Malays. J. Bioeng. Technol. Vol. 1, No. 1, (2024): 48-65
https://doi.org/10.70464/mjbet.v1i1.1275

MALAYSIAN JOURNAL OF BIOENGINEERING AND TECHNOLOGY

Advancing Agricultural Sustainability: Smart Monitoring of Oil Palm Seedlings Through Innovative Image Processing Techniques

Nur Ain Najwa Mohd Adib, Shaparas Daliman*

Faculty of Earth Science, Universiti Malaysia Kelantan, Jeli Campus, 17600 Jeli, Kelantan, Malaysia.

**Corresponding author: shaparas@umk.edu.my*

ARTICLE INFO

Received:21 March 2024
Accepted:19 April 2024
Online: 22 June 2024
eISSN: 3036-017X

ABSTRACT

Good oil palm seedlings planted in Malaysia can enhance the nursery area, the economy, and rural employment. Nitrogen (N), potassium (K), \and magnesium (Mg) deficiencies in oil palm seedlings could have an adverse effect on growth and seedling quality. In contrast, excess fertiliser in oil palm seedlings could reduce macronutrients and soil organic matter levels. Moreover, various diseases appear on the oil palm seedling leaves resulting from nutrient deficiencies. Oil palm seedlings with nutrient deficiencies were more susceptible to pathogens and diseases than healthy oil palm seedlings. Thus, image processing made it possible to quickly and accurately control oil palm seedlings' growth and avoid diseases. In this research study, image processing was proposed to classify nutrient deficiencies in oil palm seedling leaves subjected to the three different fertiliser rates. The dataset for this research study was collected from a nursery and consisted of 868 images classified into four classes (Healthy, Nitrogen, Potassium and Magnesium). According to the experiment's findings, the Xception model achieved the highest percentage of classification accuracy, 98.60%, within a short time. It can be concluded that the proposed implementation of image processing for the classification of nutrient deficiencies in oil palm seedling leaves was effective. However, more datasets could be added in the future to achieve a better balance and enhance classification performance.

Keywords: Oil Palm Seedlings; Nutrient Deficiency; Image Processing, Agricultural Sustainability; Fertiliser Management; Classification Accuracy

1. Introduction

Cultivating oil palm (*Elaies guineensis*) is widespread in West Africa, but effectively controlling and fertilising large nursery areas poses considerable difficulties. The inefficiency and high expense of traditional manual approaches emphasise the necessity for an affordable and easy methodology to analyse the growth of oil palm seedlings. Image processing has become an acceptable method, allowing for effective monitoring and fertilisation.

Previous research findings have investigated the application of machine learning and deep learning techniques for the detection of plant diseases [1]. However, conventional approaches necessitated human involvement in the collection of data. Recent research conducted by [2] has utilised advanced deep learning models such as AlexNet and AlexNet-SVM to automate the process of representing data. This has led to great progress in identifying deficiencies in nutrients in oil palm leaves. Image processing techniques have been shown in multiple studies to be useful in enhancing agricultural activities by offering non-destructive and precise data about plants.

The studies conducted by [3] and [4] have shown that Convolutional Neural Networks (CNNs) can identify nutritional deficiencies. However, there remained difficulties in precisely recognising particular nutrients. Additional investigation is required to enhance these techniques and investigate their suitability in other agricultural environments and circumstances.

This research aims to fill these knowledge gaps by creating an image-processing method that can accurately categorise nutrient deficiencies in oil palm seedlings across three different NPK treatments. The results provided valuable insights for optimising fertilisation techniques, improving seedling vitality, minimising expenses, and maximising agricultural productivity. The greater implications include the sustainability of the environment and the development of agricultural policies, highlighting the importance of using modern technologies in the field of agriculture. Farm experts have the capability to utilise mobile applications that are equipped with machine learning and deep learning algorithms in order to capture images of plant leaves. The application utilises image analysis to identify and diagnose nutrient deficiencies by examining patterns and characteristics recognised by the trained model. These programs offer prompt feedback, recommending modification solutions such as accurate fertiliser treatments.

2. Materials and Methods

2.1 Materials

The materials used in this research included fifteen 3-month-old oil palm seedlings and NPK (16:16:16) fertiliser, specifically YaraMila fertiliser (16:16:16). The dataset was collected and captured using a smartphone. The models were developed using Python programming 3.8.3 and executed on a Jupyter Notebook with plenty of space, sufficient RAM, and a capable Intel(R) Core (TM) i5-4210U CPU @ 1.70GHz 2.40 GHz. Data was imported into the Jupyter Notebook software. The 3-month oil palm seedlings were purchased from Semaian Gua Musang Sdn Bhd, which is located in Gua Musang, approximately 143.0 km from Kota Bharu, Kelantan. The coordinates of the source are 4°49’05.2 "N 101°57'24.4"E. They were taken to Kota Bharu for further analysis. Seedlings of oil palm were planted into UV polybags that measured 12’ × 15’ and were filled with a soil combination consisting of coco peat, topsoil, and rice husk ash. Agricultural wastes such as cocopeat and rice husk ash were chosen for their affordability and accessibility. A recent study conducted by [5] found that cocopeat has a remarkable ability to retain water, effectively preserving the humidity levels within. Furthermore, the structure of rice husk ash was characterised by high porosity, which enhanced both drainage and aeration. When the soil is porous, the roots of oil palm seedlings can easily penetrate it, allowing for proper growth. The spacing between planting is 0.40 m × 0.40 m triangular to minimise the impact of watering on nearby oil palm seedlings. Each oil palm seedling in a polybag is given a consistent amount of water, ranging from 0.25 to 0.50 litres. Watering is done twice a day, once in the morning and once in the evening. The seedlings were given time to establish themselves before receiving fertiliser treatments. The oil palm seedlings are divided into three different treatments: those with adequate fertiliser, those with a limited amount of fertiliser, and those without any fertiliser, according to Table 1.

Table 1: Description of treatments to examine the effect of different rates of NPK fertiliser on growth of oil palm seedlings

Treatment	T1	T2	T3
Rate of Fertiliser, (g)	2g	1g	No fertiliser application

Source: [6]

Small amounts of YaraMila fertiliser were used as NPK fertiliser, which was 2 g and 1 g because oil palm seedlings possess different demands on nutrients when compared with mature-grown oil palm plants [7]. Excessive use of fertiliser can result in nutrient toxicity, whereas insufficient amounts may not adequately promote optimal development. Employing a small quantity ensures that the oil palm seedlings have sufficient nutrients while avoiding the potential of over-fertilization. Utilising small quantities of fertiliser is more economical, particularly during the early stages of the growth of plants. This is especially crucial in research environments where limitations on funding and the effective use of resources are vital factors to consider.

eISSN Number: 3036-017X

2.2 Methodology

2.2.1 Integrated Image Processing Workflow with Machine Learning and Deep Learning

Fig. 1 shows an image processing method that started with taking an image using a mobile device and then transferring it to a computer for data analysis in a Jupyter Notebook. The image underwent pre-processing steps, including resizing and splitting, within the computer. The processed data went through analysis using both machine learning and deep learning methods. The deep learning process utilised a Convolutional Neural Network (CNN) and various pre-trained models for transfer learning and fine-tuning, such as VGG16, VGG19, InceptionV3, and others, for feature extraction and classification. The network's final layers consisted of batch normalisation, dense layers, and activations that ultimately led to the final output. The CNN's classification process was explained using layers like ReLu, Softmax, and Flatten, resulting in an output that classified the image into categories like Healthy, Nitrogen, Potassium, and Magnesium. Additionally, the conventional machine learning process employed a multiclass Support Vector Machine (SVM) with a one-against-all approach to generate the final output.

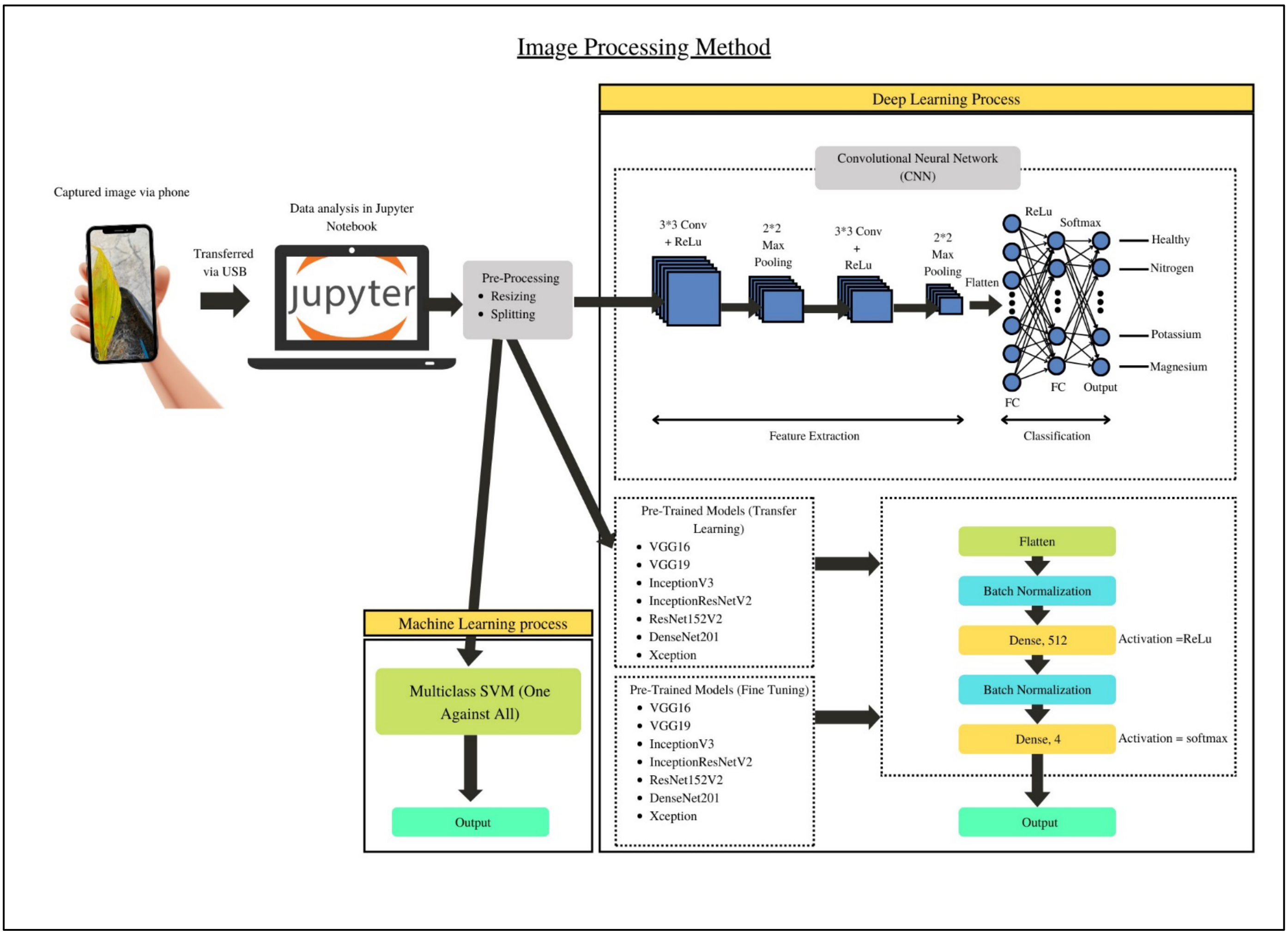


Fig. 1: Integrated Image Processing Workflow with Machine Learning and Deep Learning

2.2.2 Model Machine Learning Frameworks: Multiclass SVM (One Against All) Approaches

According to Fig. 2, in machine learning workflow, after the dataset is loaded, it goes through a critical enhancement process known as data augmentation. This method, as described by [8], effectively increases the size of the training dataset by creating modified versions of existing images. This helps improve the overall performance of the model and reduces the risk of overfitting. Image data augmentation is implemented, specifically by applying a width shift. Nine augmented images are generated and displayed from a single original image. The images are saved to a designated output directory. When it comes to organised move images, the dataset is systematically split into separate batches or sets. This section transfers the initial 100 images from the nitrogen dataset batch to a different directory. Instead of dealing with a large batch, which can be quite demanding in terms of computation or time, a smaller subset is utilised for initial analysis or experimentation [9].

Matplotlib plots are displayed, and the CIFAR10 dataset has been loaded. The shapes of the data are also displayed. CIFAR-10 is widely utilised as a standard in the machine learning community for assessing and contrasting the effectiveness of various algorithms, particularly in image classification tasks [10]. Splitting data into training, validation, and test sets enables effective model training, fine-tuning of hyperparameters, and accurate evaluation of the final model, avoiding the risk of overfitting. Effective data preprocessing, including reshaping and normalisation, plays a crucial role in numerous algorithms. The data was resized to a 32x32 pixel dimension for processing. The Train-Validation-Test split ratio was determined with careful consideration. The training dataset consisted of 1500 samples, while both the validation and test datasets were allocated 300 samples each. The training ratio was approximately 71.43%, obtained by dividing the total number of training datasets by the sum of training, validation, and test datasets. Specifically, this was calculated as 1500 divided by 2100. The validation and testing ratios were both calculated to be 14.29%. This was determined by dividing the total number of validation and test samples by the aggregate of 2100. The dataset was split into three parts, with a ratio of approximately 71.43% for the training set and 14.29% for both the validation and test sets. Applying preprocessing techniques, like subtracting the mean to normalise data, can enhance the convergence speed and performance of machine learning algorithms. This ensures data consistency and scale, as highlighted by [11]. Reshape the image data into rows to properly prepare it for model training. Modify datasets to ensure they meet the necessary requirements for model input.

A naive implementation of the SVM loss is evaluated on a random set of weights [12]. In the realm of SVMs (Support Vector Machines), understanding the loss from random weights acts as a reference point, providing clarity on how the training process should evolve. Gradient computation, fundamental to machine learning models, undergoes a verification process termed gradient checking [13]. This step ensures that models trained using gradient-based optimisation techniques are learning correctly. Stochastic Gradient Descent (SGD), a pivotal optimisation algorithm, is employed for training. Stochastic Gradient Descent (SGD) is a widely used optimisation algorithm for training machine learning models, especially with large datasets [14]. Stochastic Gradient Descent (SGD) is applied to train a linear SVM classifier. Hyperparameter tuning comes into play, helping find the optimal settings for learning [14]. Experiment with different hyperparameters to find the optimal configuration for the SVM model. Using the validation set, learning rates and regularisation strengths are tuned to find the best combination. Furthermore, distinct learning rates were set at 2e-7, 3e-7, and 5, while regularisation strengths were chosen to be 5e3, 5e4, and 5. Visualisation tools, like heatmaps, assist in making sense of these hyperparameters, offering a comprehensive view of model performance across various configurations. Finally, the model is put to the test. The test set evaluation acts as an unbiased yardstick, simulating how the model would perform in real-world scenarios [14]. Evaluate the trained SVM model on a separate test set to assess generalisation performance. If test accuracy is achieved above 90%, it proceeds to the learned weights, and if it is below 90%, it undergoes checking hyperparameter tuning. A confusion matrix for the predictions on the test set is displayed. Compute various performance metrics to evaluate the model. A detailed classification report is generated, which includes precision, recall, and F1 scores for each class.

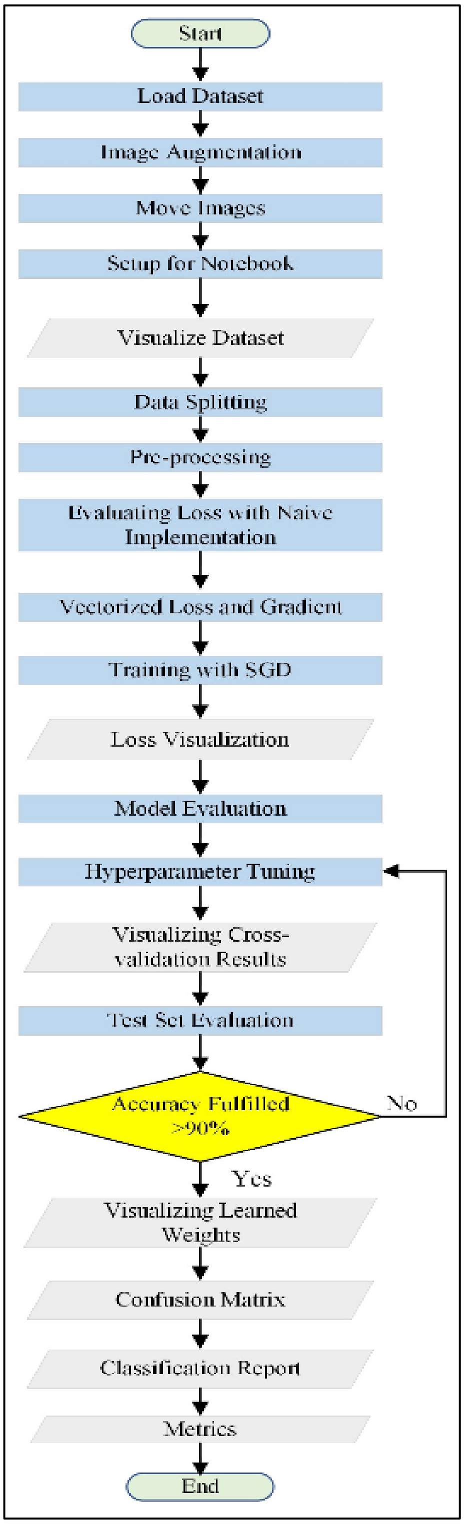


Fig. 2: The Proposed Model Multiclass SVM (OAA)

2.2.3 Multi-Model Deep Learning Frameworks: From CNN to Transfer Learning and Fine-Tuning Approaches

According to Fig. 3, during the model-building phase, the CNN architecture is established by utilising convolutional layers to extract features from images, pooling layers for downsampling, and dense layers for classification with TensorFlow's Keras API. The model is compiled using an optimiser known for its efficiency [15]. This architecture employs the hierarchical pattern in data and constructs more complex patterns by utilising smaller and simpler patterns. The training phase closely monitors the performance metrics of both training and validation to ensure that overfitting is avoided. The model is trained using a designated batch size and number of epochs. Validation, a subset of the training data, helps in monitoring overfitting. It is an important aspect of the training process [16]. Keeping track of the accuracy and loss throughout the training process offers valuable insights into the model's progress.

Finally, the model evaluation phase provides insights into the model's generalisation capabilities on unseen data. Post-training, the model's real-world utility is tested on unseen data. The model was assessed using the test data to determine its accuracy and loss. Predictions were made on the test data. Images from the test dataset were displayed alongside their predicted labels for visual assessment of the model's performance. A confusion matrix was created to analyse the model's performance across various classes. The model's performance was quantified by calculating and displaying accuracy, precision, recall, and F1 scores. A classification report was generated for a thorough evaluation. An important consideration is the specified accuracy criterion for the CNN, Transfer Learning model, and Fine-Tuning model. An accuracy threshold of 90% has been established for this study. The purpose of implementing this strict criterion is to ensure that the image classification model is not only functional but also highly reliable.

The transfer Learning model includes foundational packages similar to those used in CNN, with a particular focus on TensorFlow for deep learning. Just like the CNN model, datasets are loaded, and their distributions are visualised. This step guarantees the quality and preparedness of the data. A Sequential model, which is a linear stack of layers, was established. A pre-trained model was incorporated as the foundation for feature extraction, with specific parameters like input shape, weights, and the top layers configured as non-trainable. A flattened layer was included to condense the features into a single dimension. A Batch Normalization layer was added to normalise the activations of the neurons. Classification layers with ReLU and Softmax activation functions were added. Ultimately, the model was compiled using the Adam optimiser and the sparse categorical cross-entropy loss function. Using this architecture, the model can fine-tune the weights based on the specific dataset, enhancing its performance. Model evaluation provides valuable insights into the performance of the model on unseen data, taking into account the combined knowledge from pre-trained weights and current data.

The Fine-Tuning model methodology is a further development of the Transfer Learning approach. The initial steps are the same: a pre-trained model was incorporated as the foundation for feature extraction, with a designated input shape, and the upper layers were excluded. Additional layers were included to enhance the processing and classification capabilities. According to the specified requirements, certain layers were designated as trainable. The model was compiled, with careful consideration given to the loss function, optimiser, and metrics used for evaluation. According to [17], the process of fine-tuning involves not only the customisation of the top layers of the pre-trained model for the particular task at hand but also the rendering of certain deeper levels of the model that are trainable. A notable difference in this case, in contrast to Transfer Learning, is that specific layers of the pre-trained Fine-Tuning model can be adjusted to meet specific requirements, enabling greater customisation. This allows for a more precise fine-tuning of the pre-trained model to match the unique attributes of the current dataset. This approach is commonly used when dealing with large datasets to avoid overfitting. Allowing deeper layers to be trainable can result in overfitting when working with smaller datasets [17]. After fine-tuning, the training and evaluation phases continue to be consistent, offering valuable insights into the model's performance.

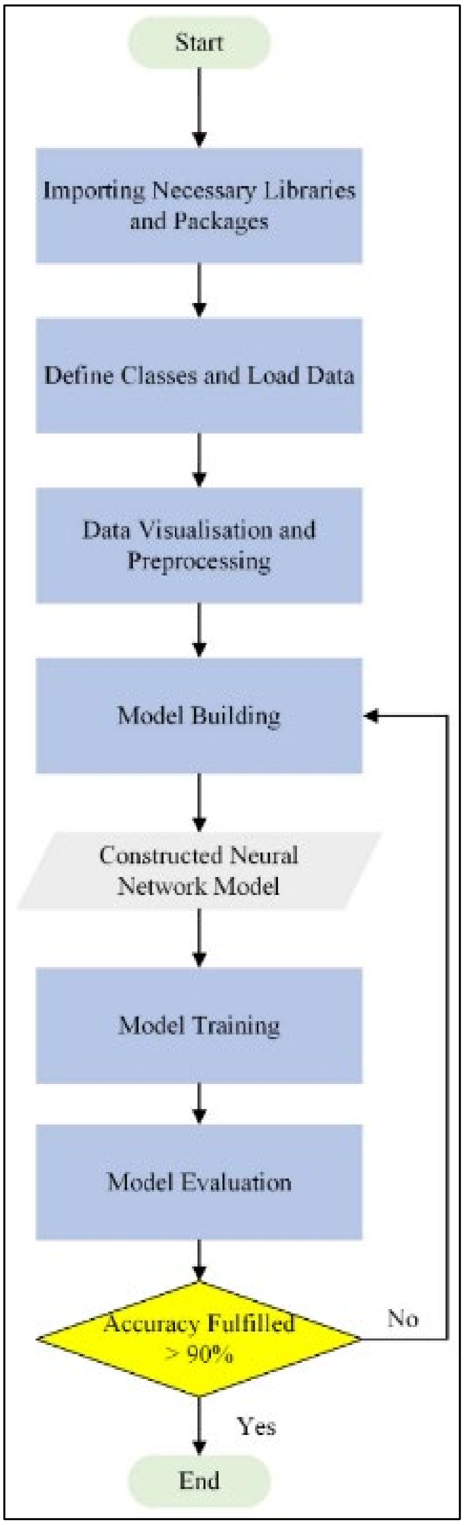


Fig. 3: The Proposed Multi-Model Deep Learning (CNN, Transfer Learning and Fine-Tuning)

2.2.4 Performance Result from Confusion Matrix

2.2.4.1 Accuracy

The accuracy of a classifier is determined by calculating the ratio of correct predictions to the total number of samples. This is done by considering both True Positives (TP) and True Negatives (TN) and summing up all entries such as True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN) in the confusion matrix. The assessment of the deep learning classifier's performance heavily depends on the accuracy metric, which is widely regarded as crucial. A way to summarise the outcome in the confusion matrix is by calculating accuracy, as shown in Equation 1.

$$\text{Accuracy (\%)} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \times 100\% \tag{1}$$

2.2.4.2 Precision

Precision, also referred to as positive predictive value (PPV), is a metric that measures the number of samples that are correctly classified as positive out of all the samples that are predicted as positive. Precision is employed as a performance metric in situations where the objective is to minimise the number of false positives. Ensuring a low rate of false positives, or high precision, is a crucial aspect of the model; in other words, precision refers to the possibility of correctly identifying a positive outcome among all the instances that were predicted as positive according to Equation 2.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

2.2.4.3 Recall

In contrast, recall measures the number of positive samples that are correctly identified by positive predictions. The recall metric is employed in situations where the objective is to correctly identify all positive samples, thereby minimising the occurrence of false negatives. To enhance the recall metric, it is essential to reduce the occurrence of False Negatives. Recall also referred to as sensitivity, hit rate, or True Positive Rate (TPR), can be denoted by various alternative terms. The calculation of recall can be derived from Equation 3:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

2.2.4.4 F1 Score

The comprehensive evaluation of precision and recall is crucial, as focusing only on either measure fails to offer a complete understanding of the situation. One method for summarising them is through the use of the f-score or f-measure, which calculates the harmonic mean of precision and recall. The F1 Score can be mathematically represented as Equation 4:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision+Recall}} \tag{4}$$

eISSN Number: 3036-017X

3. Results and Discussions

3.1 Image Datasets

The image dataset recorded various categories of nutrient deficiencies throughout a six-month time frame. The dataset contained a total of 868 images. Out of these, there were 586 images of healthy oil palm seedlings, 175 images of nitrogen deficiencies, 42 images of potassium deficiencies, and 65 images of magnesium deficiencies.

3.2 Classification of Nutrient Deficiency in Oil Palm Seedling Leaves Using Machine Learning Techniques

The presented Fig. 4 showcases two plots illustrating the performance of a Multiclass SVM in the task of classifying nutrient deficiencies in oil palm seedling leaves. The top graph pertains to the model's training accuracy, while the bottom one relates to its validation accuracy. Nutrient Deficiencies training accuracy suggests that the top graph visualises the accuracy of a model trained to predict nutrient deficiencies based on various combinations of hyperparameters during its training phase. Nutrient Deficiencies validation accuracy suggests that the bottom graph shows how well the model performs on new, unseen data (validation data) with the same hyperparameters. The performance of the SVM model can differ based on various combinations of two hyperparameters, such as the learning rate and regularisation strength. The graph displays logarithmic values on both the x-axis and y-axis. The top graph shows red dots with a high level of accuracy, approaching 0.99. Similarly, the bottom graph displays red dots with an accuracy of 0.82, as indicated by the colour bar on the right side of each graph. Blue dots are indicative of reduced accuracy. Orange dots fall in between the blue and red dots, indicating a moderate level of accuracy.

The model's accuracy seems to fluctuate depending on the combination of the learning rate and regularisation strength. In the training accuracy graph (top), as it moves from left to right, the model's accuracy shows a slight increase with more red dots towards the right. It appears that using higher learning rates could be advantageous for this specific task. Nevertheless, the validation accuracy (bottom) does not exhibit a distinct pattern with the learning rate. It appears that using higher learning rates can be advantageous during training, but it may not necessarily result in improved generalisation of unseen data. The presence of red dots on the y-axis indicates that a variety of regularisation values can yield strong training accuracies. Nevertheless, when it comes to validation, the red dots representing higher accuracies appear to be closely grouped together, suggesting that superior generalisation is achieved only with certain regularisation strengths. It appears that some combinations of these hyperparameters do not produce optimal results, as indicated by the presence of blue and orange dots mixed in with the red dots. To summarise, certain combinations of learning rates and regularisation strengths can result in high training accuracies.

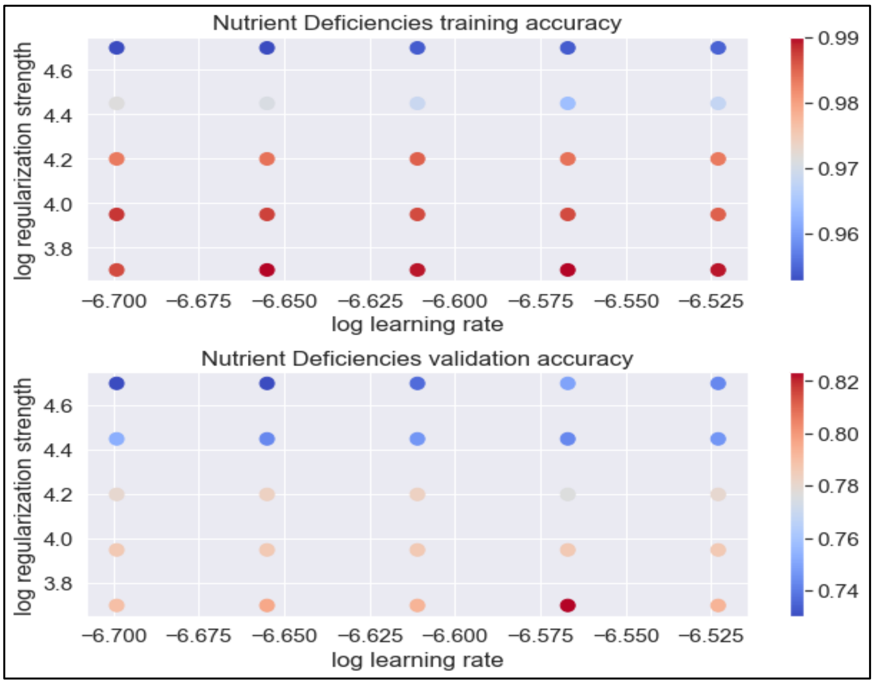


Fig. 4: Multiclass SVM Graph

eISSN Number: 3036-017X

3.3 Classification of Nutrient Deficiency in Oil Palm Seedling Leaves Using Deep Learning Techniques

The nutrient deficiency classification in oil palm seedlings has been conducted using advanced techniques such as transfer learning and fine-tuning of pre-trained CNN models. The accuracy and loss values in the training and validation process were depicted for each of the CNN, transfer learning, and fine-tuning models. These graphs-based Fig. 5 depicts the accuracy and loss metrics for both training and validation.

The CNN model rapidly converges to a minimal loss and high accuracy, indicating effective learning and the appropriateness of the model architecture and training strategy. Despite achieving 100% accuracy on training data, the model does not seem overfitting since the validation accuracy is also high, and the validation loss is consistently decreasing. The model seems to generalise well to unseen data, as evidenced by high validation and test accuracies [18]. Since the model reaches 100% training accuracy and very low training loss quite early, around epoch 14, one could potentially employ early stopping to halt the training process and save computational resources. This CNN model has demonstrated remarkable learning capability, generalisation, and reliability, showing low loss and high accuracy on training, validation, and test datasets.

While both the Transfer Learning models and Fine-Tuning models perform well, there are distinct differences in their learning behaviours. Transfer Learning is more time-efficient, showing faster convergence and requiring less computational resources, which is evident from the shorter epoch times. Fine Tuning is computationally more expensive and takes more time per epoch, but it might be a better choice when striving for the best possible performance and when computational resources are not a limiting factor [19]. The high validation and test accuracies in both methods suggest that both models generalise well to unseen data. However, overfitting should be monitored, especially considering that the training accuracy reached 1 in both methods. Fine Tuning might be more adaptable to the specifics of the given dataset due to the adjustment of more parameters in the network, as suggested by the marginally better test accuracy. Transfer Learning is more rigid but provides quick and efficient solutions with relatively less risk of overfitting, given the small dataset scenario [20]. In practical scenarios, the choice between Transfer Learning and Fine Tuning often depends on the availability of computational resources, the size and specificity of the dataset, and the required model performance [21]. Transfer learning is often preferable when there is a lack of computational power or when dealing with small datasets due to its efficiency and reduced risk of overfitting. In contrast, fine-tuning can be more suitable when the dataset is large and varied and the aim is to achieve the highest performance possible.

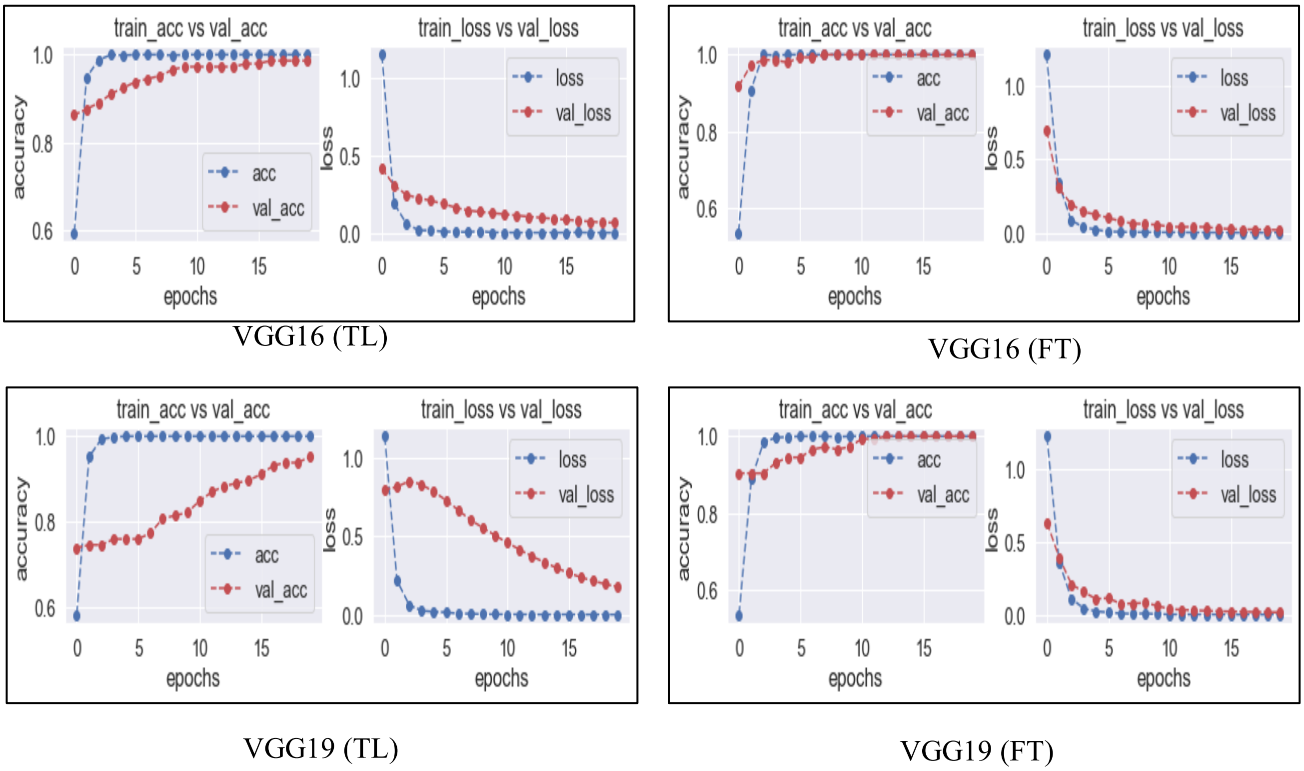
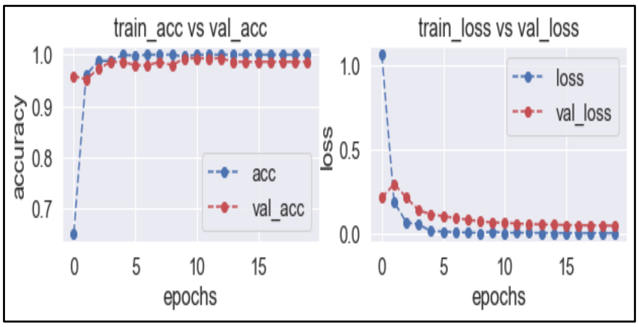
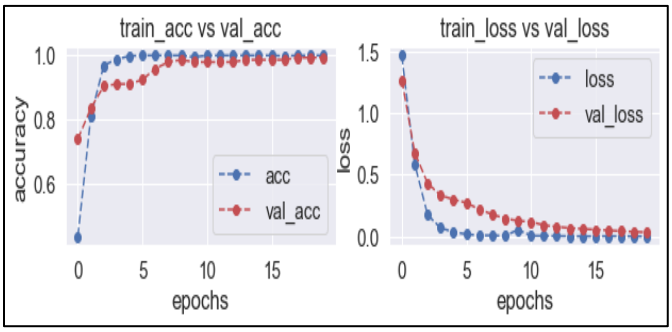


Fig. 5: Classification Using Deep Learning Models [22; 23; 24; 25; 26; 27]

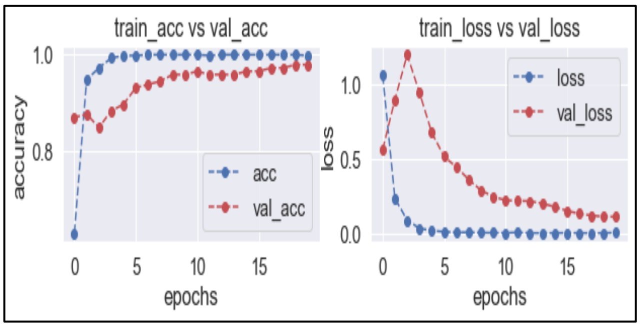
...continue Fig. 5



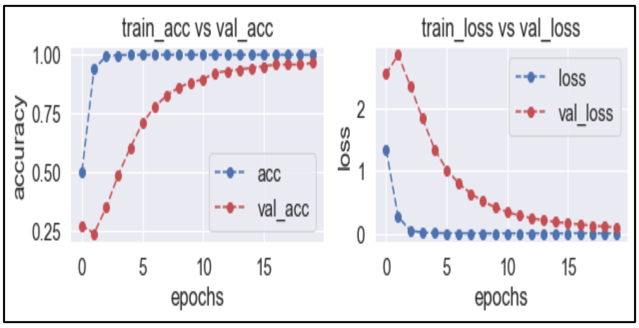
InceptionV3 (TL)



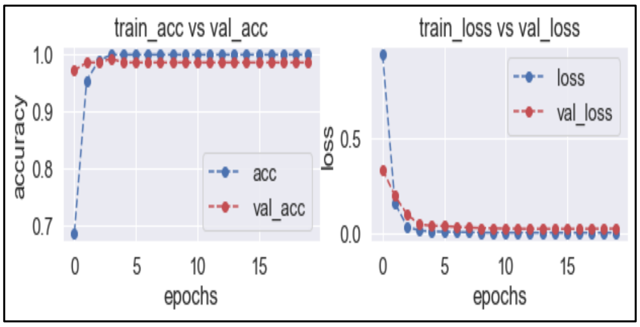
InceptionV3 (FT)



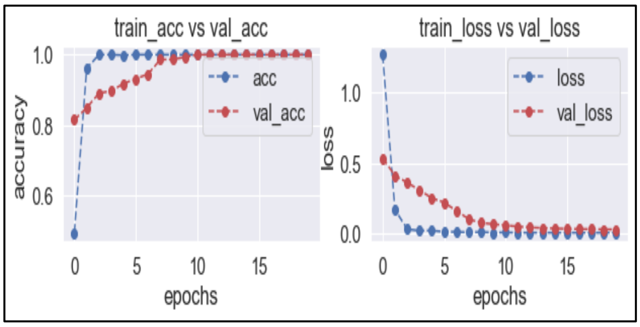
InceptionResNetV2 (TL)



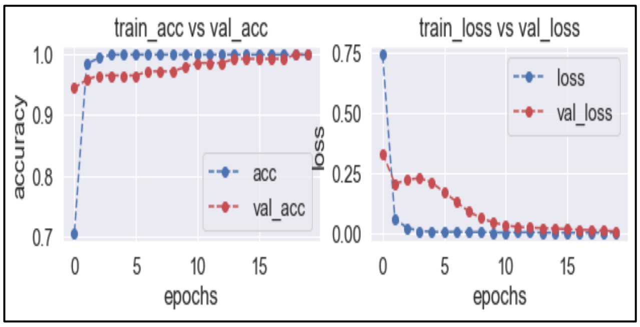
InceptionResNetV2 (FT)



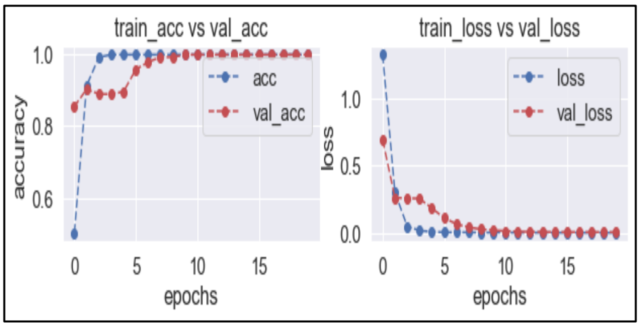
ResNet152V2 (TL)



ResNet152V2 (FT)



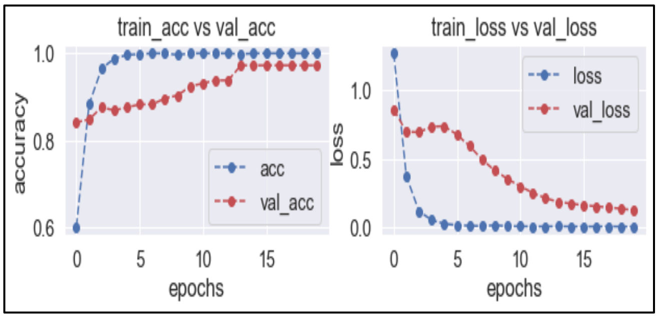
DenseNet201 (TL)



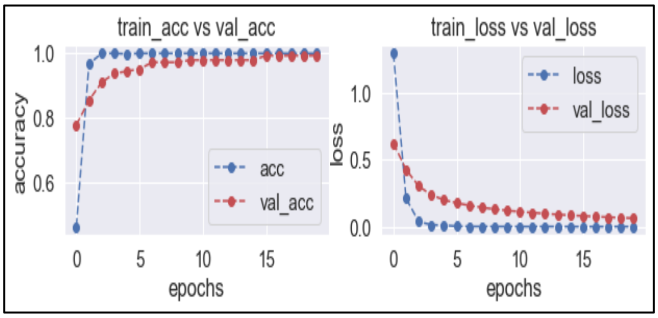
DenseNet201 (FT)

eISSN Number: 3036-017X

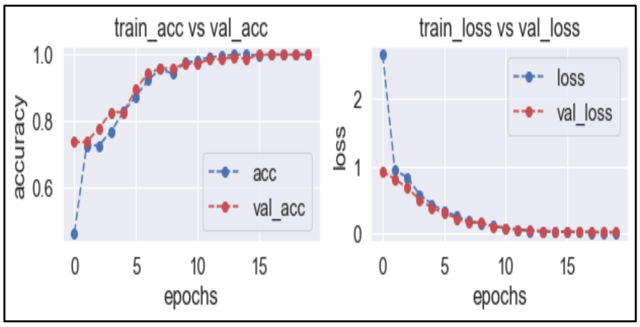
...continue Fig. 5



Xception (TL)



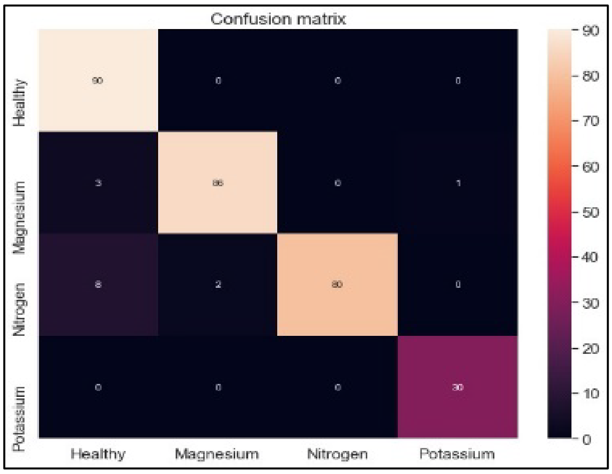
Xception (FT)



CNN

3.4 Assessing the Efficacy of Multiclass SVM (Machine Learning) for Nutrient Deficiency Detection in Oil Palm Seedlings: A Confusion Matrix Approach

The Multiclass SVM model shows strong performance, particularly in identifying true negatives across all classes. This means it is generally good at identifying when a specific deficiency is not present. The false negatives, especially in the nitrogen deficiency class, suggest that the model might sometimes fail to diagnose this condition. This is crucial as it might lead to a lack of treatment. While false positives can lead to unnecessary treatment, false negatives are more critical as they result in a lack of treatment [28]. The model seems to balance these well, but improvements in reducing false negatives, particularly in Nitrogen deficiency, could be beneficial.

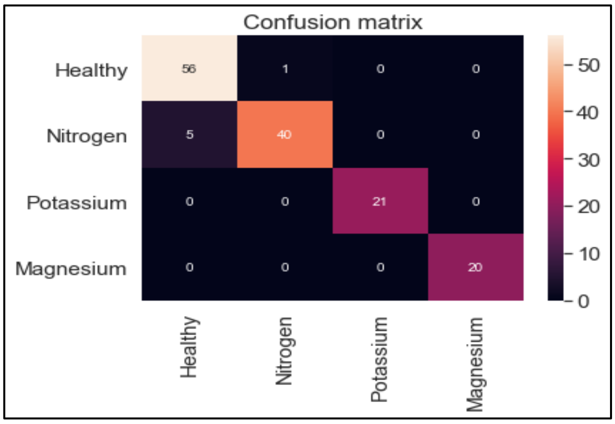


Multiclass SVM

Fig. 6: Confusion Matrix of Multiclass SVM

3.5 Confusion Matrix Analysis of CNN's Capability in Nutrient Deficiency Classification for Oil Palm Seedlings

According to Fig. 7, the CNN model has demonstrated strong classification abilities across all four classes. The absence of FPs and FNs for Potassium and Magnesium deficiencies suggests that the model has distinct and well-segmented features for these conditions, allowing for precise classification. For the Healthy class, the model has shown high accuracy but with a small number of FP, indicating some confusion with Nitrogen deficiency. The FN for the Healthy class is minimal, which is favourable for a classification model since it reduces the risk of missing a diagnosis. The Nitrogen deficiency class, with a few more FN, indicates a challenge for the model in differentiating between Healthy and Nitrogen-deficient cases. However, the very low FP rate is a strong point, suggesting that when the model predicts Nitrogen deficiency, it is likely to be correct. Overall, CNN has shown robust performance, particularly in the Potassium and Magnesium deficiency classes, where it has achieved perfect classification. The performance on the Healthy and Nitrogen deficiency classes is also commendable, though with slight room for improvement in reducing FN.



CNN

Fig. 7: Confusion Matrix of CNN Model

3.6 Comparative Confusion Matrix Analysis of Transfer Learning and Fine-Tuned Models in Nutrient Deficiency Detection in Oil Palm Seedlings

Fig. 8 and 9 presented a comparative analysis of the performance of various pre-trained convolutional neural network models applied to a classification task with two different approaches: Transfer Learning (TL) and fine tuning (FT). The metrics provided are the number of True Positives (TP), False Positives (FP), False Negatives (FN), and True Negatives (TN) for each model across four classes: Healthy, Nitrogen deficiency, Potassium deficiency, and Magnesium deficiency.

According to Fig. 8, VGG16 and VGG19 (TL) models show a somewhat lower performance for the Healthy and Nitrogen classes compared to others, as indicated by higher FP and FN rates. VGG19, in particular, has the highest FP rate for the Healthy class. InceptionV3 (TL) showed a balanced performance with relatively low FP and FN across classes but with slight misclassifications in the Potassium and Magnesium classes. InceptionResNetV2 (TL) is similar to InceptionV3 but with slightly higher FP rates for Potassium and Healthy classes. ResNet152V2 (TL) excels in the Healthy class with no FP and a high TN rate but has some misclassifications in the Nitrogen and Potassium classes. DenseNet201 (TL) has a strong performance, especially in the Healthy class with high TN and no FN, but shows a slight increase in FP for Potassium. Xception (TL) is similar to DenseNet201, as it performs well across all classes with very few misclassifications.

eISSN Number: 3036-017X

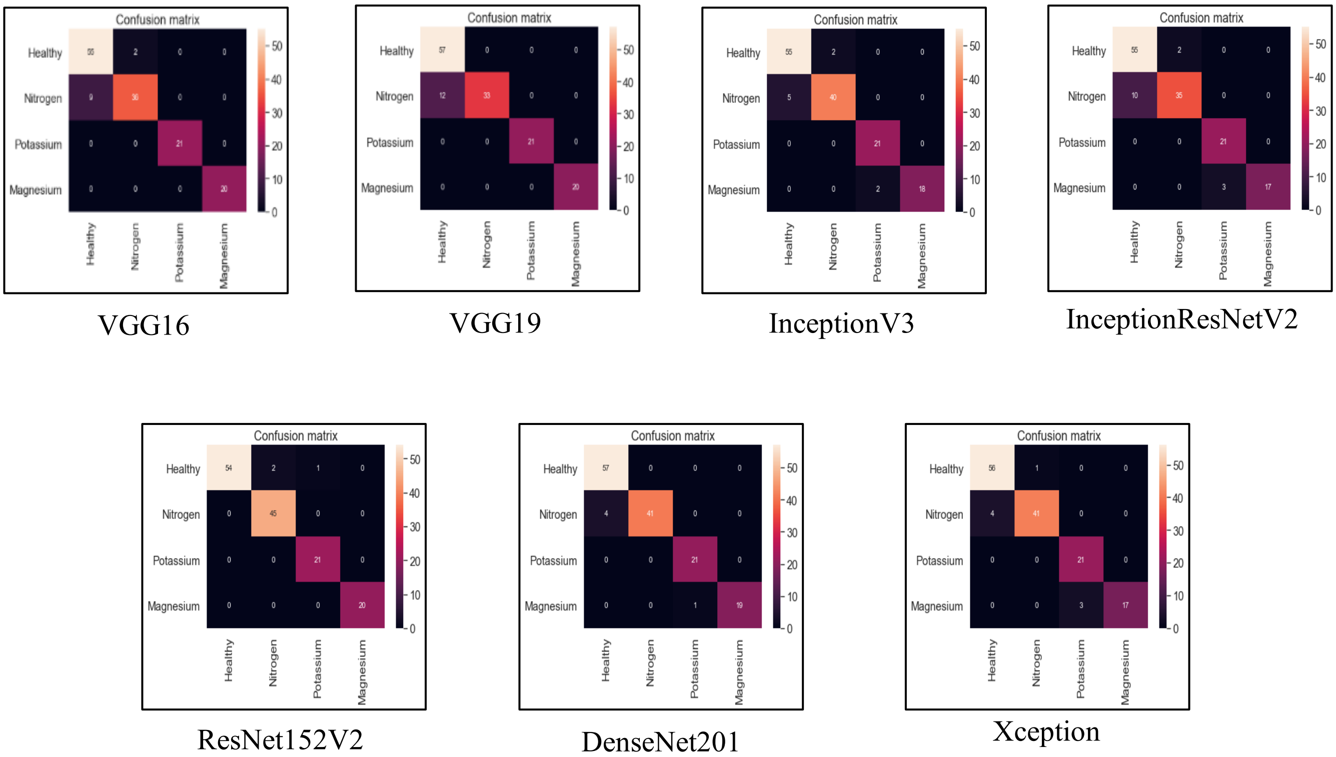


Fig. 8: Confusion Matrix of Transfer Learning Pre-Trained Models

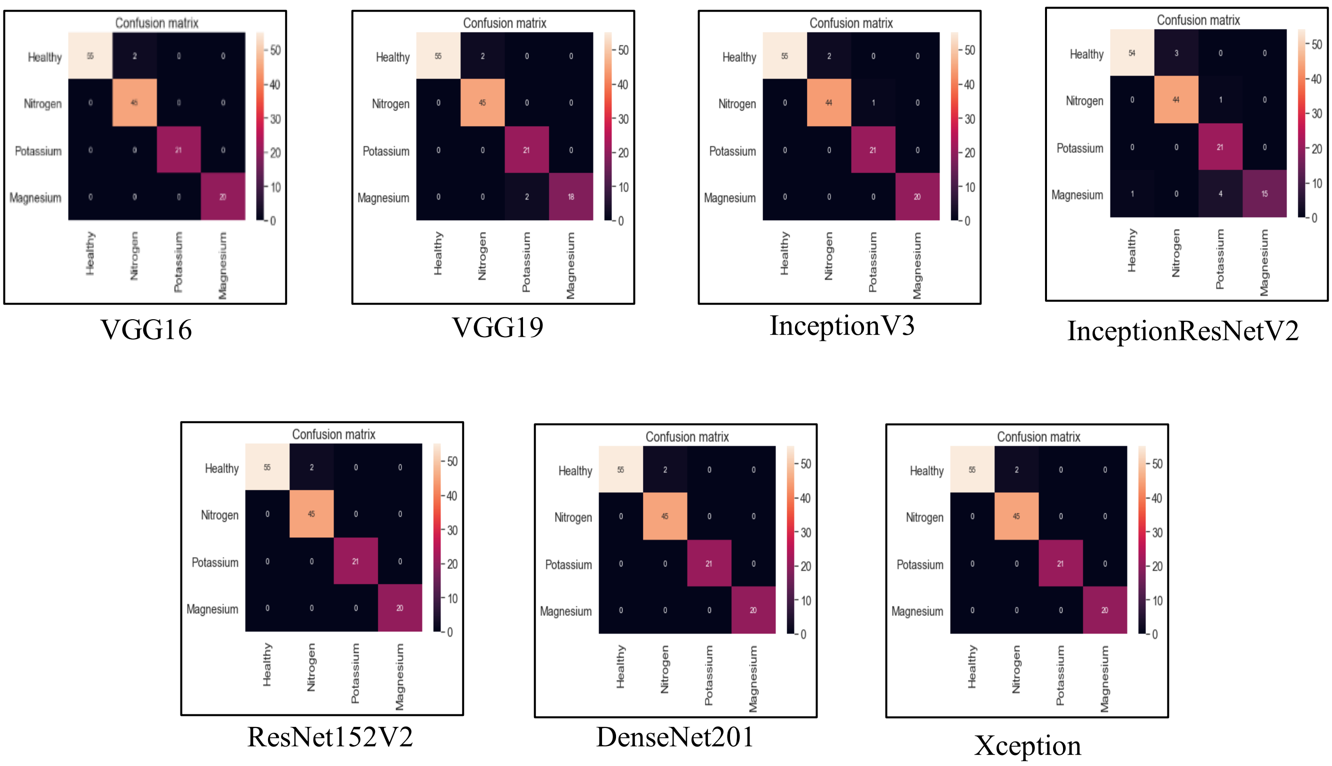


Fig. 9: Confusion Matrix of Fine Tuning Pre-Trained Models

3.7 Comprehensive Model Evaluation

In reviewing the performance of machine learning and deep learning models for the classification of nutrient deficiency in oil palm seedling leaves, the provided data from Fig. 10 allows for an insightful analysis. The Multiclass SVM exhibits a high performance with an accuracy of 95.33%, precision of 0.957283, recall of 0.953333, and F1 score of 0.953333, taking 13 seconds to produce test results. This indicates that SVM is quite reliable, though not the fastest. The CNN model outperforms the SVM slightly in terms of accuracy (95.80%), precision (0.959653), recall (0.958042), and F1 score (0.957777), and it is notably the fastest with a test time of only 1 second, making it particularly suitable for real-time applications. The Multiclass SVM and CNN models exhibit high performance across all metrics, with the CNN slightly outperforming the SVM. The CNN is particularly notable for its speed, being the fastest among all models, which is beneficial for real-time applications.

Among the Transfer Learning (TL) models, VGG16 (TL) shows lower accuracy (92.31%) and F1 score (0.922058) than other models, taking 23 seconds for testing. VGG19 (TL) further lowers in accuracy (91.61%) with an F1 score of 0.913625 and the longest test time of 31 seconds, indicating it may be the least suited for this task within the TL group. InceptionV3 (TL) has better metrics with an accuracy of 93.71%, an F1 score of 0.936796, and a moderate test time of 6 seconds. InceptionResNetV2 (TL) drops in accuracy (89.51%) and F1 score (0.893612) and takes 10 seconds to complete testing. ResNet152V2 (TL) stands out in the TL category with the highest accuracy (97.90%) and F1 score (0.978971), albeit with a longer test time of 21 seconds. DenseNet201 (TL) offers a high accuracy of 96.50% and an F1 score of 0.96485, taking 13 seconds for test results. Xception (TL) has an accuracy of 94.41% and an F1 score of 0.94375, with test results in 9 seconds. Among the Transfer Learning (TL) models, ResNet152V2 (TL) stands out with the highest accuracy and F1 score, suggesting superior performance in both predicting the correct class and maintaining a balance between precision and recall. However, it is also one of the slower TL models. VGG16 (TL) and VGG19 (TL) show lower performance metrics compared to other models, indicating less suitability for this specific task.

Fine-tuning (FT) models generally improve upon TL models. VGG16 (FT), ResNet152V2 (FT), DenseNet201 (FT), and Xception (FT) all achieve very high accuracy and F1 scores of 98.60% and 0.986041, respectively, but with varying test times (26, 23, 12, and 10 seconds, respectively). VGG19 (FT) has a slightly lower accuracy of 97.20% with an F1 score of 0.972005 and a test time of 27 seconds. InceptionV3 (FT) also achieves high accuracy (97.90%) with an F1 score of 0.979093, impressively with the lowest test time of 4 seconds among the FT models, making it highly efficient. InceptionResNetV2 (FT) records an accuracy of 93.71% and an F1 score of 0.93648, taking 10 seconds for testing. Fine-tuning (FT) models generally show improved performance over their TL counterparts, with VGG16 (FT), ResNet152V2 (FT), DenseNet201 (FT), and Xception (FT) all achieving the same high accuracy, precision, recall, and F1 scores. This indicates that fine-tuning has effectively adapted these models to the specific features of nutrient deficiency in oil palm seedling leaves.

It is also evident that fine-tuning can significantly enhance the precision and recall, as demonstrated by the similar high F1 scores of the VGG16 (FT) and ResNet152V2 (FT) models, which are perfect or near-perfect. This indicates that fine-tuning the models on the specific domain of the dataset reduces false positives and false negatives, crucial for accurate diagnosis in agricultural applications. In terms of time efficiency, the fine-tuned InceptionV3 (FT) model is particularly impressive, combining high accuracy with the lowest test time among the fine-tuned models. This suggests that this model could offer a good trade-off between speed and performance for real-world applications where both accuracy and quick results are necessary.

In conclusion, the choice of model would depend on the specific requirements of the application. If speed is crucial, a CNN or an InceptionV3 (FT) might be the best choice. For the best balance between accuracy and computational time, ResNet152V2 (TL) and DenseNet201 (FT) emerge as strong candidates. However, for applications where the highest possible accuracy is required, and computational resources and time are less of a concern, fine-tuned models such as VGG16 (FT), ResNet152V2 (FT), DenseNet201 (FT), and Xception (FT) should be considered.

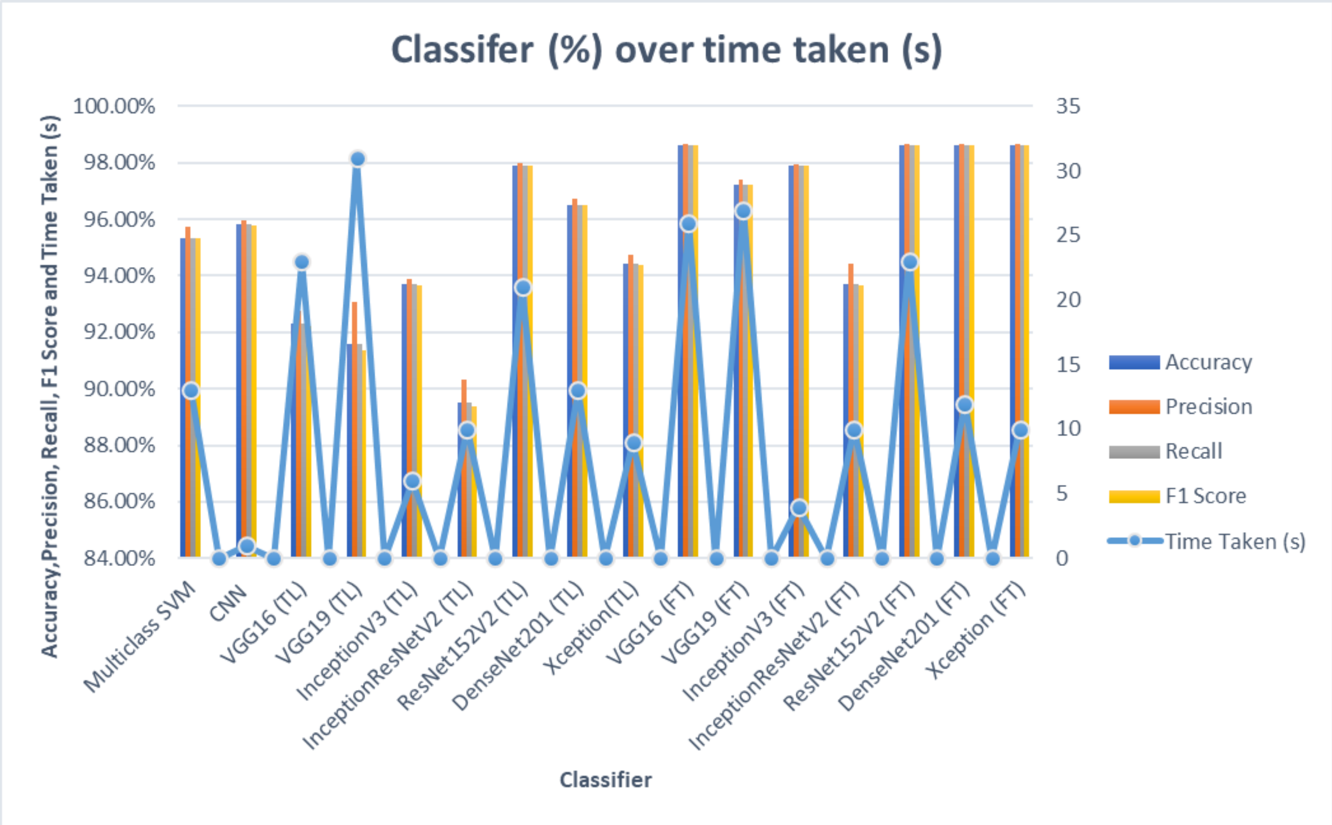


Fig. 10: Comparative Analysis of Model Performance on Classification Task

4. Conclusion

In conclusion, the research highlights the potential of machine learning and deep learning techniques in classifying nutrient deficiencies in oil palm seedling leaves. Deep learning models, particularly the fine-tuned ones, have proven to be highly effective in terms of accuracy and precision. Nevertheless, there is a balance between the complexity of these models and the amount of time they require for computation. This aspect indicates that choosing the most appropriate model depends on the specific priorities, whether it is achieving fast results or ensuring high accuracy. Overall, this research has made notable progress in highlighting the intricacies of oil palm seedling cultivation and its interactions with NPK fertilisers while also utilising technological advancements for accurate nutrient deficiency classifications. Future research could explore the long-term effects of these treatments and investigate whether the initial variations observed have an impact on the overall plant yields as they mature. Further research and optimisation of these models can potentially reveal more advanced solutions, focusing on accuracy and time efficiency in detecting nutrient deficiencies. This will promote a more sustainable approach to oil palm cultivation and support the overall goals of the National Key Economic Areas (NKEA), which aim to ensure the long-term and economically viable growth of the oil palm industry in response to increasing global demand.

Acknowledgement

I want to thank the Faculty of Earth Science, Universiti Malaysia Kelantan, for their continuous support and valuable assistance in this research. Their feedback and recommendations greatly improved the quality of this work. This work was supported by the UMK Prototype Research Grant Scheme under UMK-PRO grant number R/PRO/A0800/01595A/003/2020/00873.

References

[1] Jeyalakshmi S, Radha R. A review on diagnosis of nutrient deficiency symptoms in plant leaf image using digital image processing. ICTACT J Image Video Process, 2017;7(4):1515-1524.

[2] Ong JH, Ong P, Lee WK. Image-based oil palm leaf disease detection using convolutional neural network. J Inf Commun Technol, 2022;21(3):383-410.

[3] Ibrahim S, Hasan N, Sabri N, Abu Samah KAF, Rahimi Rusland M. Palm leaf nutrient deficiency detection using convolutional neural network (CNN). Int J Nonlinear Anal Appl, 2022;13(1):1949-1956.

[4] Srisook N, Tuntoolavest O, Danphitsanuparn P, Pattana-anake V, Joseph FJJ. Convolutional neural network-based nutrient deficiency classification in leaves of Elaeis guineensis Jacq. Int J Comput Inf Syst Ind Manag Appl, 2022;14:19-27.

[5] Djaingsastro AJ, Sinaga H, Sitorus RM. The effect of cocopeat and rice husk planting media hydroponically on the growth of palm oil in pre-nursery. J Biol Lingk Ind Kes, 2021;7(2):195-203.

[6] Felda Agricultural Services Sdn Bhd (FASSB). Manual Ladang Sawit Lestari. 2nd ed., 2012.

[7] Yara International. YaraMila 16-16-16. 2018. https://www.yara.us/crop-nutrition/fertilizer-products/yaramila/yaramila-16-16-16/

[8] Perez L, Wang J. The effectiveness of data augmentation in image classification using deep learning. arXiv, 2017

[9] Ganokratanaa T, Ketcham M, Pramkeaw P. Advancements in cataract detection: The systematic development of LeNet-convolutional neural network models. J Imaging, 2023;9(10):197.

[10] Sharma N, Jain V, Mishra A. An analysis of convolutional neural networks for image classification. Procedia Comput Sci, 2018;132:377-384.

[11] Subasi A. Data preprocessing. In: Practical Machine Learning for Data Analysis Using Python. Elsevier, 2020; p.27-89.

[12] Müller AC, Guido S. Introduction to Machine Learning with Python: A Guide for Data Scientist. O'Reilly Media, Inc, 2016.

[13] Chen R, Jin H, Chen J, Zheng H. AdvCheck: Characterizing adversarial examples via local gradient checking. arXiv, 2023.

[14] Géron A. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. 2nd ed. O'Reilly Media, 2019.

[15] Karim MR, Sewak M, Pujari P. Practical Convolutional Neural Networks: Implement Advanced Deep Learning Models Using Python. Packt Publishing, 2018.

[16] Goodfellow I, Bengio Y, Courville A. Deep Learning. MIT Press, 2016.

[17] Chollet F. Deep Learning with Python. Deep Learning with Python, 2017.

[18] Khandelwal P, Yushkevich P. Domain generalizer: A few-shot meta learning framework for domain generalization in medical imaging. Lecture Notes Comput Sci, 2020;12444:73-84.

[19] Guo Y, Shi H, Kumar A, Grauman K, Rosing T, Feris R. Spottune: Transfer learning through adaptive fine-tuning. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit, 2019:4800-4809.

[20] Almisreb AA, Saleh MA. Transfer learning utilization for banknote recognition: A comparative study based on Bosnian currency. Southeast Eur J Soft Comput, 2019;8(1).

[21] Li H, Chaudhari P, Yang H, Lam M, Ravichandran A, Bhotika R, Soatto S. Rethinking the hyperparameters for fine-tuning. Proc 8th Int Conf Learn Represent, 2020.

[22] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. Proc 3rd Int Conf Learn Represent, 2014

[23] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. IEEE Conf Comput Vis Pattern Recognit, 2016:2818-2826.

[24] Szegedy C, Ioffe S, Vanhoucke V, Alemi A. Inception-v4, inception-ResNet and the impact of residual connections on learning. AAAI Conf Artif Intell, 2017;31(1):4278-4284.

[25] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. IEEE Conf Comput Vis Pattern Recognit, 2016:770-778.

eISSN Number: 3036-017X

[26] Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. IEEE Conf Comput Vis Pattern Recognit, 2017:2261-2269.
[27] Chollet F. Xception: Deep learning with depthwise separable convolutions. IEEE Conf Comput Vis Pattern Recognit, 2017:1251-1258.
[28] Schiffman M, De Sanjose S. False positive cervical HPV screening test results. Papillomavirus Res, 2019;7:184-187.

eISSN Number: 3036-017X