was performed to determine the optimal learning rate and number of epochs for model training.

$$BCE_{loss}(\hat{y}, y) = -\left(\sum_{i=1}^{N} y_i log\hat{y}_i + (1-y_i)\log\left(1-\hat{y}_i\right)\right) \tag{2}$$

where $\mathrm{y_i}$ = ith label y of N classes, and $\hat{y}$ = ith element of model output $\hat{y}$.

```
Algorithm 1: Methodology
Input: Dataset $D_s$,

// split train set, $D_{tr}$
$D_{tr} \approx 0.7(D_s)$

// split test set, $D_{ts}$
$D_{ts} \approx (1 - 0.7)(D_s)$

// augment $D_{tr}$ to form $D_{tr,\ aug}$
$D_{tr,\ aug} = D_{tr} + T_v(D_{tr}) + T_h(D_{tr})$

// compute weight for each class
$w = Class\ Weighting\ (D_{tr,\ aug})$

// store validation results
$R_{val} = []$

// train model with SFFCV
for $K$, $(K - 1)$ in $SFFCV(\mathrm{D_{tr,\ aug}})$ do
    for epoch ∈ epochs do
        for $X$, $y$ batch ∈ $D_{tr,\ aug,\ (K-1)}$ do
            $\hat{y} = Model(X, \theta, w)$
            Compute $loss = BCE_{loss}\ (\hat{y}, y)$
            Compute the gradient, $\nabla_{loss}$ of the $\theta$ with respect to the $loss$
            Update $\theta = Adam(\nabla_{loss}, \theta)$
        end for
    end for
end for
// model evaluation with SFFCV
$\hat{y} = Model(\mathrm{D_{tr,\ aug,\ K}}, \theta)$
$result = Confusion\ Matrix\ (\hat{y}, y)$
Append $result$ into $\mathrm{R_{val}}$

// Once the model is optimized and stable
// train model with whole $\mathrm{D_{tr,\ aug}}$ and test with $\mathrm{D_{ts}}$
for epoch ∈ epochs do
    for $X$, $y$ batch ∈ $\mathrm{D_{tr,\ aug}}$ do
        $\hat{y} = Model(X, \theta, w)$
        Compute $loss = BCE_{loss}\ (\hat{y}, y)$
        Compute gradient, $\nabla_{loss}$ of the $\theta$ with respect to the $loss$
        Update $\theta = Adam(\nabla_{loss}, \theta)$
    end for
end for
// Model Evaluation
$\hat{y} = Model(\mathrm{D_{ts}}, \theta)$
$Test\ result = Confusion\ Matrix\ (\hat{y}, y)$
```

FIGURE 4. Algorithm 1: Methodology.

Algorithm 1 of Figure 4 provides a detailed description of the entire methodology. First, D_s was divided into D_{tr} and D_s via train-test split. Next, D_{tr} was transformed into $D_{tr,aug}$ where $D_{tr,aug} = D_{tr} + T_v(D_{tr}) + T_h(D_{tr})$. Subsequently, five-fold stratified cross-validation split was done to split $D_{tr,aug}$ into five folds. For each fold, batches of tensor inputs with $D_{tr,aug}$ were generated for model training. After instantiating the model, the model was fitted with the $D_{tr,aug}$ by minimizing the BCE_{loss} with Adam optimizer for 100 epochs. When the stability of the model reaches a satisfactory level through five-fold stratified cross-validation, the model was fitted with the whole $D_{tr,aug}$, and tested on D_{ts}. Finally, confusion matrix was done to evaluate the model performance metrics.

3) QUANTITATIVE EVALUATION

The model performance were evaluated comprehensively, using five performance metrics, namely accuracy, precision, recall, F1-score, and inference time, with macro-average technique. Given that true positive TP, true negative TN, false positive FP, and false negative FN, the precision is computed with TP divided by $(TP + FP)$. The recall is computed with TP divided $(TP + FN)$. Precision and recall were evaluated in order to have a comprehensive view over the model performance. Thus, the F1-score was utilized as the primary evaluation metric since F1-score was expressed as the harmonic mean of precision and recall. The purpose of using the F1-score is to ensure a more balanced summarization of model performance. The macro-average technique calculates each class metric independently and averages the results, ensuring that all classes were treated equally. The equations below depict the evaluation metrics.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{3}$$

$$Recall_{macro} = \frac{1}{|C|}\sum_{i=1}^{|C|}\frac{TP_i}{TP_i+FN_i} \tag{4}$$

$$Precision_{macro} = \frac{1}{|C|}\sum_{i=1}^{|C|}\frac{TP_i}{TP_i+FP_i} \tag{5}$$

$$F1score_{macro} = 2 * \frac{Precision_{macro} \times Recall_{macro}}{Precision_{macro}+Recall_{macro}} \tag{6}$$

$$Inference\ time\ (s) = \frac{T_{final}-T_{initial}}{N_s} \tag{7}$$

where C = number of classes, T_{final} = final inference time for X in D_{ts} or $D_{tr,aug,k}$, $T_{initial}$ = initial inference time for X in D_{ts} or $D_{tr,aug,k}$, and N_s = total number of X in D_{ts} or $D_{tr,aug,k}$.

IV. RESULTS

As mentioned in the previous chapter (Refer Quantitative Evaluation), the F1-score was utilized as the primary evaluation metric since F1-score was expressed as the harmonic mean of precision and recall. The research objective is to detect the infected oil palm tree from the TLS data. Table 8 illustrates the five-fold stratified cross-validation results, while Table 9 illustrates the test results of the Baseline model and the five pre-trained CNN models in two modes (feature extractor and fine-tuning) on the palm tree infection detection task (H class and NH class). According to the validation results (see Table 8), the fine-tuned ResNet50 topped other models in the classification task, achieving up to 0.886 macro F1-score with approximately 37 ms of inference time. For the test results (see Table 9), although the fine-tuned ResNet50 achieved a relatively high macro F1-score (0.778) compared to the other models, the fine-tuned DenseNet121 outperformed the fine-tuned ResNet50, achieved 0.798 macro F1-score with approximately 92 ms of inference time.

It was also noted that from the Macro F1 score, baseline model performed better than all feature extractor models and some fine-tuned models. This trend was also observed in Figure 6.

V. DISCUSSION

In this study, the application of CNN to classify oil palm tree images into H and NH classes were explored and studied. A plain vanilla CNN was developed as the baseline model and selected five state-of-the-art pre-trained CNNs (MobileNet, EfficientNet B0, VGG16, ResNet50, and DenseNet121) for the classification task.

A. COMPARISON BETWEEN DIFFERENT CNNS

As discussed within the methodology, the purpose of choosing the 5 CNN models mentioned in Table 6 was to explore multiple CNN architecture families in order to have a general sense as to which model architecture performed better under the specific application. The finding will allow opportunities to further research using deeper model when the research is scaled up when more data become available.

Among all of the models chosen for this study, ResNet and DenseNet were the top performers (see Figure 5 showing ResNet performed best during SFFCV while Figure 6 showing DenseNet performed best during the model testing stage). Figure 8 (page below) shows the confusion matrices for both ResNet during SFFCV and DenseNet during model testing. It was noteworthy that ResNet model during SFFCV has quite a high count of false negative, while DenseNet performed relatively well during testing phase, with low incidence of false positive and false negative. However, both false negative and false positive cases are not ideal since a false positive case would allow *G. Boninense* to spread continuously without being detected, whereas a false negative case will cause a waste in disease control resources.

It was hypothesized that ResNet and DenseNet performed well owing to their architectural novelty. Consider an image $\mathrm{x_o}$ that is fed forward through a CNN. The CNN comprises L layers, each of which adopts a non-linear transformation $H_\ell\ (\cdot)$, where ℓ indexes the layer. The output of the ℓ th layer is represented as x_ℓ. Conventional CNN connects the output of the ℓ th layer as input to the $(\ell + 1)$th layer [39], which yields the layer transition: $x_\ell = H_\ell(x_{\ell-1})$. ResNet (Residual Neural Network) were developed to address the vanishing gradient problem by adding a skip connection that bypasses the non- linear transformation with an identity function $x_\ell = H_\ell\ (x_{\ell-1}) + x_{\ell-1}$. It was observed that the gradient can flow directly through the identity function from later layers to the earlier layers in ResNet.

Likewise, DenseNet (Dense Convolutional Network) was developed to further optimize the data flow between layers. DenseNet introduced direct connections from any layer

TABLE 8. The SFFCV performance of the baseline model and five ImageNet pre-trained CNNs in terms of accuracy, macro precision, macro recall, macro F1-score and inference time. The results based on Macro F1-score suggested that the ResNet50 achieved the highest performance compared to other CNNs. The inference time listed are less than 0.1 second thus the differences are negligible.

Model	Accuracy	Macro Precision	Macro Recall	Macro F1-Score	Inference Time (s)
Baseline (LR = 0.0001)	0.734±0.070	0.714±0.073	0.725±0.056	0.709±0.066	0.006±0.003
Frozen ImageNet weights (learning rate = 0.0001)					
MobileNet	0.749±0.078	0.753±0.060	0.786±0.066	0.739±0.074	0.016±0.004
EfficientNetB0	0.766±0.112	0.755±0.110	0.793±0.126	0.753±0.116	0.040±0.007
VGG16	0.735±0.093	0.745±0.068	0.762±0.053	0.721±0.083	0.024±0.020
ResNet50	0.781±0.090	0.771±0.080	0.809±0.093	0.769±0.092	0.031±0.007
DenseNet121	0.771±0.055	0.781±0.041	0.820±0.040	0.764±0.053	0.062±0.013
Fine-tune ImageNet weights (learning rate = 0.00001)					
MobileNet	0.823±0.101	0.800±0.101	0.830±0.116	0.806±0.108	0.017±0.005
EfficientNetB0	0.792±0.089	0.771±0.088	0.803±0.094	0.776±0.093	0.047±0.006
VGG16	0.860±0.072	0.841±0.085	0.834±0.079	0.836±0.082	0.024±0.020
ResNet50	**0.902±0.072**	**0.882±0.080**	**0.892±0.086**	**0.886±0.083**	**0.037±0.007**
DenseNet121	0.875±0.054	0.865±0.064	0.868±0.035	0.859±0.052	0.069±0.016

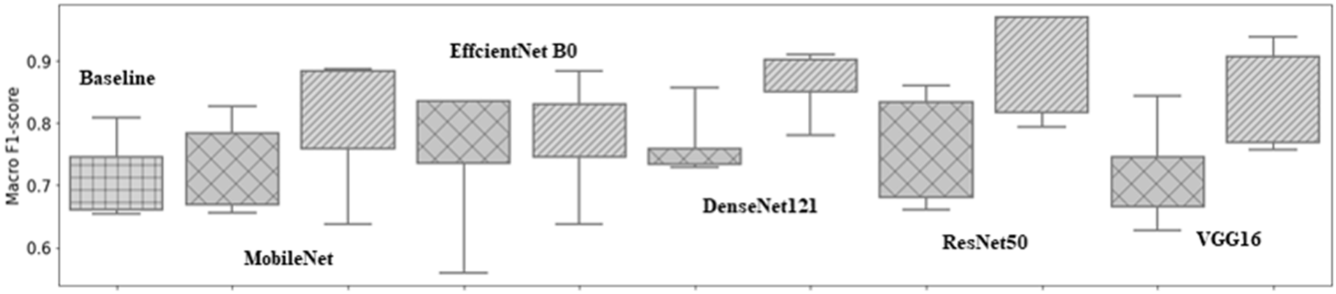

FIGURE 5. The macro F1-score of the baseline model and the selected five ImageNet pre-trained CNNs obtained from SFFCV. Square pattern plot indicates the baseline model, the crosses ('x') plots indicate models that act as feature extractors while the diagonal lines plots indicate fine-tuned models. The overall plot suggests that fine-tuned models achieve higher stability and performance compared to those which act as feature extractors.

to all consequent layers. The ℓ^{th} layer obtains the feature maps of all previous layers, $x_0, \ldots, x_{\ell-1}$, as input: $x_\ell = H_\ell([x_0, \ldots, x_{\ell-1}])$, where $[x_0, \ldots, x_{\ell-1}]$ denotes as the concatenation of the feature maps yielded in layers $0, \ldots, \ell - 1$ [35].

It was observed that the baseline model performed surprisingly well in D_{ts} (see Figure 6). Despite the simplicity, the baseline model outperformed all selected pre-trained *CNN*s as feature extractors and some fine-tuned models. Therefore, the ability of simple vanilla CNN should not be underestimated in exploring the implementation of *CNN* in different applications.

B. COMPARISONS BETWEEN FEATURE EXTRACTION AND FINE-TUNING

Despite the fact all models can segregate healthy oil palm trees from non-healthy trees with up to 0.7 macro F1-score, it was discovered that fine-tuned models performed better than those models that acted as feature extractors. Feature extraction relies on the learned knowledge of the f_θ from D_{source} to extract salient features from D_{target}. On the other hand, fine-tuning optimizes the θ in the f_θ with micro steps, adjusting the feature representations of the classifier to be more applicable to the D_{target}. As a result, fine-tuned models can achieve a better performance than non-fine-tuned models.

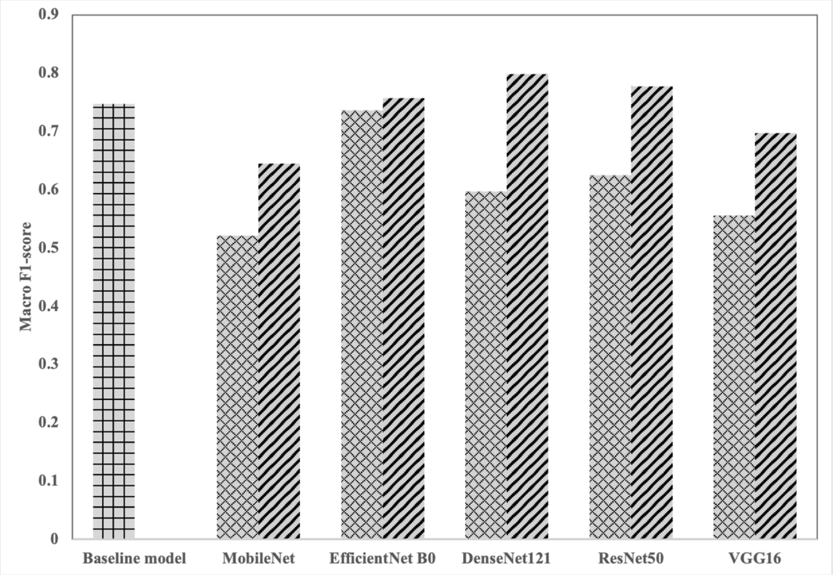

FIGURE 6. The macro F1-score of the baseline model and the selected five ImageNet pre-trained CNNs obtained from D_{ts}. The square pattern plot indicates the baseline model, the crosses ('x') plots indicate models that act as feature extractors while the diagonal lines plots indicate fine-tuned models. The overall chart suggests that fine-tuned models achieve higher scores compared to others. Surprisingly, the baseline model outperforms all CNNs that act as feature extractors and some fine-tuned models.

For each model, the best model weights were selected during model training to perform model inference on the test set. However, according to accuracy and loss curves for the 2 best performing pre-trained CNNs, namely DenseNet 121 and ResNet50 (see Figure 7), it was observed that the training loss curve of the fine-tuned models (see second column in Figure 7) decreased gradually with experience while the validation loss curve decreased to a point and stagnated or even showing a trend of increase. This trend proved that fine-tuning the models with large learning rates resulted in model overfitting. Additionally, it was also noticed that fine-tuning the pre-trained models suffered from longer training time.

There were many ways to implement the fine-tuning techniques to the model. Fine-tuning in this research was done by unfreezing the whole feature extractor (i.e. having no frozen layers) while keeping the learning rate at the minimum (0.00001). Grid search was not done in the search of the optimum number of frozen layers in each model. This ablation study can be further studied in future.

C. LIMITATIONS OF STUDY

In this study, the application of CNN in classifying oil palm trees into H and NH classes was thoroughly explored. As reviewed, no prior study adopted the TLS dataset for deep learning research purpose. Hence, the results are only applicable to the study dataset. For the experiment, five $CNNs$ (MobileNet, EfficientNet B0, VGG16, ResNet50, and DenseNet121) were selected and pre-trained with the ImageNet dataset. Early CNN architectures such as AlexNet [39] and GoogleNet [40] were omitted since these architectures were considered to be outdated. Thus, the study findings would be more meaningful and comparable. Two experimental modes were conducted: (1) pre-trained model as feature extractor and (2) fine-tuning the whole pre-trained model. However, the option of selecting specific convolutional block to be fine-tuned while maintaining the parameter θ of the remaining block was not investigated since selecting the optimal layers for fine-tuning was considered to be resource-extensive [41].

D. CHALLENGES OF STUDY

This study encountered two significant challenges, namely imbalanced data and model overfitting. The proportion of the H and NH classes in the dataset was unequal (see Table 1). Fitting imbalanced data into the CNN may cause the model to be prone to predict the higher proportioned class. To combat this issue, class weighting technique was implemented to give the minority class a more significant weight in the BCE_{loss} to impose a greater penalty on the minority class. For the second challenge, horizontal and vertical flipped images were artificially induced into D_{tr} to form $D_{tr,aug}$. Furthermore, real-time random augmentations (rotation, brightness, zoom) were applied to $D_{tr,aug}$ during model training. In addition, dropout layers in the $C(\cdot|W_b)$ that randomly nullify input units at a specific rate during model training were also implemented to combat model overfitting issue.

E. COMPARISON TO OTHER STUDIES

Compared to other BSR studies, the trend of small dataset leading to model overfitting can be observed.

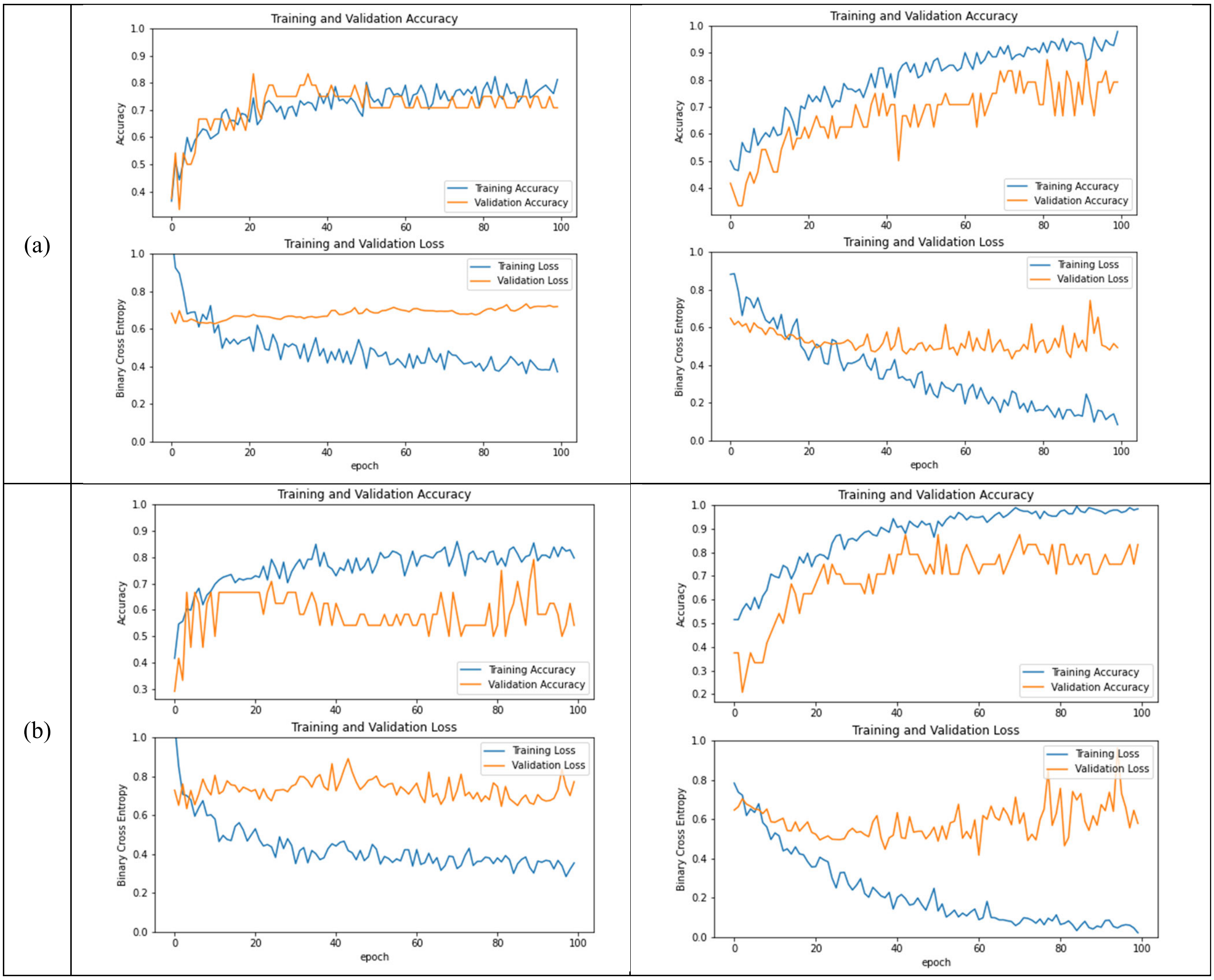

FIGURE 7. The accuracy and loss graphs of the two best performing pre-trained CNNs during the SFFCV stage, i.e. (a) DenseNet121 and (b) ResNet50, that act as feature extractors (first column) and fine-tuned (second column) in D_{ts}. Best performing models were chosen based on the Macro-F1 score during SFFCV stage. It was noted that fine-tuned models suffered from model overfitting owing to the limited samples in D_S.

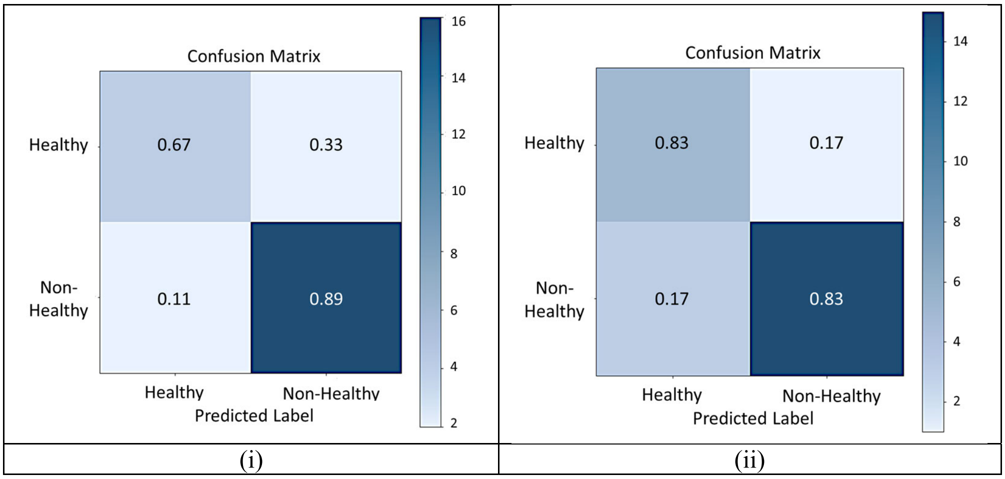

FIGURE 8. The confusion matrix of the best performing pre-trained CNNs (i) ResNet50 in SFFCV and (ii) DenseNet121 in D_{ts}.

P. Ahmadi et al. (2022) utilized an UAV together with a NIR and collected a total of 287 oil palm tree canopy image samples. The images were then classified into three disease levels. The images were then analyzed and classified using

TABLE 9. **The test performance of the baseline model and five ImageNet pre-trained CNNs in terms of accuracy, macro precision, macro recall, macro F1-score and inference time. The results suggest that the DenseNet121 outperformed other CNNs in the classification task. The inference time listed are less than 0.1 second thus the differences are negligible.**

Model	**Accuracy**	**Macro Precision**	**Macro Recall**	**Macro F1-Score**	**Inference Time (s)**
Baseline	0.833	0.800	0.722	0.747	0.006
Frozen ImageNet weights (learning rate = 0.0001)					
MobileNet	0.542	0.563	0.583	0.521	0.018
EfficientNetB0	0.792	0.727	0.750	0.736	0.057
VGG16	0.667	0.556	0.556	0.556	0.010
ResNet50	0.667	0.629	0.667	0.625	0.041
DenseNet121	0.667	0.594	0.611	0.597	0.073
Fine-tune ImageNet weights (learning rate = 0.00001)					
MobileNet	0.667	0.667	0.722	0.644	0.021
EfficientNetB0	0.792	0.744	0.806	0.758	0.065
VGG16	0.750	0.688	0.722	0.698	0.011
ResNet50	0.833	0.778	0.778	0.778	0.044
DenseNet121	**0.833**	**0.781**	**0.833**	**0.798**	**0.092**

ANN model. The training accuracy achieved was 97.52% for healthy palm trees, while the testing accuracy dropped to 72.73%. Similar can be said for early infected palm trees, with the training accuracy of 88.50% and testing accuracy of 57.14%. A model that is accurate during training while not being very accurate during testing is a sign of an overfitted model [22].

While increasing the sample size, the accuracy of training and testing can be substantially improved. L.Z Yong et al. (2023) demonstrated that by applying VGG16 model on 1610 original images at 938nm wavelength, which 1127 images were used for training and 483 images for testing, provided an accuracy of 91.93% and F1 score of 91.72%. Despite showing very promising results, this is not suitable for commercial use as this research is for nursery application only [24].

Therefore, it is recommended to analyze TLS images using CNN models but a significantly larger TLS dataset is required. As shown, CNN and deep learning is a viable method and even with limited dataset, macro F1-score of around 0.8 can be achieved. With a significantly larger dataset, it may be possible to utilize similar CNN models to accurately classify the oil palm TLS images into different infection levels (T0, T1, T2 and T3 sub-classes), thus detecting the oil palm trees that are infected but showing little symptoms for being in the early infected stage.

VI. CONCLUSION

From the study, it can be seen that CNN is a viable method to detect *G. boninense* infection from a sample of oil palm canopy TLS images. The best performing CNN models during the performance test were ResNet50 and DenseNet121, which gave a macro F1-score of 0.778 and 0.798 respectively. It was concluded that DenseNet121 was the best performing fine-tuned CNN model as compared to the baseline model and the other 4 CNN models tested during the model testing phase.

It was noted that baseline model performed relatively well compared to all feature extraction models and even some fine-tuned models. The capability of baseline CNN models should not be underestimated.

However, it was noted that fine-tuned CNN model suffered from model overfitting owing to the limitation on the dataset. To potentially improve the training and testing process, more data is needed as CNN works better when more images are provided [25], [26], [27].

The current model is only capable of discriminating TLS dataset into 2 classes, which is H and NH class. The ultimate goal of using deep-learning to classify the TLS images is to potentially classify the images into different infection level, namely T0, T1, T2 and T3 in order to detect oil palm trees that are infected early to allow early intervention and prevent the spread of *G. boninense*.

VII. FUTURE WORK

For future work, it is recommended to have a significantly larger TLS dataset in order to improve the reliability of the deep learning model. It is also recommended to test the dataset using other CNN models such as the Inceptions, NASNets, ConvNeXt, EfficientNetV2s etc. to determine if other models will work better using similar dataset.

Besides, to improve the fine-tuning technique, it is recommended to conduct ablation study to determine the optimum number of frozen layers in each model.

The ultimate goal of BSR infection classification is to classify images of oil palm trees into different infection class in order to accurately capture the trees that are infected at the early stage. The purpose of this is to allow effective sanitization and early disease control efforts. Therefore it is recommended to also attempt multiclass classification using deep learning to observe if CNN can be used on TLS dataset to accurately classify the TLS images into different infection severity classes, thus finding if the model can accurately detect early infected oil palm trees.

REFERENCES

[1] Y. Siddiqui, A. Surendran, R. R. M. Paterson, A. Ali, and K. Ahmad, “Current strategies and perspectives in detection and control of basal stem rot of oil palm,” *Saudi J. Biol. Sci.*, vol. 28, no. 5, pp. 2840–2849, May 2021.

[2] D. Bentivoglio and A. G. Finco Bucci, “Factors affecting the Indonesian palm oil market in food and fuel industry: Evidence from a time series analysis,” *Int. J. Energy Econ. Policy*, vol. 8, no. 5, pp. 49–57, 2018.

[3] I. Bharudin, A. F. F. Ab Wahab, M. A. A. Samad, N. X. Yie, M. A. Zairun, F. D. A. Bakar, and A. M. A. Murad, “Review update on the life cycle, plant–microbe interaction, genomics, detection and control strategies of the oil palm pathogen ganoderma boninense,” *Biology*, vol. 11, no. 2, p. 251, Feb. 2022.

[4] A. S. A. F. Alam, A. C. Er, and H. Begum, “Malaysian oil palm industry: Prospect and problem,” *J. Food, Agricult. Environ.*, vol. 1313, no. 22, pp. 143–148, 2015.

[5] A. P. G. Kadir, “Oil palm economic performance in Malaysia and R&D progress in 2020,” *J. Oil Palm Res.*, vol. 33, no. 2, pp. 181–214, 2021.

[6] S. A. Nusaibah, A. S. N. Akmar, A. S. Idris, M. Sariah, and Z. M. Pauzi, “Involvement of metabolites in early defense mechanism of oil palm (elaeis guineensis Jacq.) against ganoderma disease,” *Plant Physiol. Biochemistry*, vol. 109, pp. 156–165, Dec. 2016.

[7] K. Sujarit, W. Pathom-aree, M. Mori, K. Dobashi, K. Shiomi, and S. Lumyong, “Streptomyces palmae CMU-AB204T, an antifungal producing-actinomycete, as a potential biocontrol agent to protect palm oil producing trees from basal stem rot disease fungus, ganoderma boninense,” *Biol. Control*, vol. 148, Sep. 2020, Art. no. 104307.

[8] Z. Nurazah, A. S. Idris, A. M. Din, M. A. A. Manaf, A. Othman, and U. S. Ramli, “Metabolite fingerprinting of oil palm (Elaeis guineensis Jacq.) root for the identification of altered metabolic pathways associated with basal stem rot (BSR) disease,” *Physiological Mol. Plant Pathol.*, vol. 115, Aug. 2021, Art. no. 101647.

[9] R. Hushiarian, N. A. Yusof, and S. W. Dutse, “Detection and control of ganoderma boninense: Strategies and perspectives,” *SpringerPlus*, vol. 2, no. 1, p. 555, Dec. 2013.

[10] C.-L. Ho, Y.-C. Tan, K.-A. Yeoh, A.-K. Ghazali, W.-Y. Yee, and C.-C. Hoh, “De novo transcriptome analyses of host-fungal interactions in oil palm (elaeis guineensis Jacq.),” *BMC Genomics*, vol. 17, no. 1, p. 66, Dec. 2016.

[11] E. Rani, S. A. Mohshim, N. H. Yusof, M. Z. Ahmad, R. Goodacre, S. A. A. Ahmad, and L. S. Wong, “Sensitive and selective detection of DNA fragments associated with ganoderma boninense by DNA-nanoparticle conjugate hybridisation,” *J. Mater. Sci.*, vol. 55, no. 30, pp. 14965–14979, Oct. 2020.

[12] M. I. S. M. Hilmi Tan, M. F. Jamlos, A. F. Omar, F. Dzaharudin, S. Chalermwisutkul, and P. Akkaraekthalin, “Ganoderma boninense disease detection by near-infrared spectroscopy classification: A review,” *Sensors*, vol. 21, no. 9, p. 3052, 2021.

[13] C. A. T. Tee, Y. X. Teoh, P. L. Yee, B. C. Tan, and K. W. Lai, “Discovering the ganoderma boninense detection methods using machine learning: A review of manual, laboratory, and remote approaches,” *IEEE Access*, vol. 9, pp. 105776–105787, 2021.

[14] N. A. Husin, S. Khairunniza-Bejo, A. F. Abdullah, M. S. M. Kassim, D. Ahmad, and A. N. N. Azmi, “Application of ground-based LiDAR for analysing oil palm canopy properties on the occurrence of basal stem rot (BSR) disease,” *Sci. Rep.*, vol. 10, no. 1, p. 6464, Apr. 2020.

[15] C. Barnes, H. Balzter, K. Barrett, J. Eddy, S. Milner, and J. C. Suárez, “Airborne laser scanning and tree crown fragmentation metrics for the assessment of phytophthora ramorum infected larch forest stands,” *Forest Ecology Manage.*, vol. 404, pp. 294–305, Nov. 2017.

[16] N. A. Husin, S. Khairunniza–Bejo, A. F. Abdullah, M. S. M. Kassim, and D. Ahmad, “Study of the oil palm crown characteristics associated with basal stem rot (BSR) disease using stratification method of point cloud data,” *Comput. Electron. Agricult.*, vol. 178, Nov. 2020, Art. no. 105810.

[17] N. A. Husin, S. Khairunniza-Bejo, A. F. Abdullah, M. S. M. Kassim, D. Ahmad, and M. H. A. Aziz, “Classification of basal stem rot disease in oil palm plantations using terrestrial laser scanning data and machine learning,” *Agronomy*, vol. 10, no. 11, p. 1624, Oct. 2020.

[18] N. A. Husin, S. Khairunniza-Bejo, A. F. Abdullah, M. S. M. Kassim, and D. Ahmad, “Multi-temporal analysis of terrestrial laser scanning data to detect basal stem rot in oil palm trees,” *Precis. Agricult.*, vol. 23, no. 1, pp. 101–126, Feb. 2022.

[19] S. M. Hassan and A. K. Maji, “Plant disease identification using a novel convolutional neural network,” *IEEE Access*, vol. 10, pp. 5390–5401, 2022.

[20] P. Ahmadi, F. M. Muharam, K. Ahmad, S. Mansor, and I. A. Seman, “Early detection of ganoderma basal stem rot of oil palms using artificial neural network spectral analysis,” *Plant Disease*, vol. 101, no. 6, pp. 1009–1016, Jun. 2017.

[21] A. Y. Khaled, S. A. Aziz, S. K. Bejo, N. M. Nawi, and I. A. Seman, “Spectral features selection and classification of oil palm leaves infected by basal stem rot (BSR) disease using dielectric spectroscopy,” *Comput. Electron. Agricult.*, vol. 144, pp. 297–309, Jan. 2018.

[22] P. Ahmadi, S. Mansor, B. Farjad, and E. Ghaderpour, “Unmanned aerial vehicle (UAV)-based remote sensing for early-stage detection of ganoderma,” *Remote Sens.*, vol. 14, no. 5, p. 1239, Mar. 2022.

[23] C. C. Lee, V. C. Koo, T. S. Lim, Y. P. Lee, and H. Abidin, “A multi-layer perceptron-based approach for early detection of BSR disease in oil palm trees using hyperspectral images,” *Heliyon*, vol. 8, no. 4, Apr. 2022, Art. no. e09252.

[24] L. Z. Yong, S. Khairunniza-Bejo, M. Jahari, and F. M. Muharam, “Automatic disease detection of basal stem rot using deep learning and hyperspectral imaging,” *Agriculture*, vol. 13, no. 1, p. 69, Dec. 2022.

[25] J.-R. Xiao, P.-C. Chung, H.-Y. Wu, Q.-H. Phan, J.-L.-A. Yeh, and M. T.-K. Hou, “Detection of strawberry diseases using a convolutional neural network,” *Plants*, vol. 10, no. 1, p. 31, Dec. 2020.

[26] P. Sharma, Y. P. S. Berwal, and W. Ghai, “Performance analysis of deep learning CNN models for disease detection in plants using image segmentation,” *Inf. Process. Agricult.*, vol. 7, no. 4, pp. 566–574, Dec. 2020.

[27] C. Nguyen, V. Sagan, M. Maimaitiyiming, M. Maimaitijiang, S. Bhadra, and M. T. Kwasniewski, “Early detection of plant viral disease using hyperspectral imaging and deep learning,” *Sensors*, vol. 21, no. 3, p. 742, Jan. 2021.

[28] G. Hu, X. Peng, Y. Yang, T. M. Hospedales, and J. Verbeek, “Frankenstein: Learning deep face representations using small data,” *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 293–303, Jan. 2018.

[29] J. Vanschoren, “Meta-learning: A survey,” 2018, *arXiv:1810.03548*.

[30] K. Singh, “How to improve class imbalance using class weights in machine learning,” *Data Sci. Blogathon*, Dec. 2022. [Online]. Available: https://www.analyticsvidhya.com/blog/2020/10/improve-class-imbalance-class-weights/

[31] S. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, pp. 1345–1359, Nov. 2010.

[32] J. Xu and X. Dong, "A survey of transfer learning in breast cancer image classification," in *Proc. IEEE 3rd Int. Conf. Safe Prod. Informatization (IICSPI)*, Nov. 2020, pp. 220–223.

[33] 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*.

[34] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 1–11.

[35] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2261–2269.

[36] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1–9.

[37] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[39] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2022, pp. 1–9. [Online]. Available: http://code.google.com/p/cuda-convnet/s

[40] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[41] C. Munien and S. Viriri, "Classification of hematoxylin and eosin-stained breast cancer histology microscopy images using transfer learning with efficientnets," *Comput. Intell. Neurosci.*, vol. 2021, Apr. 2021, Art. no. 5580914.

YU HONG HAW received the B.Eng. degree in biomedical from Universiti Malaya (UM). With more than six years of working experience in the medical device industry, he has vast experience in the technical aspect of medical device and clinical research processes. His current research interests include image processing and machine learning. He is a registered Professional Engineer with the Board of Engineers Malaysia.

YAN CHAI HUM (Senior Member, IEEE) received the B.Eng. (Hons.) and Ph.D. degrees in biomedical engineering from Universiti Teknologi Malaysia, Skudai, Malaysia. He is currently an Associate Professor with the Department of Mechatronics and Biomedical Engineering, Lee Kong Chian Faculty of Engineering and Science, Universiti Tunku Abdul Rahman. His current research interests include image processing, artificial intelligence, machine learning, signal processing, and medical image analysis. He registered as a Chartered Engineer (U.K.) and Professional Engineer (MY), in 2016.

JOON HUANG CHUAH (Senior Member, IEEE) received the B.Eng. degree (Hons.) from Universiti Teknologi Malaysia, the M.Eng. degree from the National University of Singapore, and the M.Phil. and Ph.D. degrees from the University of Cambridge. He is currently the Head of the VIP Research Group and an Associate Professor at the Department of Electrical Engineering, Faculty of Engineering, Universiti Malaya. He is a Chartered Engineer registered under the Engineering Council, U.K., and also a Professional Engineer registered under the Board of Engineers, Malaysia. He was the Honorary Treasurer of IEEE Computational Intelligence Society (CIS) Malaysia Chapter and the Honorary Secretary of IEEE Council on RFID Malaysia Chapter. He is the Chairman of the Institution of Engineering and Technology (IET) Malaysia Network. He is a Fellow and he was the Honorary Secretary of the Institution of Engineers, Malaysia (IEM). His main research interests include image processing, computational intelligence, IC design, and scanning electron microscopy.

WINGATES VOON (Member, IEEE) received the bachelor's degree (Hons.) in biomedical engineering from Universiti Tunku Abdul Rahman, where he is currently pursuing the Ph.D. degree in engineering. He is an Aspiring Engineer who is a member of the Board of Engineers Malaysia (BEM). His research interests include deep learning, meta learning, and medical image analysis.

SITI KHAIRUNNIZA-BEJO received the B.E. degree in computer systems and communication engineering from UPM and the Ph.D. degree in image processing from the University of Surrey, U.K. She is currently an Associate Professor with the Department of Biological and Agricultural Engineering, Faculty of Engineering, Universiti Putra Malaysia (UPM), and the Head of Smart Farming Research Centre (SFTRC), Faculty of Engineering, UPM. She is also a Research Associate with the Institute of Plantation Studies, UPM. She has more than 150 scientific publications, one patent, and four copyrights. Her research interests include imaging technology, precision agriculture, remote sensing, and artificial intelligence. She is a Professional Member, a Malaysian Society of Agricultural Engineers (MSAE), and a member of the Malaysian Remote Sensing Society (MRSS), The Institution of Engineering and Technology (IET), and the International Society of Precision Agriculture (ISPA). She is also the Editorial Board Member of *Information Processing in Agriculture*, *Journal of Agricultural Engineering*, and *Advances in Agriculture, Horticulture and Entomology*.

NUR AZUAN HUSIN received the B.E. degree in biological and agricultural engineering from Universiti Putra Malaysia (UPM), the M.S. degree in biological and agricultural engineering from Texas A&M University, USA, and the Ph.D. degree in agricultural mechanization and automation from the Department of Biological and Agricultural Engineering, Faculty of Engineering, UPM. He is currently a Senior Lecturer with the Department of Biological and Agricultural Engineering, Faculty of Engineering, UPM, and a Young Research Associate with the Smart Farming Research Centre (SFTRC), Faculty of Engineering, UPM. His research interests include remote sensing and precision agriculture. He is a registered Professional Technologist of Malaysia Board of Technologists (MBOT), currently an Executive Member of the Malaysian Society of Agricultural Engineers (MSAE) and a Graduate Member of the Institute of Engineers Malaysia (IEM) and the Board of Engineers Malaysia.

POR LIP YEE (Senior Member, IEEE) received the Ph.D. degree from Universiti Malaya, Malaysia, in 2012. He is currently an Associate Professor with the Department of System and Computer Technology, Faculty of Computer Science and Information Technology, Universiti Malaya. His research interests include neural networks (such as supervised and unsupervised learning methods, such as support vector machine and extreme learning machine), bioinformatics (such as biosensors and pain research), computer security [such as information security, steganography, and authentication (graphical password)], grid computing, and e-learning framework.

KHIN WEE LAI (Senior Member, IEEE) received the B.Eng. degree (Hons.) from Universiti Teknologi Malaysia (UTM), Malaysia, and the Ph.D. degree from Technische Universität Ilmenau, Germany, and UTM, Malaysia, under the DAAD Ph.D. Sandwich Programme. He is currently an Associate Professor with the Faculty of Engineering, Universiti Malaya. His research interests include machine learning, medical image processing, and healthcare analytics. He is a registered Professional Engineer with Practicing Certificate (Ir) of Board of Engineers Malaysia, a fellow of the Engineers Australia (FIEAust), the Institute of Engineers Malaysia, CEng (U.K.), APEC Engineer IntPE (Australia), and a Chartered Professional Engineer (CPEng.) at NER, Australia. He is currently an Associate Editor of *IET Image Processing*.