EVOLUTION IN ELECTRICAL AND ELECTRONIC ENGINEERING
e-ISSN: 2756-8458

EEEE

Vol. 6 No. 1 (2025) 34-42
https://publisher.uthm.edu.my/periodicals/index.php/eeee

Mobile Image Processing to Enhance Visual Accessibility for Pineapple Maturity Indices

Muhamad Zufar Aziz[1], Ain Nazari[1*] , Siti Noor Aliah Baharom[2] , Badril Hisham Abu Bakar[2] , Mohd Nizam Zubir[2],Mohd Taufik Ahmad[2]

[1] *Faculty of Electrical and Electronic Engineering*
Universiti Tun Hussein Onn Malaysia, Batu Pahat, Johor, 86400, MALAYSIA

[2] *Engineering Research Centre*
MARDI Head Quarters, Serdang ,43400, Selangor, Malaysia

*Corresponding Author: ain@uthm.edu.my
DOI: https://doi.org/10.30880/eeee.2025.06.01.005

Article Info

Received: 10 July 2024
Accepted: 25 April 2025
Available online: 9 May 2025

Keywords

Deep learning, Object Detection, Classification, MD2 Pineapple, YOLOv5s, YOLOv7s, YOLOv8s

Abstract

The primary objective is to employ advanced image processing techniques capable of accurately distinguishing between the various stages of pineapple maturation. To achieve this, the study integrates cloud computing into the process, enabling seamless synchronization of annotated data and expediting the training of a cutting-edge object detection model. The methodology introduced in this research involves the incorporation of an updated model of object detection tools, aiming to elevate the precision and efficiency of maturity index differentiation. Leveraging the power of cloud computing infrastructure ensures that the model is continually refined and improved through a cyclical process of annotation, training, and deployment. Furthermore, the study extends its impact beyond the research domain by developing a userfriendly mobile application. This application serves as a practical interface, allowing users, including farmers and agricultural stakeholders, to conveniently assess pineapple maturity on-site. The highest accuracy model is deployed within the mobile, which is the YOLOv8 model with mAP50 value of 0.741 ensuring that end-users benefit from the cutting-edge technology without the need for specialized expertise in image processing or data analytics. Pre -trained model is implemented during app development process by recalling the model parameter. In conclusion, this interdisciplinary approach, combining mobile image processing, cloud computing, and user-centric application development, holds substantial promise for revolutionizing pineapple production and, by extension, contributing to the advancement of precision agriculture.

1. Introduction

The market for Malaysian MD2 pineapple has surged dramatically this year because of China's limitations on Taiwanese pineapple. Malaysian MD2's price has risen by 10% because to increased demand in China and the rest of the region. The Malaysian Pineapple Industry Board (MPIB) has acknowledged this and announced a 50% increase in MD2 pineapple output under the 12th Malaysia Plan (12MP).

Evolution in Electrical and Electronic Engineering Vol. 6 No. 1 (2025) p. 34-42

Determining the maturity of MD2 pineapple before harvest requires a high level of expertise, and human grading can be time-consuming. It may lead to inaccuracies, especially for new workers. According to the Lembaga Perindustrian Nenas Malaysia (LPNM), the maturity indices consist of 7 different index to categorize the pineapple. The first 3 initial index is categorized as under-ripe meanwhile Index 4 and 5 is partially ripe and Index 6 and 7 is categorized as over-ripe. Each index has different purposes which is mainly to determine the fruits is preserved to export duties and retail market for household consumption [1].

Table 1: Classification on maturity indices of MD2 pineapples

Stage of Maturity	Percentage of Yellow (%)
Index 1	0-5
Index 2	5-10
Index 3	10-20
Index 4	20-30
Index 5	30-50
Index 6	50-75
Index 7	75-100

While human assessment has its flaws in gauging the ideal condition for ripening MD2 pineapples, technology may help. To address this challenge head-on, researchers proposed harnessing smartphones to precisely detect and assess pineapple maturity. More accurate evaluations could streamline the harvesting process, a key step for quality fruit production. Under this proposed system, an Android device could autonomously determine pineapple maturity in real-time, expediting the process significantly. Accordingly, the aim of this undertaking was developing a YOLOv8 model with the ability to reliably classify maturation levels in MD2 pineapples. This model found its way into an Android application intended to facilitate convenient identification of pineapple development status. Adjusting the system could help address any limitations and realize the full benefits of automated maturity evaluation for efficient, high-quality pineapple harvesting operations.

2. Datasets and System Overview

MD2 pineapples datasets was obtained from Malaysian Agricultural Research and Development Institute (MARDI). The dataset was separated randomly by using Roboflow into three partitions which is the training, validation and testing datasets. The ratio between each partition is set to 7:2:1 to overall images that are used by using the datasets given from the MARDI which is shown in **Table 2.** This dataset division process is executed using Google Collab by using the external graphic cards (GPU) by implementing three different YOLO versions to compare which is the highest accuracy parameters from the training and validation process.

Table 2: Total number of datasets for each index of MD2 pineapple

Stage of Maturity	Total image
Index 1	251
Index 2	229
Index 3	254
Index 4	133
Index 5	192
Index 6	205
Index 7	209
Total	1486

Table 2 shows a proportionate balance among these indices is essential for picture object recognition. By maintaining this balance, bias towards any stage of maturity is avoided and the detection model is trained uniformly across all stages. The model may perform well on stages with more photos but poorly on stages with fewer images if the distribution is uneven.

2.1 Software Development

In this project there are two main parts to execute the project. The first part is to develop or create one training parameter that uses YOLO model by using the transfer learning with the help of Roboflow for the image annotation and Google Collab for training model parameter. Second main part is to create the mobile application using Android Studio to design the interface of the application and adds on the pre-trained model by converting the YOLO model into TensorFlow Lite. Figure 1 shows the flowchart of the software development process. **Fig 1** shows the flowchart of the software development process.

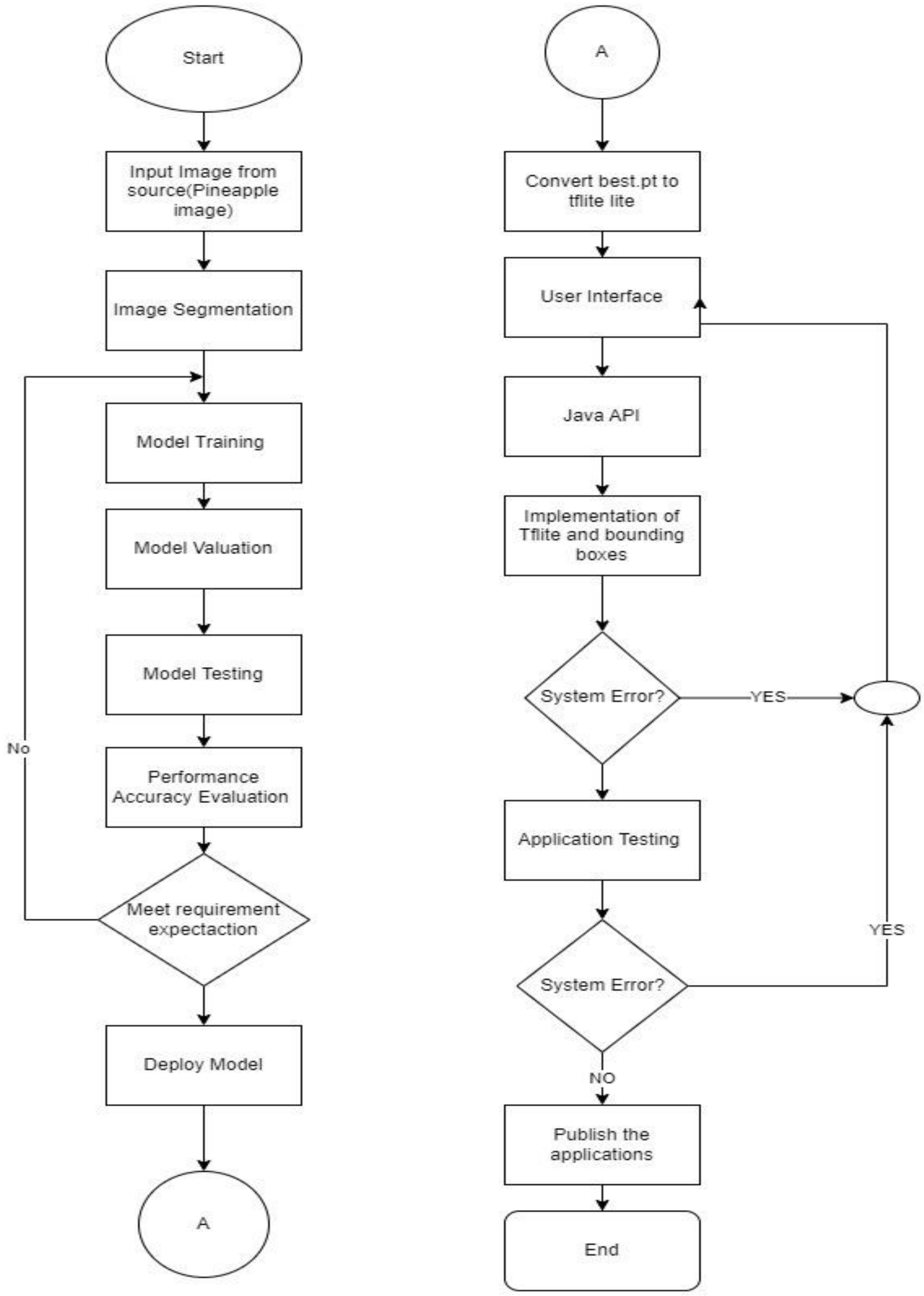

Fig. 1 *Flowchart of development applications*

2.1.1 Image Annotation

By using the features in the Roboflow Tools, Smart Polygon Is used to annotate the region at the MD2 Pineapple more accurately rather than depends on Bounding Boxes which increase the value of confidence in the annotation process because by using Smart Polygon increasing the amount dotted points in the annotation according to the desired point compared to Bounding Box which is using 4 dotted points and it causes lack of confidence in the annotation process. The process is done based on **Fig.2.**

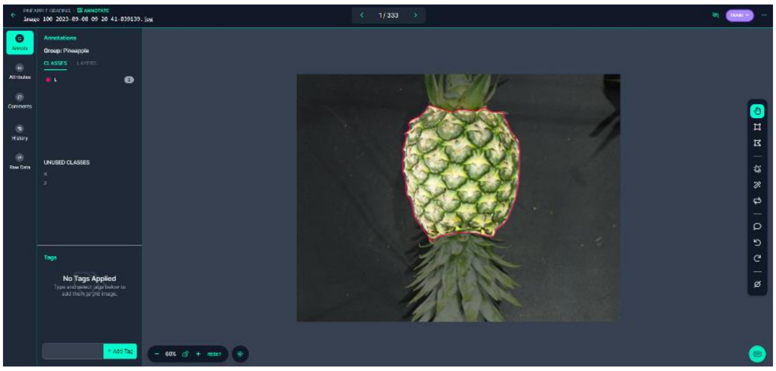

Fig. 2 *Flowchart of annotation in Roboflow*

2.1.2 Image Pre-processing

Before training the model, the images undergo preprocessing to ensure optimal training, validation, and testing outcomes while preventing model overfitting. **Fig. 3** depicts the preprocessing steps applied to the dataset images. Initially, the images are automatically oriented to the correct alignment. Subsequently, they are resized to a standardized dimension of 640x640 pixels. This preprocessing standardizes the image dimensions and orientations, preparing the dataset for effective model training and evaluation.

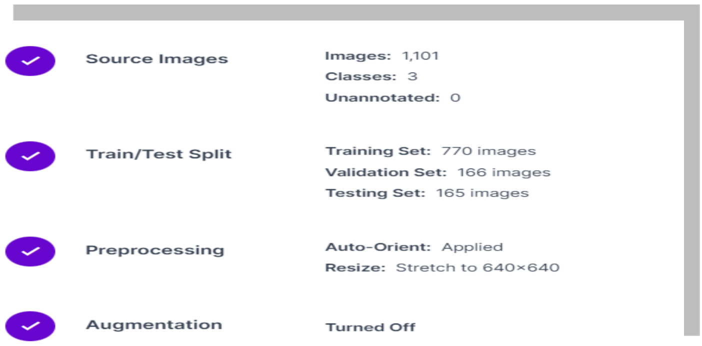

Fig. 3 *Pre-processing of images*

2.1.3 Training Datasets

Training data is the biggest subset of the original dataset, which is used to train or fit the machine learning model. In Roboflow, once completing the annotation process, it creates as dataset by dividing the annotated images into train, validation, and test sets. So, from the Fig. 4 the training model is set to ratio 70:20:10.

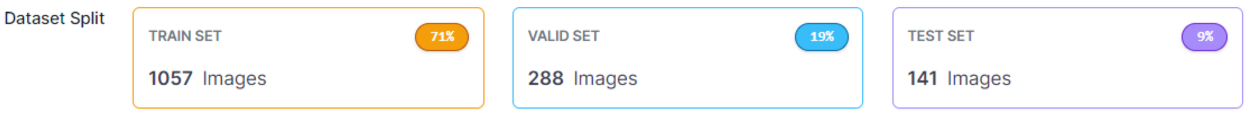

Fig. 4 *Train, test split images for training*

2.1.4 Performance Metrics of YOLO Models

For the quantitative evaluation, three performance metrics are selected which are precision (P), recall (R) and mean average precision (mAP) for each trained YOLO model as calculated in Eqs. 1,2 and 3. [2]

$$\text{mAP} = \frac{1}{n}\sum_{k=1}^{k=n} AP_k \quad (1)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (2)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (3)$$

where TP is the number of true positives, FP is the number of false positives, TN is the number of true negatives, FN is the number of false negatives, n is the number of thresholds and AP_k is the average precision of class k.

2.1.5 Google Collab for Training Model for YOLOv8

The Google Collab Platform is used for deep learning which to create one model that can be used to identify the indices of the maturity of the pineapple by using several processes from importing the document, model training, model detecting, and lastly model evaluation shown in **Fig 5**. The platform is free to use amongst all users, but it comes with limited time usage. In this situation, the training model needs to use the external GPU that is provided by Google to speed up the process of training and model evaluation. The coding language implemented in this model training parameter is by using the Python language.

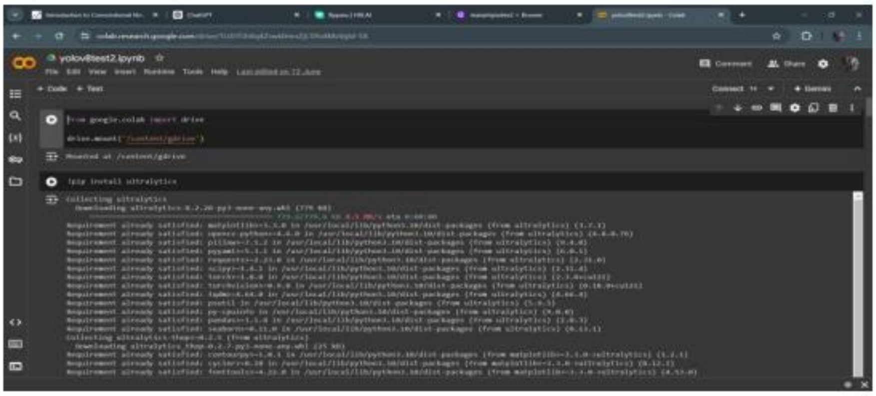

Fig. 5 YOLOv8 Training Process

2.1.6 Conversion Parameters into TensorFlow Lite Model

The TensorFlow Lite converter efficiently optimized the voluminous TensorFlow model for deployment on mobile devices and construction of android programs. The transformed TensorFlow Lite design, or streamlined TFlite formation, can presently be drawn down from the Google Colab environment. The TensorFlow Lite converter is an expedient tool permitting the transformation of the TensorFlow structure to a leaner, more rapid rendition suitable for simple distribution across mobile apparatus.

2.1.7 Android Application Development

The design editor within Android Studio affords developers a visual means of structuring and modifying their application's user interface through intuitive layout tools for conveniently placing buttons, images and other elements to optimize user interaction. Chief among the main screen is the title branded prominently above a vibrant logo with a natural fruit arrangement below classified for the user's review. Options are provided for users to personally capture and categorize snapshots while adjusting the backdrop hue to suit their personal preference of light or dark modes.

As depicted in **Fig. 6**, the application's schematic grants insight into its conceived design. Having trained a machine learning model using Google Collab's Jupyter notebooks, the converted TensorFlow Lite file is imported into Android Studio for evaluation through its built-in Java runtime environment. This seamlessly integrates with the Android Neural Networks API to expediently perform tasks such as image identification, prediction and selection by leveraging optimized computer kernels for heightened performance relative to a standalone interpreter.

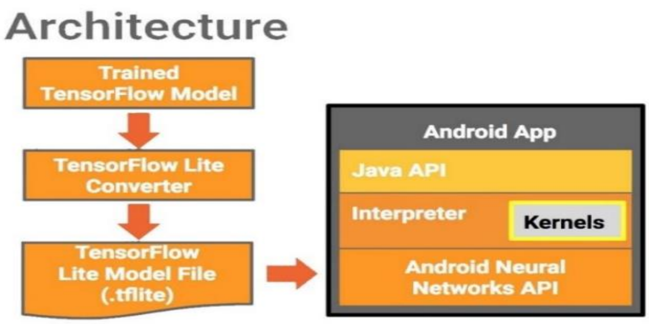

Fig. 6 *Performance graphs of YOLOv8*

3. Results and Discussion

3.1 Google Collab Datasets Result

Based on the **Fig.7** , the value of loss between the validation and losses tends to be downwards from the initial epoch which is the value 0 to to the highest epoch which is 300. The graph indicates the confidence level in classification to determine the grade of the MD2 Pineapples are high due to the low value of losses. The metrics and mAP(mean average precision) also indicates the model that are trained can be totally depends and can be deploy to desired output by applying to manufacturing or agriculture industries. The graphs indicates this model is quite accuate in determining the maturity of of MD2 pineapple bacause the losses value between training and validation is slightly similar to each other which both of the paramaters losses reached 0.2 losses but the validation process tends to scatters from epoch to another compare to the training data which concludes this model is slightly overfitting.

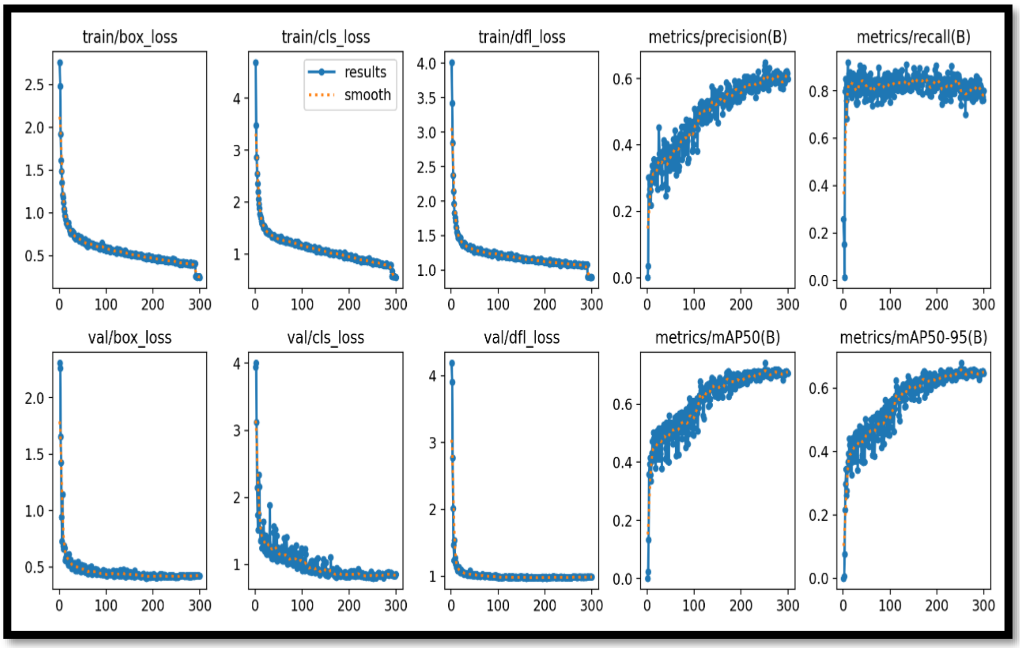

Fig. 7 *Performance graphs of YOLOv8*

Based on the Fig.8 the result of the model classification can be misclassified in certain. The order of number from highest to the lowest of confidence in classification are starting from Index 1 until Index 7refers by (Navy Blue Boxes). The "background" class in a confusion matrix for object detection acts as a sentinel. It helps identify both missed objects (low value on background diagonal means objects missed) and false alarms (high values in other background cells means the model is incorrectly seeing objects where there are none). This way, it ensures the model's accuracy by revealing where it's struggling to differentiate between actual objects and empty space

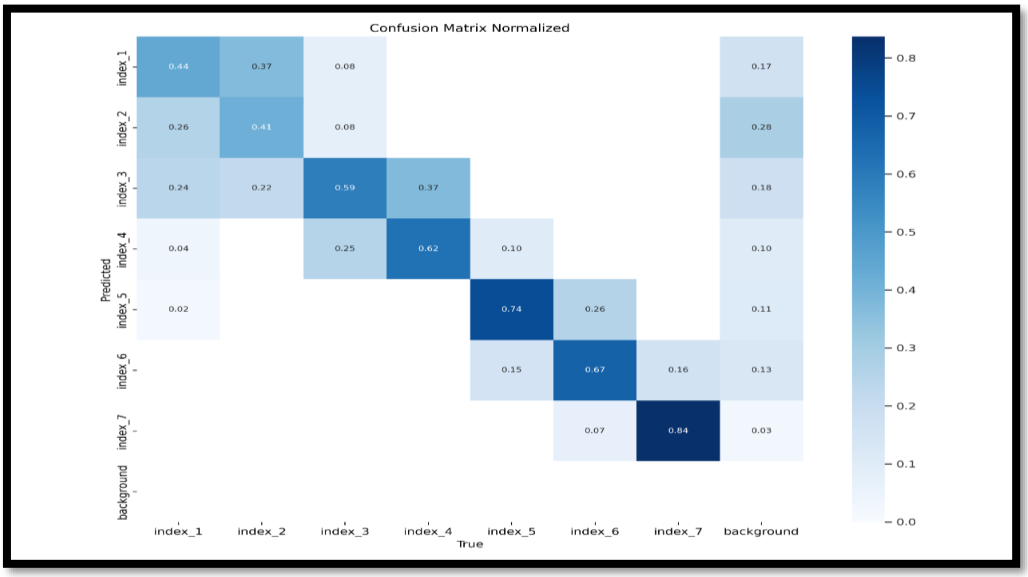

Fig. 8 *Confusion Matrix of MD2 classification model*

3.2 YOLO Models Result

Table 3 highlights performance of the MD2 pineapple detection models across different versions of YOLO and in mAP50 and precision, showcasing its outstanding ability in identifying medium-grade MD2 pineapples

Table 3 : Result for three different versions of YOLO

Version of YOLO	Type of Index	Precision	Recall	Mean Average Precision (mAP50)
	All	0.649	0.788	0.741
	Index 1	0.542	0.62	0.546
	Index 2	0.527	0.509	0.572
	Index 3	0.584	0.743	0.644
YOLOv8	Index 4	0.422	0.875	0.65
	Index 5	0.8	0.974	0.9496
	Index 6	0.73	0.907	0.875
	Index 7	0.936	0.884	0.953

The novel YOLOv8 convolutional model proved remarkably adept at classifying pineapple maturation levels, attaining overall metrics of 0.649 precision, 0.788 recall, and a mean average precision of 0.741 across testing images. Though initially achieving somewhat lower exactness for younger fruit, precision dramatically increased or more developed pineapples, most notably attaining pinpoint accuracy of 0.936 precision and a mAP of 0.953 for fully ripe specimens graded at Index 7. This customized deep learning system was then incorporated into a intuitive mobile application, creating a user-friendly tool for on-site inspection capable of advancing precision in pineapple ripeness evaluation for both consumer and agricultural use through streamlining the process of pineapple maturity assessment

3.3 Image Testing using Google Collab

Fig. 9 shows the confidence score of example random test images . During random test images ,we cant detect the if is the sample image is correcly is classified or not because it is random and during pre-test the image file is encrypted. To solve the issue , the model is tested using each categories of grade one by one to determine the accuracy to be implement after process which is the deploy process. These result obtained from the training model from Google Colab by using the YOLOv8 parameters.

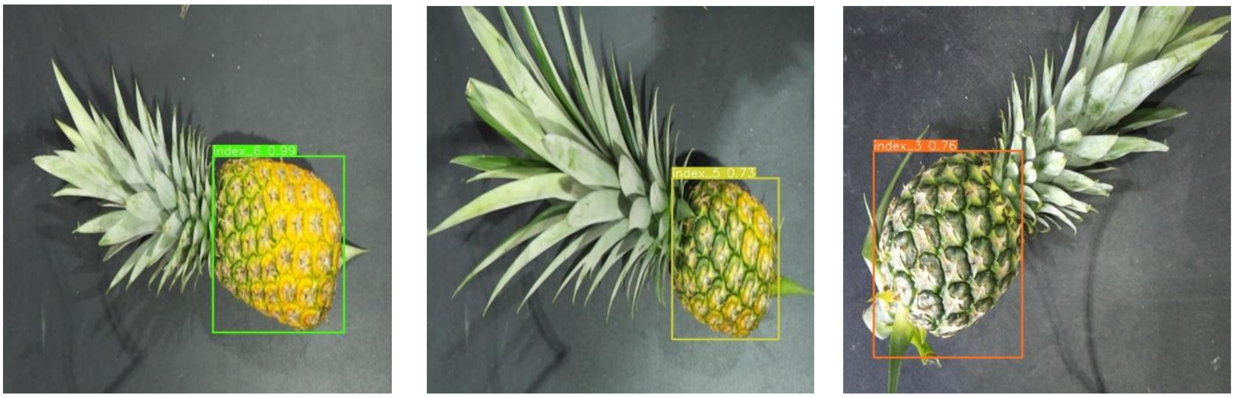

Fig. 9 *Output Accuracy from YOLOv8 model*

3.4 Mobile application result

Fig. 9. presents the results of evaluating the maturity of pineapples using the Samsung J7 Prime (LM-G610F) smartphone. Images were pre-processed by rescaling to a resolution of 640x640 pixels before analysis. The proposed model classified images into categories of overripe, ripe, underripe, and unripe pineapples, demonstrating the detection system on the LM-G610F. The application successfully assessed pineapple maturity from Index 1 to Index 7, with each detection taking approximately 5-7 seconds. Additionally, the Android application size was approximately 58 MB. Seven totals of fruits are used to sample the mobile applications to determine the effectiveness of the transfer learning.

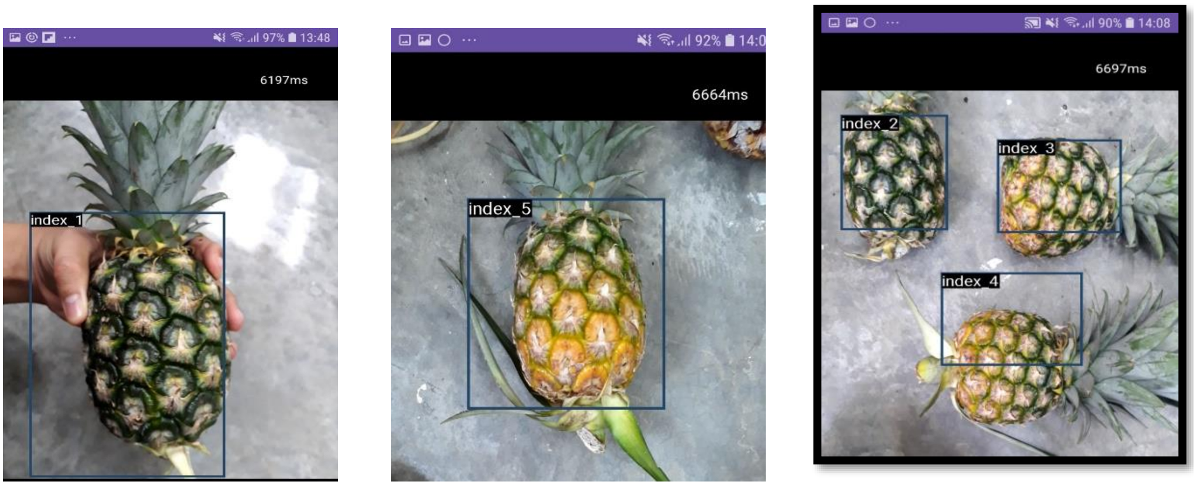

Fig. 10 *Mobile application result*

4. Conclusion

The project drew knowledge and findings of prior studies and literature reviews to gain exposure in developing the project by taking an example of past research which using both deep learning methods and by using hardware eccentric design that use various sensor to determine or differentiate fruits. The training model of YOLOv8 excels in detecting Index 5 until 7 but for Index 1 until 4 is moderate in detecting the index due to the number of datasets affects the quality of detecting the index. The solution for the problem is to add augmentation steps and retrain the model to obtain a higher output in detecting the lower index.

Overall, all models have their pros and cons when it comes to detecting object detection. These cutting-edge technologies have shown to be successful in accurately and swiftly recognising and classifying fruits based on their quality. The real-time object identification capabilities of YOLOv5 and YOLOv8 provide a practical means of automating the fruit grading process, reducing the need for manual labour, and increasing overall productivity in the import-export industry, small retail fruit company, and agricultural sector.

Acknowledgement

The authors would like to sincere gratitude to Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia and all those who have supported and contributed to the successful completion of this research.

Conflict of Interest

Authors declare that there is no conflict of interests regarding the publication of the paper.

References

[1] Buku Pengurusan Tanaman Nanas 2020. (n.d.).
[2] A.A. Azman and F.S. Ismail. (2017). "Convolutional Neural Network for Optimal Pineapple Harvesting.", ELEKTRIKA- Journal of Electrical Engineering, 16(2), 1–4. Available: https://doi.org/10.11113/elektrika.v16n2.54