ASEAN Artificial Intelligence Journal 2, Issue 1 (2025) 44-58

KARYA ILHAM PUBLISHING
CONNECTING MINDS, BRIDGING IDEAS, INSPIRING READERS

ASEAN Artificial Intelligence Journal

Journal homepage:
https://karyailham.com.my/index.php/aaij/index
ISSN: 3083-9971

Computer Vision and Artificial Intelligence (AI)-Based Ripeness Classification of Oil Palm Fruits in Oil Palm Plantations

Chong Wei Han[1], Nor Azuana Ramli[1,*], Wan Mohd Rozaimi Wan Mustafa[2], Lilik Jamilatul Awalin[3]

[1] Centre for Mathematical Sciences, Universiti Malaysia Pahang Al-Sultan Abdullah, Kuantan, Pahang, Malaysia
[2] MySpatial Sdn Bhd, MTDC Technology Centre, Universiti Putra Malaysia, Co9P Drone and Geospatial Technology Centre Level 1, Co9P Idea Tower, 1, UPM, 43400 Seri Kembangan, Selangor, Malaysia
[3] Department of Engineering, Faculty of Advanced Technology and Multidicipline, Universitas Airlangga, Jl. Mulyorejo, Surabaya 60115, Indonesia

ARTICLE INFO

Article history:
Received 10 March 2025
Received in revised form 13 May 2025
Accepted 21 May 2025
Available online 3 June 2025

Keywords:
Oil palm; image classification; computer vision

ABSTRACT

Palm oil is one of the most important export products, and Malaysia is the world's second-largest exporter of palm oil. Before the invention of technology, the ripeness of the oil palm fruit bunches was assessed through a traditional method involving personnel with considerable expertise in such identification processes. However, it was very time-consuming and was accompanied by human error that resulted in poor decision-making in the harvesting, thereby cutting down the yields. These challenges are solved in this research through the incorporation of a Convolutional Neural Network (CNN) model, which is in the deep learning (DL) domain of artificial intelligence (AI). The main goal of this research is to establish an AI-based system with the help of Google Teachable Machine (GTM) to classify the ripeness of the oil palm fruit bunches. Data for this research was obtained from Google Images and samples during a site visit in Negeri Sembilan, Malaysia. Posterior performance measures were obtained, and an evaluation was made on the pre-trained model after model training by measuring the confusion matrix and accuracy, as well as accuracy per epoch and the loss per epoch. Before the images are fed to the modelling process, they undergo preprocessing for image enhancement, resizing, and annotation. This research confirmed that GTM could classify the ripeness stage with an overall accuracy of 98%. This research could help shorten the harvesting period and increase the volume of the oil palm fruit bunches produced. It is also intricately linked to the Sustainable Development Goals (SDGs), specifically SDG 12: Responsible Consumption and Production, allowing for mainly proper identification of ripeness, thus enabling less wastage, optimisation of resource use, and support of sustainable agriculture.

1. Introduction

1.1 Research Background

Oil palms and African trees belong to the palm family Aceraceae and are grown for their oil [1]. Approximately 90% of the oil palm trees worldwide are produced in Indonesia and Malaysia, home

* *Corresponding author.*
E-mail address: azuana@umpsa.edu.my

https://doi.org/10.37934/aaij.2.1.4458

to the planet's most biodiverse tropical forests [2]. In addition to those countries, the oil palm is also widely farmed in its native West and Central Africa, such as Nigeria. The oil palm may reach a height of sixty feet or more and have a lifespan of at least one hundred years.

In the context of oil palm agriculture, one of the critical determinants of the quality and yield of palm oil products is the ripeness of the fruit bunches. The oil palm tree, which can only thrive in tropical regions, produces high-quality oil mainly used for cooking in developing countries. Moreover, a limited amount of biofuel is also made with it, along with food items, detergents, and cosmetics. The seeds' palm kernel oil makes numerous medications and gastronomic goods, including ice cream, margarine, biscuits, bread, and chocolate confections [1]. Each oil palm fruit comprises a hard kernel or seed encased in an endocarp shell encircled by a soft mesocarp [3].

In Malaysia, 5.65 million hectares of land were used for the oil plant plantation in 2023 [4]. Although it was slightly less than 2024, it is still the second-largest palm oil producer in the world after Indonesia. It produced 2.20 tonnes of palm kernel oil and 19.14 million tonnes of palm oil. In addition, the palm oil sector generated RM 73.25 billion in export earnings and contributed 3.6% of Malaysia's Gross Domestic Product (GDP). Malaysia is now the second-largest producer and exporter of palm oil worldwide, thanks to palm oil. In 2023, Malaysia holds a 25.8% and 34.3% share in the global production and export of palm oil, respectively. These data prove the immense contribution of oil palm to the Malaysian economic sector. It also shows that the oil palm industry is pivotal in agriculture.

The oil palm sector has traditionally relied on manual inspection by skilled workers for ripeness assessment. Conventional techniques: the number of loose fruits from the fresh fruit bunches and using human eyesight to observe the colour changes during ripening are ineffective, subjective, and labour-intensive [5]. It might be challenging for observers to identify ripe fruit in towering trees due to the distance and sunlight [6]. This traditional method consumes a lot of time and worker energy. Additionally, the intricate features of oil palm fruit, such as the uneven colour of mature fruit, the fruit's appearance in small bunches, and the varying stages of fruit maturity in various types, have prevented them from getting satisfying results [6]. Therefore, it is a must to identify the diversity of the oil palm fruit ripeness stages.

There are a few methods through deep learning to classify the ripeness stages of the oil palm fruit bunches in developing the model, including the Convolutional Neural Networks (CNN) family and the You Only Look Once (YOLO) family. Each of these has its advantages. CNN is an algorithm in image processing methods that works well for image identification and object recognition [7]. R-CNN may be used for real-time object identification because it is faster than its predecessors, which is a significant advance [8]. Meanwhile, YOLO provides a one-time CNN to categorise numerous candidates and frame position prediction [8].

YOLO is one of the most accurate and exact object identification algorithms available today [8]. YOLO has been used to detect objects designed for real-time processing [9]. The CNN and YOLO families have the requirement for cohesive networks in common [10]. In short, R-CNN solves the detection outcomes, which are object categories and object positions, while YOLO is unified as a regression issue [10]. Apart from that, Faster R-CNN is the ideal option for dealing with a small dataset and does not require real-time results [8]. Otherwise, YOLOv3 is the best option for studying a live video feed. Since YOLO has a few upgraded versions, YOLOv5, YOLOv6, YOLOv7, and YOLOv8 have been applied in a similar study. For instance, a study by [11] utilised YOLOv3, YOLOv4, and YOLOv5m, a deep learning technique for oil palm plant detection. In another study by [12], CNN was used to identify fresh fruit bunches of oil palm with a 98% high accuracy value during the training phase and 76% during the model testing phase. CNN has proven to have excellent accuracy and precision in detection and classification. However, YOLO can identify objects in real-time with high accuracy.

Hence, this study was conducted to achieve these objectives:

i. To identify the ripeness stages of oil palm fruit bunch using Google Images and real-life datasets from a site visit.
ii. To classify oil palm fruit bunch at different ripeness stages using the Google Teachable Machine (GTM).
iii. To evaluate the model's performance based on the confusion matrix, accuracy, accuracy per epoch, and loss per epoch.

In a nutshell, this research explores the need to fill the gap between conventional methods of manually assessing the ripeness of fruits and other products and the enhancement that new approaches, such as deep learning, can offer. It aims to enhance the efficiency of ripeness classification, which will explore the optimisation of innovative technology in the oil palm sector, and, on top of that, strives to improve the sustainable practices that involve production and consumption while drawing on the gains of essential agriculture.

2. Methodology

2.1 Data Collection

The data collection consists of two datasets: Google Images and a site visit at an oil palm plantation in Negeri Sembilan, Malaysia. The data from the site visit is considered primary data as the image needs to be captured manually. Meanwhile, the dataset from Google Images is considered secondary data where the data is readily available. During the site visit, the iPhone 14 was used to snap high-resolution photographs of oil palm fruit bunches. These images covered various ripeness stages of oil palm fruit bunches.

During the site visit, a few challenges were encountered while collecting data. The first challenge was related to the background. Capturing images of the oil palm fruit was difficult due to the lush greenery surrounding it, which created intricate and crowded backdrops in the photos. The second challenge involved the colour complexity of the oil palm fruit bunch, which made it difficult to determine the ripeness stage of the fruit and potentially introduced noise into the dataset. Lastly, the height of the oil palm trees posed a challenge in capturing images of the fruit bunches.

2.2 Image Preprocessing

Image preprocessing is crucial because the model's accuracy would be affected without it. Preprocessing was applied to the gathered datasets to standardise the image format, resolution, and labelling. This step is essential in ensuring consistency in the input data for model testing. The dataset's quality significantly impacts the final model's performance. The image preprocessing involves two parts: the first part is on the Google Images dataset before creating the pre-trained model, and the second part is on the site visit dataset after exporting the pre-trained model.

The next step is image extraction, where the dataset from Google Images was extracted and loaded into Roboflow. Once the images were extracted, the enhancement was implemented on the datasets in Roboflow. These parts may include editing the contrast or brightness, as shown in Figure 1. Next, images were labelled to train machine learning models, and this process is known as image annotation. Image labelling aims to locate and classify certain aspects within a picture. An image's labelling was classified with the ripeness stages of the oil palm fruit bunches. Figure 2 indicates that the ripe oil palm fruit bunch was added to the "ripe" class.

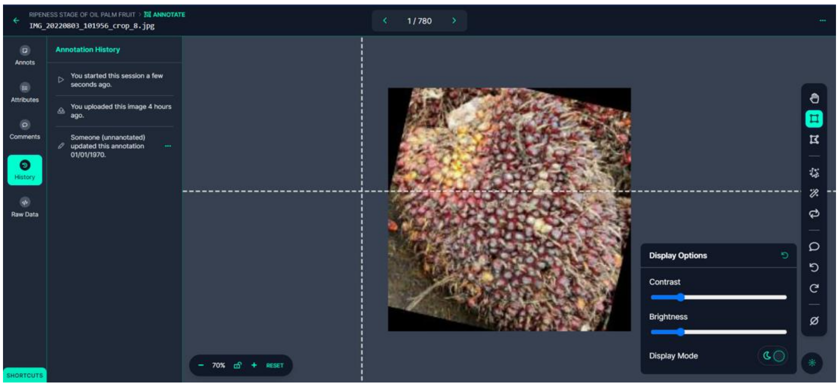


Fig. 1. Image enhancement

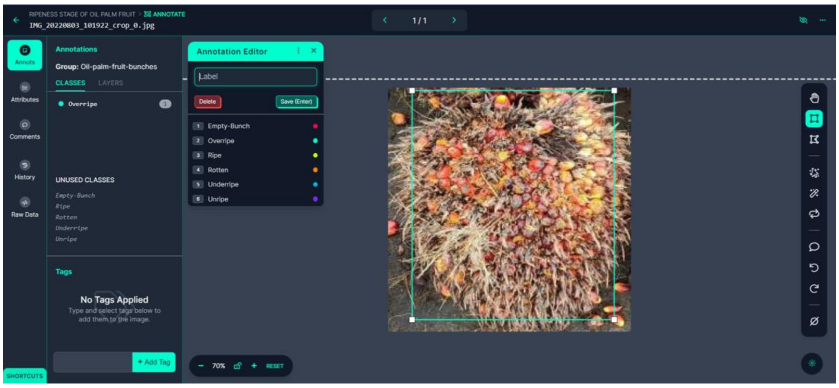


Fig. 2. Labelling

Image resizing was implemented in this research. The image data of oil palm fruit bunches was resized to 640x640 pixels by stretching, as shown in Figure 3. This resizing ensures consistency and suitability for the model's input layer.

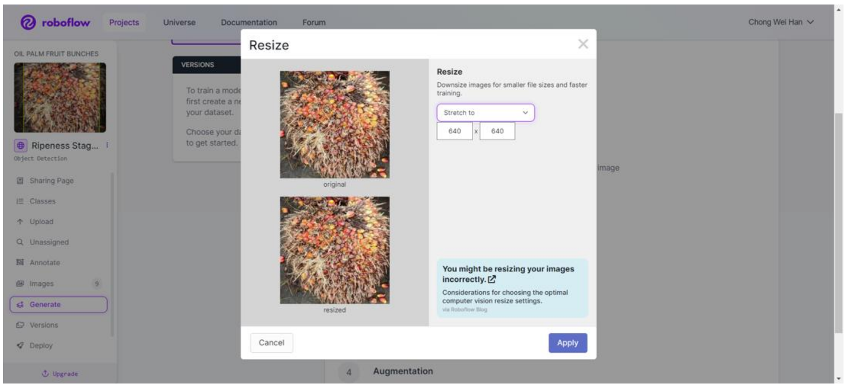


Fig. 3. Image resizing

The second part of the image preprocessing involves exporting the pre-trained model using Google Collab for image resizing and normalisation. The site visit dataset was resized to a fixed size of 224x224 pixels using the Lanczos resampling method. This method is commonly used for scaling digital images, as it enhances the sampling rate of a digital signal or allows for movement by a fractional portion of the sampling interval. Additionally, if any images were not in RGB format, they were converted to RGB.

Image normalisation is a crucial step in the image preprocessing stage. After resizing, the pixel values of each image are normalised. This process modifies the pixel values to improve the learning efficiency of machine learning models. Specifically, normalisation involves dividing the value of each pixel by 127.5 and then subtracting 1. The primary goal of this process is to stabilise training and enhance the convergence of neural networks.

2.3 Model Development

A Google Teachable Machine (GTM) was utilised in this research to classify the different ripeness stages of oil palm fruit bunches. It is a free tool and an online platform developed by Google that allows users to generate deep learning models without coding. This tool utilises TensorFlow.js to provide neural network training and inference from inside the browser [13]. The underlying mechanism of the GTM relies on a widely used deep learning approach known as transfer learning [14]. Transfer learning involves reusing a pre-existing model to perform another task [15], allowing faster results and fewer data requirements. It also consists of modifying an existing model's last layer to recognise images' characteristics.

Pre-trained models are deep learning models trained on extensive datasets to perform specific tasks. They are applicable across many industries and can be used as is or modified to meet specific requirements. Pre-trained models are driving the progress of AI in various domains, including speech AI, Natural Language Processing (NLP), computer vision, cybersecurity, healthcare, and workflows. Although the architecture related to the GTM could not be found in research papers, a few online sources have proven that the GTM could be built using MobileNet in TensorFlow.js through the transfer learning approach.

GTM uses MobileNet as the foundation for its transfer learning process [16]. MobileNet is a low-latency, lightweight neural network optimised for tiny devices. The training process is characterised by relatively fast execution times, and it may be started with a smaller number of pictures. MobileNet is a simplified structure that utilises Depthwise Separable Convolutions (DSC) to create an efficient deep CNN [17]. It is specifically designed for mobile and embedded vision applications, making it suitable for these tasks. MobileNet is extensively used in several practical applications, such as object identification, facial attributes, localisation, and fine-grained classifications [18]. MobileNet architecture is constructed using DSC, as shown in Figure 4.

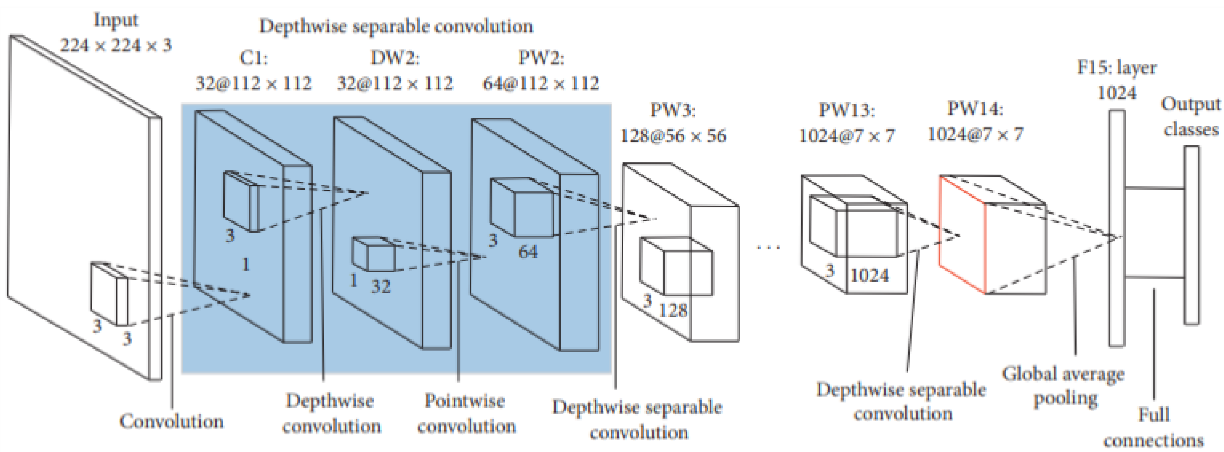


Fig. 4. Architecture of MobileNet [8]

For model development using GTM, the process started with uploading the dataset based on its class. Then, the model was trained and tested before it was exported to Python code. For this study, hyperparameters were set by default using 50 epochs, 16 batch sizes, and a learning rate of 0.001. Epoch refers to the total number of iterations the training algorithm performed on the training dataset. Batch size refers to the number of training samples processed by the algorithm before changing the model's internal hyperparameters. This hyperparameter controls the step size of the algorithm's hyperparameter updates. Opting for an excessively high learning rate may result in model instability, while a small learning rate may significantly prolong the training process. A learning rate of 0.001 corresponds to a suitable value, indicating that the algorithm modified its weights slightly throughout each iteration.

Finally, the model will be evaluated through a comprehensive evaluation process to determine whether it is good enough to be deployed in the system by exporting or downloading the model. The metrics used for model evaluation will be discussed in the following subsection. If the evaluation result is unsatisfactory, the hyperparameters can be fine-tuned for better performance.

2.4 Model Evaluation

Once the model is exported to the Python environment, the performance of the developed model can be assessed through a comprehensive evaluation process. The metrics such as confusion matrix, precision, recall/ sensitivity, mean average precision (mAP), and losses are used to measure the model's overall performance. Figure 5 shows the formula for sensitivity, specificity, accuracy, precision, and negative predictive value that can be obtained from the confusion matrix.

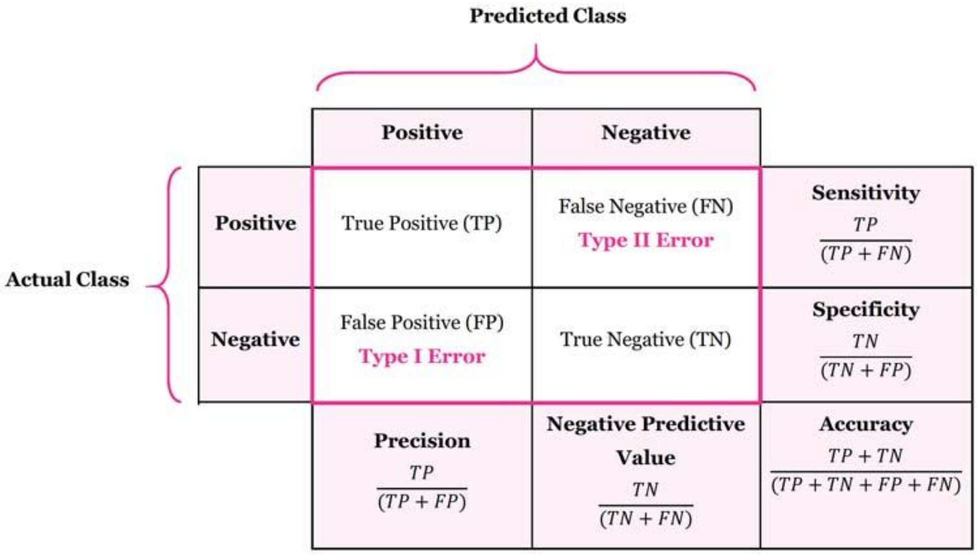


Fig. 5. Confusion Matrix [19]

A loss function is a tool used to guide a model toward convergence, indicating its proximity to making the correct prediction. It describes the model's "badness," with a smaller value indicating better performance. Categorical Cross Entropy (CCE) is one of the types of loss functions and an alternative term for Softmax Loss. The model employs a Softmax activation function in combination with a Cross-Entropy loss function for multi-class classification. By using this loss function, it is possible to train a Convolutional Neural Network (CNN) to generate a probability distribution over the *N* classes for every picture. Eq. (1) shows the CCE formula for each image, where *N* is the number of classes, y_i represents actual values, and $\bar{y}$ refers to predicted values [20].

$$CCE = -\sum_{i=1}^{i=N} y_i \cdot \log(\bar{y}_i) \tag{1}$$

This research involved two parts in the model evaluation: training the model on GTM and the output after inserting the site visit dataset. First, the evaluation result can be viewed on the GTM by choosing "Under the hood" in the Advanced options once the pre-trained model has completed the training. The evaluation result includes accuracy per class, confusion matrix, accuracy per epoch, and loss per epoch, as shown in Figure 6.

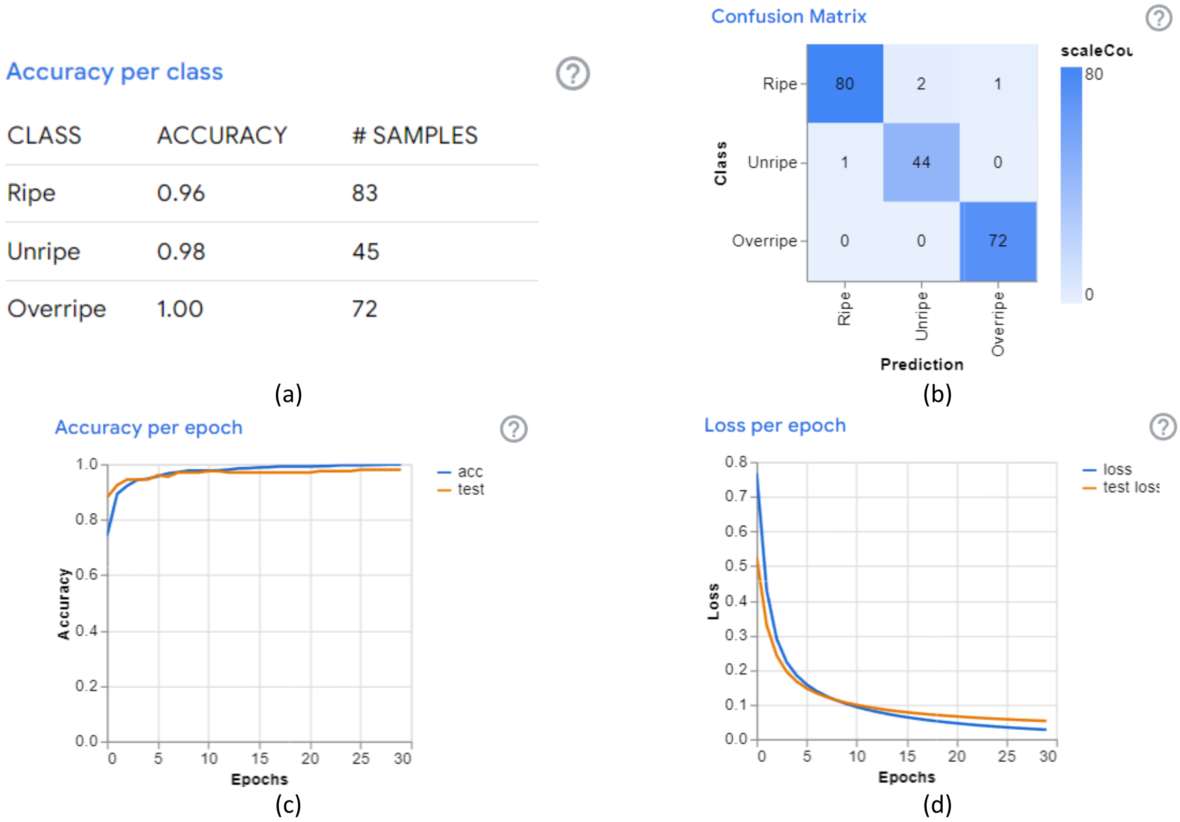


Fig. 6. Evaluation result (a) Accuracy per class (b) Confusion matrix (c) Accuracy per epoch (d) Loss per epoch

For the second part, since the confusion matrix is in multi-class classification, it needs to be calculated using the function of "confusion_matrix" and computing the classification report, as shown in Figure 7. The classification report includes metrics such as precision, recall, F1-score, and support.

```python
from sklearn.metrics import classification_report
import numpy as np

# Define the confusion matrix
confusion_matrix = np.array([[80, 2, 1],
                             [1, 44, 0],
                             [0, 0, 72]])

# Create the true and predicted labels based on the confusion matrix
true_labels = []
pred_labels = []

for i, row in enumerate(confusion_matrix):
    for j, count in enumerate(row):
        true_labels.extend([i] * count)
        pred_labels.extend([j] * count)

# Define the target class labels
target_names = ['Ripe', 'Unripe', 'Overripe']

# Compute the classification report
report = classification_report(true_labels, pred_labels, target_names=target_names)

print(report)
```

Fig. 7. Compute the classification based on the confusion matrix

3. Result and Discussion

3.1 Preliminary Analysis

Three ripeness stages, including ripe, overripe, and unripe, are used in this research to classify oil palm fruit ripeness stages. This research used two image datasets, including the dataset from Google Images with 1324 images and a site visit dataset with 30 images. The number of images for each ripeness stage is stated in Table 1. The Google Images dataset was used for the pre-trained model purposes; meanwhile, the site visit dataset was used for ripeness classification purposes using the pre-trained model. Table 2 shows the different ripeness stages of the dataset from Google Images, and Table 3 illustrates the ripeness stage of the site visit dataset before classification.

Table 1
Number of images for each ripeness stage

Ripeness stages	Number of images	
	Google images	Site visit
Ripe	550	-
Overripe	477	-
Unripe	297	-
Total	1324	30

Table 2
Different ripeness stages of the Google Images dataset

Ripeness stages	Google Images dataset
Ripe	
Unripe	
Overripe	

Table 3
Ripeness stages of site visit dataset before classification

Site visit dataset


3.2 Results from Analysis
3.2.1 Google Images

One advantage of GTM is that it splits the training data (85%) and testing data (15%) before training the model by default. The number of testing datasets from Google Images for each stage of ripeness was mentioned earlier in Table 1. Figure 8 shows the data partitioning between the training and the testing dataset. Based on the result, the unripe stage has the least sample dataset among the three stages, and the ripe stage has been classified as the highest number of images.

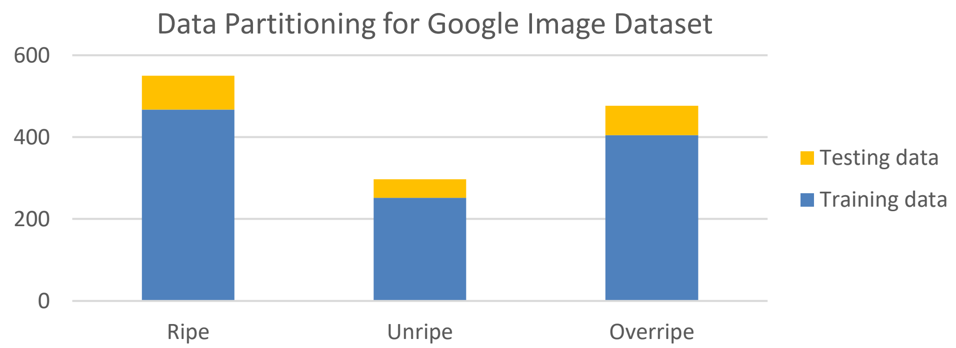


Fig. 8. Data partitioning for Google image dataset

Overall, there is a slightly imbalanced training dataset. This is because the unripe stage in another dataset from the online sources only has a few numbers of datasets, which is the least number of stages in another dataset. An imbalanced dataset in the CNN model might lead to suboptimal performance for the class with fewer instances, affecting the overall performance. Hence, it is crucial to get a balanced dataset to achieve better performance.

Figure 9 shows the confusion matrix for the pre-trained model after multiple attempts to fine-tune epochs, batch size, and learning rate. Based on the confusion matrix, the model correctly classified the ripeness stages as ripe, unripe, and overripe with 80, 44, and 72 images, respectively. For the ripe stage, it was incorrectly classified as unripe with two images and misclassified as overripe with an image. Only an image misclassification is ripe for the unripe stage; meanwhile, there is no misclassification for the overripe stage as ripe and unripe.

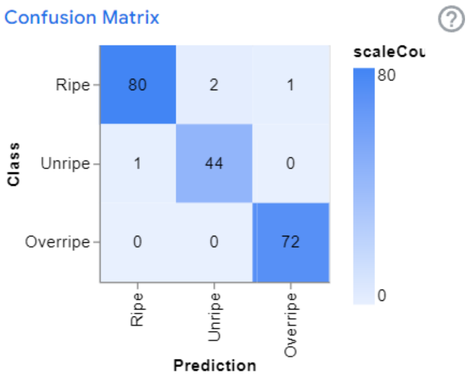


Fig. 9. Confusion matrix

Figure 10 shows the accuracy for each class by using the testing data after training the dataset in the pre-trained model. The result indicates that each class achieves high accuracy, scoring 0.90 or above. The overripe oil palm fruit has the highest accuracy (1.00) among the ripeness stages. Meanwhile, the unripe stage has the second-highest accuracy (0.98), followed by the ripe stage (0.96).

Accuracy per class

CLASS	ACCURACY	# SAMPLES
Ripe	0.96	83
Unripe	0.98	45
Overripe	1.00	72

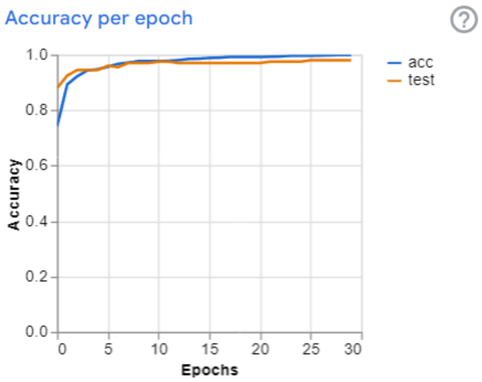
Fig. 10. The accuracy of each class

Based on the classification report in Figure 11, the ripe and overripe stages have the same highest value with 99% precision; meanwhile, the unripe stage only has 96%. For recall, the highest value was overripe stage (100%), followed by unripe stage (98%), and the lowest was ripe stage (96%). For the F1-score, the ranking is slightly different from the recall. The highest value was the overripe stage (99%), followed by the ripe stage (98%), and the lowest was the unripe stage (97%). Overall, the results indicate that the classifier performs excellently in all stages, with notably high accuracy in classifying the overripe stage and consistently good precision and recall across all ripeness stages. The model also has a high level of accuracy, with an overall rate of 98%, suggesting its robust and reliable performance.

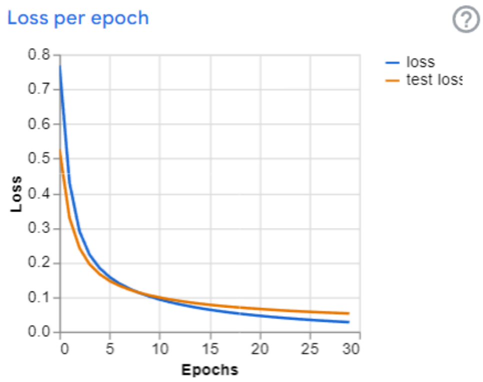
```
              precision    recall  f1-score   support

        Ripe       0.99      0.96      0.98        83
      Unripe       0.96      0.98      0.97        45
    Overripe       0.99      1.00      0.99        72

    accuracy                           0.98       200
   macro avg       0.98      0.98      0.98       200
weighted avg       0.98      0.98      0.98       200
```

Fig. 11. Classification report

After being fine-tuned multiple times, the performance of accuracy per epoch and loss per epoch is shown in Figure 12. The accuracy and testing accuracy increased to nearly 0.9 at three epochs and were maintained at about 0.9 until the end of 30 epochs. Meanwhile, the loss and testing loss decreased drastically to below 0.1 at ten epochs and slowly reduced to 0.05 at the end of 30 epochs. Overall, the accuracy of the epoch remains at 0.9 above, and the loss per epoch remains below 0.1. The results indicate that the pre-trained model performed well.



(a)



(b)

Fig. 12. (a) Accuracy per epoch (b) loss per epoch

3.2.2 Site Visit Dataset

After the dataset was inserted into the pre-trained model, the result was stored in a zip file, which needed to be extracted after extraction, as shown in Figure 13. 30 images were randomly captured from the Negeri Sembilan site visit. Figure 13 depicts that most site visit datasets are ripe and unripe, while overripe only has a few images.

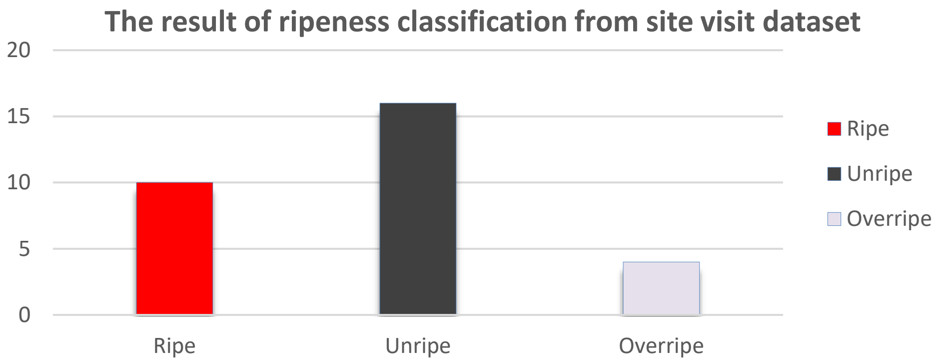


Fig. 13. The result of the ripeness classification dataset from the site visit

As mentioned earlier, some problems were faced when capturing the dataset, such as the height of the oil palm tree. Moreover, the lack of ripe and overripe stages of oil palm fruit also made the data collection more challenging. The images, such as ripe and overripe oil palm fruit from online sources, were combined with the site visit dataset to fix the problem. Hence, there was enough data to test the pre-trained model.

Table 4 shows some of the sample outputs. Overall, the output with the ripeness classification is good, but some inaccurate classifications exist. For example, the first three images in Figure 14 should be overripe, but the model classified them as ripe in a ripe folder. This could be due to the dataset's lack of various angles. The limitation of various angle training datasets from Google Images leads to misclassification.

Table 4
Ripeness stages of site visit dataset after classification

Ripeness stages	Google Images dataset
Ripe	
Unripe	
Overripe	

Fig. 14. The output classification of the ripeness stage of oil palm fruit bunch

4. Conclusions

The technology used in the research has the potential to replace manual ripeness assessment by significantly reducing the harvesting time of oil palm fruit bunches. Additionally, it minimizes human error and effort, making the process more efficient. Given the limitations of existing approaches in assessing the ripeness stage of the fruits, this research proposed the design, implementation, and validation of a deep learning model for oil palm fruit bunch classification. The model's performance was evaluated using the confusion matrix, accuracy, accuracy per epoch, and loss per epoch.

This study successfully identified the ripeness stages of oil palm fruit bunches using a dataset comprising 1324 images from Google Images and 30 images from real-life site visits in Negeri

Sembilan. Preprocessing steps were applied to improve model generalization, including image enhancement, resizing, and annotation using RoboFlow. A deep learning model was developed using Google Teachable Machine (GTM), a user-friendly online platform for image classification. Hyperparameters such as batch size, number of epochs, and learning rate were fine-tuned to enhance accuracy. The trained model achieved high classification accuracy, with 96% for ripe, 98% for unripe, and 100% for overripe fruit bunches, resulting in an overall accuracy of 98%. Despite these results, minor misclassifications between ripe and unripe stages were observed, likely due to dataset limitations. Performance metrics, including the confusion matrix, accuracy per epoch, and loss per epoch, provided insights into the model's effectiveness.

While previous studies employed deep learning models such as YOLOv4, YOLOv5, MobileNetV1/V2, AlexNet, and CNN-based oil palm fruit classification techniques, some key research gaps remain. Most studies relied on computationally intensive models, whereas this research demonstrated the effectiveness of GTM for oil palm classification, which is less explored in existing research. Additionally, many studies did not emphasize preprocessing techniques to enhance model generalization, while this research improved dataset quality using RoboFlow. Unlike prior studies focusing solely on accuracy, this study highlights the potential for deploying the GTM model on mobile devices, drones, and automated harvesting systems. Furthermore, while previous studies reported mAP, F1-score, and recall, this study emphasized accuracy per epoch and loss per epoch, providing insights into model training behavior. Existing research primarily focused on technical accuracy, while this study linked AI-based classification to supply chain efficiency, post-harvest loss reduction, and sustainability in the palm oil industry.

This research bridges the gap by providing an efficient, accessible AI-based classification system while addressing dataset, usability, and deployment challenges. Future work should explore larger datasets, multi-source data fusion, and real-time IoT integration for improved classification accuracy and practical implementation. Accurate ripeness assessment is essential to minimize post-harvest losses, optimize shelf space and energy use, and promote sustainable agricultural practices, aligning directly with the Sustainable Development Goals (SDGs).

Acknowledgment
This research was not funded by any grant. However, the dataset for this study was provided by MySpatial Sdn. Bhd.

References
[1] Britannica. "Oil Palm." Encyclopedia Britannica, September 17, 2021.
[2] Muhammad Hanafi Asril Rajo, Ahmad Faiz Mat Zin, and Norzelawati Asmuin. "Palm Fatty Acid Distillate (PFAD) as Source of Biofuel: Availability and Suitability Analysis." *Journal of Advanced Research in Fluid Mechanics and Thermal Sciences* 76, no. 1 (2024): 156–163. https://doi.org/10.37934/arfmts.76.1.156163
[3] Zolfagharnassab, Shahrzad, Abdul Rashid Bin Mohamed Shariff, Reza Ehsani, Hawa Ze Jaafar, and Ishak Bin Aris. "Classification of oil palm fresh fruit bunches based on their maturity using thermal imaging technique." *Agriculture* 12, no. 11 (2022): 1779. https://doi.org/10.3390/agriculture12111779
[4] Statista. "Total Planted Areas for Palm Oil in Malaysia from 2014 to 2023." Statista, January 2024.
[5] Shiddiq, Minarni, Fitmawati, Ridho Anjasmara, Nurmaya Sari, and Hefniati. "Ripeness detection simulation of oil palm fruit bunches using laser-based imaging system." In *AIP conference proceedings*, vol. 1801, no. 1, p. 050003. AIP Publishing LLC, 2017. https://doi.org/10.1063/1.4973101
[6] Suharjito, Franz Adeta Junior, Yosua Putra Koeswandy, Debi, Pratiwi Wahyu Nurhayati, Muhammad Asrol, and Marimin. "Annotated datasets of oil palm fruit bunch piles for ripeness grading using deep learning." *Scientific Data* 10, no. 1 (2023): 72. https://doi.org/10.1038/s41597-023-01958-x
[7] Sunyoto, Andi, and Alva Hendi Muhammad. "Detection of Palm Fruit Maturity Using Convolutional Neural Network Method." *JAIA-Journal of Artificial Intelligence and Applications* 2, no. 2 (2022): 33-37. https://doi.org/10.33372/jaia.v2i2.859

[8] Srivastava, Shrey, Amit Vishvas Divekar, Chandu Anilkumar, Ishika Naik, Ved Kulkarni, and V. Pattabiraman. "Comparative analysis of deep learning image detection algorithms." *Journal of Big data* 8, no. 1 (2021): 66. https://doi.org/10.1186/s40537-021-00434-w

[9] Hui, Jonathan. "Real-time object detection with yolo, yolov2 and now yolov3."(2018).

[10] Cheng, Richeng. "A survey: Comparison between Convolutional Neural Network and YOLO in image identification." In *Journal of Physics: Conference Series*, vol. 1453, no. 1, p. 012139. IOP Publishing, 2020. https://doi.org/10.1088/1742-6596/1453/1/012139

[11] Wibowo, Hery, Imas Sukaesih Sitanggang, Mushthofa Mushthofa, and Hari Agung Adrianto. "Large-scale oil palm trees detection from high-resolution remote sensing images using deep learning." *Big Data and Cognitive Computing* 6, no. 3 (2022): 89. https://doi.org/10.3390/bdcc6030089

[12] Ashari, Suci, Gomal Juni Yanris, and Iwan Purnama. "Oil palm fruit ripeness detection using Deep Learning." *Sinkron: jurnal dan penelitian teknik informatika* 6, no. 2 (2022): 649-656. https://doi.org/10.33395/sinkron.v7i2.11420

[13] AI:How to Use Teachable Machine to Create an Image Classification Application on STM32 - Stm32mcu. 2025. Stmicroelectronics.cn. 2025.

[14] Janakiram MSV. "Teachable Machine from Google Makes It Easy to Train and Deploy ML Models." Forbes, November 29, 2020.

[15] Gillis, Alexander S., and Ben Cole. "What Is Transfer Learning?" CIO, September 2023.

[16] Siddiqui, Nabeel. "Creating Deep Convolutional Neural Networks for Image Classification." *The Programming Historian* (2023). https://doi.org/10.46430/phen0108

[17] Wang, Wei, Yutao Li, Ting Zou, Xin Wang, Jieyu You, and Yanhong Luo. "A novel image classification approach via dense-MobileNet models." *Mobile Information Systems* 2020, no. 1 (2020): 7602384. https://doi.org/10.1155/2020/7602384

[18] Pa, S. "An overview on mobilenet: an efficient mobile vision CNN." *medium. com, Jun* 10 (2020).

[19] Sirsat, M. "Data Science and Machine Learning: Confusion Matrix." *Data Science and Machine Learning* (2019).

[20] Neuralthreads. "Categorical Cross-Entropy Loss — the Most Important Loss Function." Medium, December 26, 2021.