Journal of Engineering Science and Technology
24th EURECA 2025 Special Issue April (2026) 96 - 109

VISION-BASED PAPAYA YIELD MONITORING UNDER OCCLUSION CONSTRAINTS

JIA SHENG SIEW, WEI JEN CHEW*

School of Engineering, Taylor's University, Taylor's Lakeside Campus,
No. 1 Jalan Taylor's, 47500, Subang Jaya, Selangor DE, Malaysia
*Corresponding Author: WeiJen.Chew@taylors.edu.my

Abstract

Accurate fruit counting bears great importance in modern precision agriculture due to its uses in yield estimation, crop management, and resource optimization. However, the detection and counting of papaya fruits under natural conditions face challenges such as occlusion, varying light conditions, and overlapping fruit bunches, which may result in missed detections or double counting. Therefore, this paper proposes an automated papaya counting system by integrating image processing techniques with a deep learning-based object detection model using the MATLAB software. The You Only Look Once version 4 (YOLOv4) algorithm is utilised as the base for object detection in the proposed system. It was trained using 60 images of papaya bunches and tested with 40 different images of papaya bunches at different occlusion levels. In the pre-processing stage, image enhancement techniques, such as bicubic interpolation and overlap tiling inference, were performed to improve the visibility of small and distant papayas for enhanced detection reliability and reduce missed detections. As for the post-processing stage, non-maximum suppression (NMS) is implemented to supress overlapping bounding boxes thus preventing double counting. The model developed in this study is evaluated on manually annotated ground truth data and presents both detection and counting accuracy. Experimental results showed that the proposed system obtained a mean average precision (mAP) of 76.7%, an average counting accuracy of 90.5%, a coefficient of determination (R^2) of 0.89, a Root Mean Square Error (RMSE) of 2.24, and a Mean Absolute Error (MAE) of 1.525. These results demonstrate that image processing integrated with deep learning effectively enhances the robustness in the detection of papaya fruits under occlusion, hence showing the potential of the system for real-time yield estimation in smart agricultural applications.

Keywords: Coefficient of determination (R^2), Image processing, Mean absolute error (MAE), mean average precision (mAP), Occlusion, Papaya yield estimation, Root mean square error (RMSE), You only look once version 4 (YOLOv4),

1. Introduction

Multi-object tracking is a fundamental task in computer vision which enables systems to detect and continuously follow multiple objects across consecutive video frames [1]. This system can be used to find or monitor objects and will be useful to be incorporated into any surveillance robot. Within the realm of digital agriculture, there has been a significant surge for an automated fruit counting system as it provides vital information for yield predictions. The papaya fruit has significantly contributed to the Malaysian economy as it represents one of Malaysia's top exports [2]. Although the papaya is a popular fruit, enjoyed for its taste and nutritional benefits, its short shelf-life highlights the need for cultivation practices to optimize its yield and quality.

As a result, an accurate count of papayas in a bunch will not only provide convenience to local farmers for yield monitoring but can also improve decision making for crop management. Traditionally, crop monitoring requires experienced farmers to manually count the number of crops per unit area, which is not only a very time-consuming process but also introduces the risk of human error when counting [3]. Thus, accurate crop count, obtained from yield predictions, will also directly contribute to rational use of resources such as water and fertilizer. However, it is important to note that precise counting of papayas is slightly complicated because they grow in overlapping clusters, which introduces substantial occlusion challenges.

Many computer vision algorithms are hindered by occlusion, which is a condition where the object being tracked is either partially or fully obscured by other objects or itself, hence causing tracking algorithms to lose or misidentify its target in real-world environments. Occlusion occurs under two categories which are self-occlusion and inter-object occlusion.

In this project's context, self-occlusion is when the body of one full papaya is obscured from a particular viewpoint whereas inter-object occlusion occurs when a papaya is obscured by another papaya or leaf [4]. Under the self-occlusion category, key visual features such as edges may become distorted, which disrupts appearance-based tracking models and results in identity switches [5].

On the other hand, inter-object occlusion is particularly problematic in crowded scenes, where frequent overlapping of individuals often leads to missed detection or incorrect associations [6]. Hence, these occlusion challenges highlight the need for occlusion-aware tracking models to maintain target identities in cluttered, dynamic environments.

Image processing, which constitutes an essential part of both human perception and digital interpretation, is the operation of deriving essential features from an image to improve its visual quality [7]. Image segmentation, feature extraction and background subtraction are some image processing tools which are fundamental to enhancing a system's ability to deduce object presence even when direct visibility is lost [8]. The challenge of complex visual data arises due to its image acquisition conditions such as lighting, viewpoint and bunch occlusion.

Traditional methods based on handcrafted features are often insufficient for capturing semantics and high-level understanding are often needed for tasks like scene interpretation and object detection [9]. Consequently, developing systems

capable of efficiently analysing, interpreting and understanding this complexity remains a major open challenge.

Over the past two decades, researchers have been leveraging deep learning models, especially on convolutional neural networks (CNN), to conduct segmentation, classification and object detection [10]. These models significantly reduce the need for manual feature design, hence allowing systems to scale and adapt to complex scenarios more effectively. For instance, region-based convolutional neural networks (R-CNN) was proposed in this field of research to tackle said challenges. R-CNN's two-stage architecture combines region proposals with CNN to identify and classify objects within an image [11].

On the other hand, the You Only Look Once (YOLO) series is a single-stage framework which processes entire images directly with a single neural network thus proposing itself as an alternative to R-CNN due to its faster detection speeds [12]. Furthermore, YOLO learns generalizable representations of objects so when trained on natural images, it is less likely to break down when applied to new domains, which outperforms R-CNN by a wide margin.

The overall objective of this project is to develop an automated papaya counting system which detects the quantity of papaya in bunches at different levels of occlusion under varying field conditions by integrating image processing techniques and the YOLOv4 detection model using the MATLAB software, in order to achieve an accurate, real-time computer vision system that is able to precisely count the number of papayas per bunch to optimize yield proficiency. The outcome of this study aims to support local farmers with real-time decision-making tools which ultimately enhances crop management, reduces labour dependency of manual counting as well as optimize resource utilization in papaya cultivation.

There are multiple research papers which discusses the fundamentals and relationships between image processing techniques and deep learning models. Due to the plethora of detection frameworks, there are also several key studies which covers modern approaches to object detection, with emphasis on occlusion-handling and domain-specific applications to agriculture. Although existing studies contributed significantly to the fields of image processing and object detection architectures, there are still several limitations which this project intends to address and to further enhance its existing models.

A review on image processing by Jingar et al. [7] emphasizes on the importance of image manipulation in improving visual clarity through a qualitative study with clear before-and-after visuals using MATLAB's Image Processing Toolbox. The author sections the entire digital image processing pipeline into pre-processing and post-processing stages. Pre-processing stages involve image resizing, resolution selection and format conversion. On the other hand, post-processing stages, such as image enhancements and segmentation, serves to extract meaningful information from cluttered environments.

However, the paper did not provide any empirical results to show that image processing directly improved object detection accuracy thus this paper serves as a theoretical discussion. Therefore, this significant research gap underscores the need to merge classical image processing with deep learning models to enable robust image interpretation in complex scenarios. This project intends to fill this research gap by employing classical image processing as the pre-processing layer before

feeding the optimized image into a deep learning model to potentially improve detection accuracy, particularly in occluded scenes.

On the other hand, research conducted by Zhou et al. [13] developed a fruit detection system which can accurately detect ripe/unripe strawberries under occlusion in videos. This study utilizes multiple mapping algorithm, which is a tracking-by-detection algorithm, to trace multiple fruits from their first appearance to their final appearance in real-time footage. Just like papayas, strawberries can overlap one another, and varying field conditions such as uneven lighting and orientation can also alter its appearance. This research utilised 440 open-source annotated images for training and 40 videos for testing.

The author employed You Only Look Once version 5 (YOLOv5) as its base detection architecture and further enhanced it with Convolutional Block Attention Module (CBAM) to improve its strawberry feature extractions. With its detection threshold fixed at 50%, their base YOLOv5 model attained a mean Average Precision (mAP) of 83.60% and when enhanced with CBAM, their mAP increased to 89.90%. As for counting accuracy without multiple mapping algorithm, the system achieved an average coefficient of determination (R^2) of 0.66 and an average Root Mean Square Error (RMSE) of 512.7 but with the multiple mapping algorithm, the system achieved an average R^2 of 0.94 and an average RMSE of 29.3.

Another study conducted by Íñiguez et al. [14] evaluated the YOLOv4 architecture for detection of grape bunches under various levels of leaf occlusion in commercial vineyards. As for dataset delegation, this study utilises 1014 images for training and 690 images for testing. Its detection architecture achieved an average mAP of 85.11% at 50% detection threshold across all levels of leaf occlusion, which indicates its robust ability in detecting grape bunches, not individual grapes. As for counting accuracy, the system achieved an average R^2 of 0.78 and an average RMSE of 1.16.

Although existing studies have achieved commendable detection and counting accuracies, their models are primarily optimized for specific environments, under controlled lighting and uniformed backgrounds. These conditions do not fully reflect diverse, cluttered visual conditions encountered in real-world agricultural fields. Furthermore, these studies do not investigate how image processing techniques could compliment deep learning models to improve detection accuracy in occluded scenarios. Therefore, this project intends to fill these research gaps by developing an occlusion-aware automated papaya counting system by integrating image processing techniques with the YOLOv4 detection model, in real-time agricultural applications.

2. Methodology

Figure 1 illustrates the methodology flowchart which summarizes the general structure of the project. At the beginning of the project, a papaya image dataset is compiled from an online database. These papaya images will be manually annotated with bounding boxes to serve as the ground truth for both training and testing datasets. The papaya images are then randomly split into training and testing datasets at a 60:40 ratio. Once the papaya database has been established, the YOLOv4 model detection model will be fine-tuned to optimize its detection performance during its training. Next, image processing techniques will be integrated into the YOLOv4 detection model to improve both detection and

counting accuracy. The individual predicted papaya results obtained from the system will then be evaluated based on a set of evaluation metrics such as Average Precision (AP), by measuring the area under the precision-recall curve, and its counting accuracy, by dividing the predicted count with its true count. The individual results will be tabulated and compiled to obtain its overall detection and counting accuracy by using a set of formulas to calculate its mean Average Precision (mAP), coefficient of determination (R^2), Root Mean Square Error (RMSE) and Mean Absolute Error (MAE).

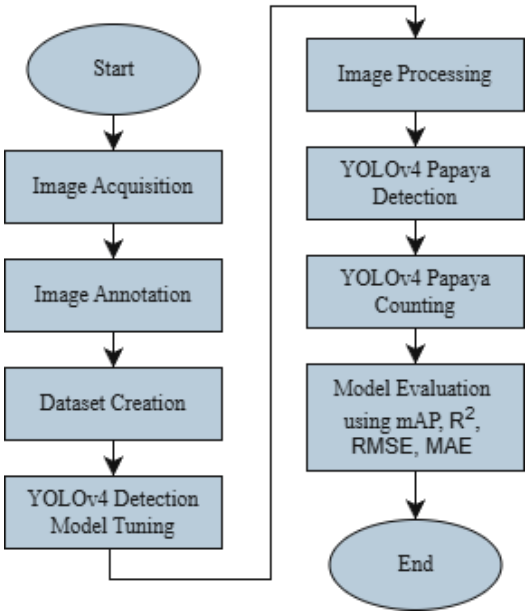


Fig. 1. Methodology structure flowchart.

2.1. Image acquisition and image annotation

The papaya bunch images for this project were taken from an online database, Roboflow, which is free to view and open source. The digital images were saved in JPG format, and the dimensions of these papaya images were fixed at 640 x 640 pixels. Each papaya images were captured without the application of artificial lighting or filters hence reflecting true, uncontrolled and natural conditions. The total number of papaya images taken from this dataset was 100.

After image acquisition, the number of papayas from each papaya trees were determined through manual labelling to obtain the true count of papayas in each bunch. The images were manually annotated, with a focus of identifying and labelling visible bunches using bounding boxes, through the imageLabeler built-in function in the MATLAB software. These labelled visible bunches will be used as reference, serving as the ground truth to train the YOLOv4 detection model.

2.2. Dataset creation

The 100 acquired papaya images will need to be divided into training and testing datasets for the subsequent YOLOv4 detection model. 60 papaya images (60%)

will be allocated for training, whereas the remaining 40 papaya images (40%) will be allocated for testing. This split composition provides a good balance for model development as recommended from a guideline written by Archaya [15] as well as the MATLAB Help Center. Training datasets will be manually annotated to provide necessary guidance for the model to recognize individual papayas. The testing dataset will possess a different set of images, as compared to the training dataset, to produce unbiased results. If the datasets were not separated, the model will produce impractical results by replicating its training data into the testing results which hinders the system from adapting to real-world conditions.

2.3. YOLOv4 detection model tuning

The base detection model selected for this task would be none other than the YOLOv4 framework due to its lightweight structure and also because the MATLAB software currently does not support YOLOv5 functions and above. As for the YOLOv4 training control parameters, the input image sizing was set to 640 x 640 pixels with a mini batch size of 2. The number of epochs was fine-tuned and set to 150 epochs as demonstrated in Fig. 2. Its initial learning rate starts at 0.001, applied with a squared gradient decay factor of 0.999, which allows the model to continue learning effectively throughout the training process by preventing the learning rate from rapidly decreasing. Last but not least, "Adam" was selected as the optimization algorithm for the YOLOv4 model by default. These training hyperparameters are summarized in Table 1.

Table 1. YOLOv4 training hyperparameters.

Item	Value
Input Size	640 x 640
Mini Batch Size	2
Epochs	150
Initial Learning Rate	0.001
Squared Gradient Decay Factor	0.999
Optimizer	"Adam"

Regarding the value selection of the training hyperparameters, the initial learning rate is set to 0.001 as it is regarded as a stable and effective default value for the Adam optimizer. Initial learning rate controls the step size taken by the optimizer when updating network weights. If the initial learning rate is set too high, it may cause unstable training as the model may overshoot the optimal solution. On the other hand, if the initial learning rate is set too low then the training process will take a longer period to complete.

Hence, selecting 0.001 as the initial learning rate achieves an ideal balance between stable training and moderate training durations which allows the model to efficiently learn discriminative features for papaya detection. Moving on, squared gradient decay factor controls how soon the Adam optimizer forgets past squared gradient when updating the learning rate. A high squared gradient decay factor like 0.999 enables the Adam optimizer to maintain a smooth and stable learning curve across long training durations. Hence a squared gradient factor of 0.999 not only helps the model to keep long-term gradient memory but also helps avoid sudden transitions in learning rate behaviour.

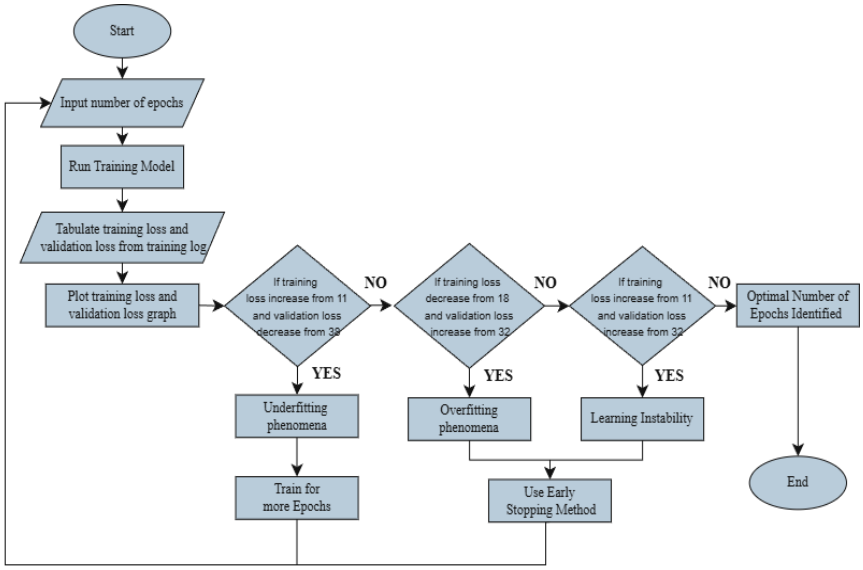


Fig. 2. Optimal epoch tuning flowchart.

Figure 2 demonstrates a methodology flowchart to fine-tune the optimal number of epochs when training the detection model. An epoch generally refers to one complete cycle through the entire training dataset, thus fine-tuning the number of epochs is necessary to prevent overfitting and underfitting phenomena [16]. Overfitting occurs when the number of epochs exceeds its necessary amount, which causes the training model to memorize the training data thus hindering its performance on unseen data.

On the other hand, underfitting occurs when the number of epochs is insufficient, which reduces the model's training time to learn complicated or underlying patterns in the dataset. Both overfitting and underfitting will significantly affect the training model's overall Average Precision (AP) if left unattended [16]. Hence, by determining the optimal number of epochs based on the trend from the training loss and validation graph, it will optimize the detection model's learning capabilities to accurately detect papaya bunches in the testing dataset. In the scenario where underfitting occurs, the number of epochs will be increased to cater for longer training duration. On the other hand, if overfitting or learning instability occurs, early stopping will halt the training when validation loss starts to increase. Data augmentation was employed at the last layer of the YOLOv4 detection model to enhance its capability of recognizing both visible papayas and partially obscured papayas in a real-world agricultural environment. As such, a variety of data augmentation techniques were applied as summarised in Table 2. Modifications were made to the colour saturation and brightness with a coefficient of 0.2, as well as the colour hue with a coefficient of 0.1, to simulate lighting variations and colour intensities. Furthermore, geometric variations were also implemented, where "Xreflection=true" randomly flips images horizontally to simulate different viewing angles, "Scale=[1,1.1]" randomly scales images to simulate small zoom variations and "BoundStyle=centerOutput" keeps the transformed images at the centre.

Table 2. Data Augmentation hyperparameters.

Item	Value
Colour Hue	0.1
Colour Saturation	0.2
Colour Brightness	0.2
XReflection	True
Scale	1:1.1
BoundStyle	centerOutput

2.4. Image processing

During the initial testing phase without image processing, the YOLOv4 papaya detection system frequently experienced missed detections of small papayas in the background and overlapping bounding boxes occasionally led to overcounting. Missed detections stemmed from smaller papayas which occupied only a few pixels in low resolution, making it difficult for the YOLOv4 model to extract visual features for detection. On the other hand, overcounting occurred when the detector generated multiple detections for the same papaya in dense and occluded regions. To overcome these limitations, image processing techniques were implemented and segregated into pre-processing and post-processing stages.

As demonstrated from Fig. 3, the size of each tile is set at 640 x 640 pixels (line 12), which matches the input size of YOLOv4 detector. The tiles are then generated with 40% overlap (line 13) in both horizontal and vertical directions so that papayas at the edges are visible at an adjacent tile. This overlapping area ensures that no papayas are missed when being sandwiched between tiles. The input test image is enlarged by 1.5x with bicubic interpolation at line 15. The upscaled image is an essential method of image preprocessing, as it magnifies the fine spatial features of papayas in the image while preserving the quality of the texture. This modification helps expand the smaller, distant papayas in the background to larger pixels, making them easier for YOLOv4 model to detect. After upscaling, the image is divided into overlapping tiles using the proposed detectWithTiling function, as shown at line 16. Finally, individual tiles are passed onto the detector, which will then produce predicted bounding boxes with confidence scores. The detections from each individual tiles are then stored and merged.

```
12      tileSize = [640 640];
13      overlap = [0.4 0.4]; % 40% overlap between tiles
14
15      I_up = imresize(I, 1.5); % 2x upsample before tiling
16      [bboxes, scores, labels] = detectWithTiling(detector, I_up, tileSize, overlap, threshold);
17      bboxes = bboxes / 1.5; % scale back to original coordinates
```

Fig. 3. Image Pre-processing MATLAB code.

Since the root cause for overcounting is derived from overlapping bounding boxes, non-maximum suppression (NMS) is implemented. The Intersection over Union (IoU) of all predicted bounding boxes are analysed and the bounding boxes with the highest confidence score are kept while the weaker duplicates are suppressed for each papaya instance. The overlap threshold is set to 0.2, meaning that two detections with more than 20% spatial overlap are considered redundant. Hence, this post-processing image processing technique prevents the system from counting the same papaya more than once, which is crucial for accurate counting.

2.5. Statistical analysis and model evaluation

Since the system is built to detect and count papayas under varying occlusion conditions, several evaluation metrics will be used to assess the accuracy and efficiency of the papaya counting system. Beginning with evaluation of detection performance, its main evaluation metrics would be Average Precision (AP), which summarizes the performance across all confidence thresholds by calculating the area under the precision-recall curve. Hence, given that P(r) is the precision at a given recall, the AP is calculated using Eq. (1).

$$AP = \int_0^1 P(r)dr \tag{1}$$

Next the counter will be evaluated through the following metrics. Primarily, the coefficient of determination (R^2) will measure the correlation between predicted counts and the actual count of papayas thus emphasizing on its overall counting accuracy, which is calculated in Eq. (2). Subsequently, the Root Mean Square Error (RMSE) reflects on the prediction error margin, and the Mean Absolute Error (MAE) measures the average absolute difference between predicted count and actual count of papayas, as calculated in Eq. (3) and Eq. (4).

$$R^2 = 1 - \frac{\sum_1^n (True_i - Predicted_i)^2}{\sum_1^n (True_i - Mean_{true})^2} \tag{2}$$

$$RMSE = \sqrt{\frac{\sum_1^n (Predicted_i - True_i)^2}{n}} \tag{3}$$

$$MAE = \frac{1}{n}\sum_1^n |True_i - Predicted_i| \tag{4}$$

where $True_i$ represents the i-th true count, $Predicted_i$ represents the i-th predicted count, $Mean_{true}$ represents the mean of all true counts and n represents the number of samples.

3. Results and Discussion

After training the YOLOv4 detection model with the optimized parameters outlined in Table 1 with 60 papaya images and subsequently integrating image processing techniques as defined in Section 2.4, the system was tested on a dataset of 40 papaya images under real-world, occlusion conditions. The primary objective is to evaluate the proposed system's detection and counting accuracy performance in comparison to the baseline YOLOv4 model without image processing enhancements.

3.1. YOLOv4 detection performance

While training the YOLOv4 detection model, the mean Average Precision (mAP) was measured on its training dataset to analyse its learning and detection consistency. As shown in Fig. 4, the YOLOv4 model achieved a mAP of 0.93 on its training dataset, meaning the model has successfully learned to identify and segment papaya images during training. Achieving a high mAP in the training dataset also validates that the model architecture. Hyperparameter tuning and data augmentation techniques were effective in teaching the network to focus on distinct attributes of papaya features such as its shape, texture and colour under varying occlusion conditions.

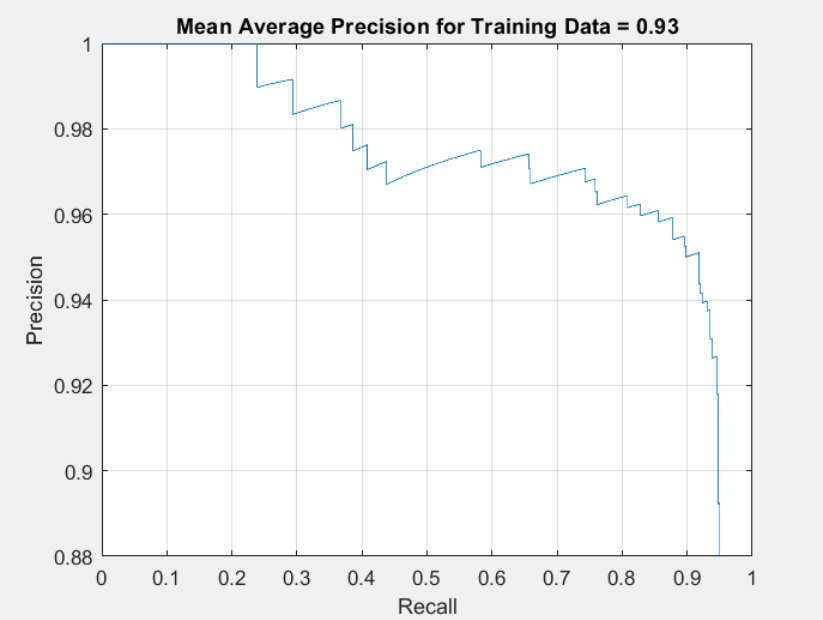


Fig. 4. Precision-recall graph on training dataset.

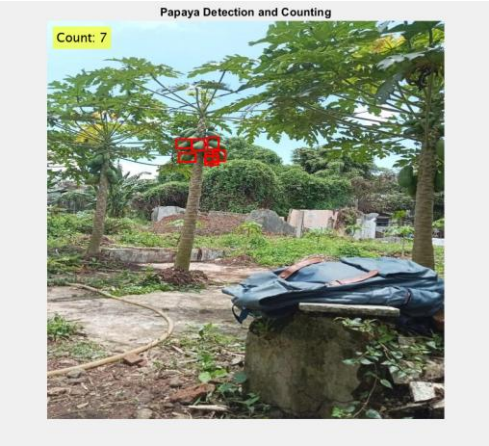
When individually tested on 40 unseen papaya images from the testing dataset, the model received a mAP of 0.77, showing a slight difference of 0.16 in its mAP as compared to its training dataset. This decrease in mAP can be attributed to the increased complexity of testing images that involves factors like lighting, background clutter and orientation of the papaya bunch, which were not adequately represented in the training images. Even so, maintaining an AP above 0.75 on the testing dataset shows that the YOLOv4 model was able to capture deep and relevant features pertaining to robust papaya detections. A similar study conducted by Íñiguez et al. [14] used YOLOv4 for grape detection and obtained mAP ranging from 0.74 to 0.92 across 4 modified models, hence supporting 0.77 mAP as a realistic high-performance range for complex agricultural environments.

3.2. YOLOv4 counting performance

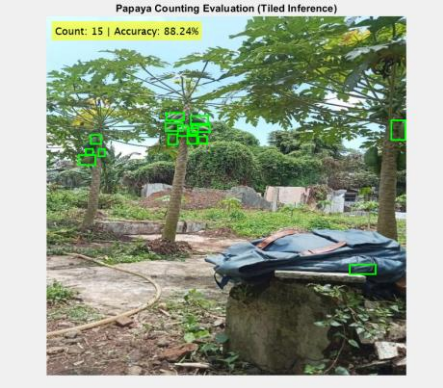
The counting performance of the YOLOv4 papaya detection system was quantitatively evaluated based on its ability to accurately predict the number of papayas per bunch under varying occlusion conditions. Prior to the integration of image processing techniques, the YOLOv4 detection model frequently resulted in missed detections and overcounting errors, especially in complex scenarios involving small, low-resolution papayas in the background and clustered papaya bunches.

As demonstrated in Fig. 5 (a), the system had very poor detection performance for small papayas in the background due to their limited pixel representation, which made its visual features indistinguishable to the YOLOv4 model. To tackle this issue, image pre-processing stage was implemented into the system, where input images were upscaled by 1.5x via bicubic interpolation before detecting. This upscale effectively magnified the fine spatial details of small papayas while preserving its texture, hence the model was able to extract more informative

features from distant papayas. As a result, Fig. 5(b) showcases the modified YOLOv4 version where the number of missed detections was highly reduced since the smaller papayas that were previously invisible at lower resolutions became detectable after upscaling.



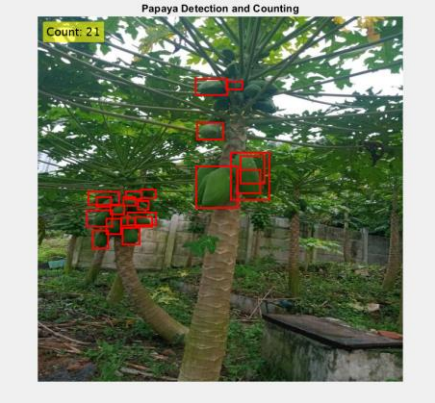
(a)



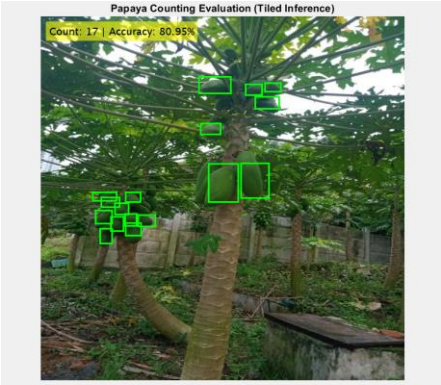
(b)

Fig. 5. (a) Baseline and (b) modified YOLOv4 detection model for missed detections on small papayas.

Furthermore, overcounting usually occurs in densely clustered regions where more than one overlapping bounding box was generated over one papaya instance, as demonstrated in Fig. 6(a). As a countermeasure to this issue, non-maximum suppression was incorporated into the image post-processing stage. It calculates the IoU of overlapping predicted bounding boxes and selects the box with the highest confidence scores. The overlap threshold is tuned to 0.2, indicating that two or more bounding boxes detected with an overlap of more than 20% will be considered as duplicates and hence suppressed. Results shown in Fig. 6(b) illustrate the modified YOLOv4 version with only one accurate bounding box per papaya, thereby overcoming the overcounting issue and producing a more accurate count.



(a)



(b)

Fig. 6. (a) Baseline and (b) modified YOLOv4 detection model for overcounting on clustered papayas.

Table 3. Counting accuracy results comparison.

Project Title	R^2	RMSE	MAE
Proposed Vision-Based Papaya Yield Monitoring under Occlusion Constraints	0.89	2.24	1.525
Occlusion-resilient Online Multiclass Strawberry Counting [13]	0.94	29.30	24.50
Deep learning modelling for non-invasive grape bunch detection under diverse occlusion conditions [14]	0.78	1.16	3.515

As illustrated in Table 3, this project demonstrates a strong balance of accuracy and consistency when compared to other occlusion-aware fruit counting models. The system attained an overall mean average counting accuracy of 90.5%, R^2 value of 0.89, RMSE value of 2.24, and MAE value of 1.525 across the testing dataset. Although the strawberry counting system [13] achieved a slightly higher R^2 value of 0.94, its high RMSE and MAE values indicate high deviations between its predicted and actual counts. In contrast, the grape bunch detection system [14] recorded the lowest RMSE value but its lower R^2 value of 0.78 reflects less consistent counting

accuracy. Hence by integrating both image pre-processing and post-processing techniques, these results prove that both undercounting and overcounting errors were minimized, leading to reliable and consistent counting performance across papaya bunches with diverse occlusion levels and field conditions.

4. Conclusions

In conclusion, this study accomplished the development of a vision-based papaya yield monitoring system under occlusion constraints by integrating image processing techniques with deep learning to perform accurate detection and counting of papayas in a natural setting. This hybrid model was able to overcome issues like missed detections of small papayas and overcounting caused by overlapping bounding boxes on one papaya instance. From the experimental results, it was concluded that the system showed an average counting accuracy of 90.5%, an R^2 value of 0.89, an RMSE of 2.24, a 1.525 MAE, and a mean Average Precision (mAP) of 76.7% on the testing data, while an 0.93 AP was documented on the training data, confirming the model consisted of strong generalization properties.

Nomenclatures

R^2	Coefficient of Determination

Abbreviations

CBAM	Convolutional Block Attention Module
CNN	Convolutional Neural Network
IoU	Intersection over Union
MAE	Mean Absolute Error
mAP	Mean Average Precision
NMS	Non-Maximum Suppression
RMSE	Root Mean Square Error
YOLO	You Only Look Once

References

1. Van Ma, L.; Nguyen, T.T.D.; Shim, C.; Kim, D.Y.; Ha, N.; and Jeon, M. (2024). Visual multi-object tracking with re-identification and occlusion handling using labeled random finite sets. *Pattern Recognition*, 156, 110785.
2. Sekeli, R.; Hamid, M.H.; Razak, R.A.; Wee, C.-Y.; and Ong-Abdullah, J. (2018). Malaysian *Carica papaya* L. var. Eksotika: Current research strategies fronting challenges. *Frontiers in Plant Science*, 9, 1380.
3. Jiang, Q.; Huang, Z.; Xu, G.; and Su, Y. (2023). MIoP-NMS: Perfecting crops target detection and counting in dense occlusion from high-resolution UAV imagery. *Smart Agricultural Technology*, 4, 100226.
4. Chandel, H.; and Vatta, S. (2015). Occlusion detection and handling: A review. *International Journal of Computer Applications*, 120(10), 33-38.
5. Luo, W.; Xing, J.; Milan, A.; Zhang, X.; Liu, W.; and Kim, T.-K. (2021). Multiple object tracking: A literature review. *Artificial Intelligence*, 293, 103448.
6. Xu, Y.; Ban, Y.; Delorme, G.; Gan, C.; Rus, D.; and Alameda-Pineda, X. (2022). TransCenter: Transformers with dense representations for multiple-

object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(6), 7820-7835.

7. Jingar, P.; Panchal, K.; and Oza, P. (2024). A review on image processing. *International Journal of Intelligent Systems and Applications in Engineering*, 12(14S), 515-520.
8. Rani, N. (2017). Image processing techniques: A review. *Journal on Today's Ideas - Tomorrow's Technologies*, 5(1), 40-49.
9. Dufaux, F. (2021). Grand challenges in image processing. *Frontiers in Signal Processing*. 1, 675547.
10. Li, F.; Li, X.; Liu, Q.; and Li, Z. (2022). Occlusion handling and multi-scale pedestrian detection based on deep learning: A review. *IEEE Access*, 10, 19937-19957.
11. Sumit; S.B.; Joshi, S.; and Rana, U. (2024). Comprehensive review of R-CNN and its variant architectures. *International Research Journal on Advanced Engineering Hub* (*IRJAEH*), 2(04), 959-966.
12. Bhattacharjee, S. (2025). A literature-based performance assessment of the YOLO (You Only Look Once) CNN approach for real-time object detection. *Computology: Journal of Applied Computer Science and Intelligent Technologies*, 4(2), 18-40.
13. Zhou, X.; Zhang, Y.; Jiang, X.; Riaz, K.; Rosenbaum, P.; Lefsrud, M.; and Sun, S. (2024). Advancing tracking-by-detection with MultiMap: Towards occlusion-resilient online multiclass strawberry counting. *Expert Systems with Applications*, 255, 124587.
14. Íñiguez, R.; Gutiérrez, S.; Poblete-Echeverría, C.; Hernández, I.; Barrio, I.; and Tardáguila, J. (2024). Deep learning modelling for non-invasive grape bunch detection under diverse occlusion conditions. *Computers and Electronics in Agriculture*, 226, 109421.
15. Acharya, A. (2023). Training, validation, test split for machine learning datasets. Retrieved November 5, 2025, from https://encord.com/blog/train-val-test-split/
16. Jiang, Y. (2023). How does epoch affect accuracy in deep learning model? Retrieved November 5, 2025, from https://www.geeksforgeeks.org/how-does-epoch-affect-accuracy-in-deep-learning-model/