Available online at website: https://jurnal.iaii.or.id/index.php/RESTI

JURNAL RESTI

(Rekayasa Sistem dan Teknologi Informasi)

Vol. 9 No. 3 (2025) 697 - 705 e-ISSN: 2580-0760

Enhancing Areca Nut Detection and Classification Using Faster R-CNN: Addressing Dataset Limitations with Haar-like Features, Integral Image, and Anchor Box Optimization

Yovi Pratama [1*,2], Errissya Rasywir[2], Suyanti[3], Agus Siswanto[3], Fachruddin[3]

[1*]Faculty of Computing, Universiti Teknologi Malaysia, Johor Bahru, Johor 81310, Malaysia

[2]Informatics Engineering Department, Computer Science Faculty, Universitas Dinamika Bangsa, Jambi 36138, Indonesia

[3]Information System Department, Computer Science Faculty, Dinamika Bangsa University, Jambi 36138, Indonesia

[1]yovi.pratama@gmail.com, [2]errissya.rasywir@gmail.com, [3]suyanti272@gmail.com, [4]agussiswanto@unama.ac.id, [5]fachruddin@unama.ac.id

Abstract

The classification and detection of areca nuts are essential for agriculture and food processing to ensure product quality and efficiency. The manual classification of areca nuts is time-consuming and prone to human error. For a more accurate and efficient automated approach, a deep learning-based framework was proposed to address these challenges. This study optimizes the Faster R-CNN by integrating Haar-like features and integral images to enhance object detection. However, dataset limitations, including low image quality, inconsistent lighting, cluttered backgrounds, and annotation inaccuracies, affect the model performance. In addition, the small dataset size and class imbalance hindered generalization. The Faster R-CNN model was trained with and without Haar-like Features and Integral Image enhancement. Performance was evaluated based on training loss, accuracy, precision, recall, F1-score, and mean average precision (mAP). The effects of the dataset limitations on detection performance were also analyzed. The optimized model achieved better stability, with a final training loss of 0.2201, compared to 0.1101 in the baseline model. Accuracy improved from 62.60% to 73.60%, precision from 0.6161 to 0.7261, recall from 0.3094 to 0.4194, F1-score from 0.2307 to 0.3407, and mAP from 0.1168 to 0.2268. Despite these improvements, dataset constraints remain a limiting factor. While the integration of Haar-like features and integral images into faster R-CNN contributes to detection accuracy, the study also reveals that high-resolution images, precise annotations, and dataset scale significantly amplify model performance.

Keywords: areca nut classification; deep learning; faster R-CNN; haar-like features; integral image; object detection

How to Cite: Y. Pratama, E. Rasywir, Suyanti, A. Siswanto, and Fachruddin, "Enhancing Areca Nut Detection and Classification Using Faster R-CNN: Addressing Dataset Limitations with Haar-like Features, Integral Image, and Anchor Box Optimization", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 9, no. 3, pp. 697 - 705, Jun. 2025.
*Permalink/*DOI: https://doi.org/10.29207/resti.v9i3.6496

Received: March 19, 2025
Accepted: June 17, 2025
Available Online: June 25, 2025

1. Introduction

The Faster R-CNN architecture is widely used for object detection tasks due to its high accuracy and robustness [1]-[3]. However, its performance may be affected when applied to complex agricultural datasets, particularly involving small objects with detailed textures, such as areca nuts [4]-[6]. To address these challenges, we propose enhancements to the original Faster R-CNN framework by integrating Haar-like features and integral image techniques, along with anchor box optimization [2], [7]-[9]. The proposed method addresses performance issues in object detection and classification caused by limited and low-quality datasets. Our experimental results show significant improvements in accuracy, precision, and mean average precision (mAP), indicating the effectiveness of the proposed modifications in supporting automated areca nut analysis [10]-[12].

This study focuses on enhancing the Faster R-CNN framework, a widely used object detection model, by integrating classical features to overcome limitations observed in small-object agricultural datasets [4], [7], [8]. One of the main challenges in this research is the suboptimal dataset quality, which includes low image resolution, inconsistent lighting, complex backgrounds, and imprecise bounding box annotations. These issues lead to low Intersection over Union (IoU) values, making it difficult for the model to recognize objects accurately [11], [13]-[16]. Furthermore, the dataset is limited and imbalanced, with a significantly higher number of healthy areca nut images compared to defective or diseased nuts. This imbalance results in prediction bias and reduced recall performance[8], [17]-[19] .

Several previous studies have successfully improved object detection accuracy using models such as YOLO and ResNet, yet these approaches still face challenges when handling objects with complex texture patterns, such as areca nuts [2], [8], [20], [21]. Therefore, this study focuses on the optimization of Faster R-CNN by integrating Haar-like Features, Integral Image, and Anchor Box Adjustment to enhance detection and classification accuracy [20]-[24]. The use of Haar-like features enables the model to more effectively recognize texture variations, while Integral Image accelerates feature extraction, and Anchor Box optimization is adjusted to dataset characteristics to improve detection accuracy. Through this approach, the study aims to overcome dataset limitations and improve areca nut detection performance, making the system more effective for automatic applications in the agricultural and food processing industries[10], [25]-[28].

Thus, this research not only explores Deep Learning methods for areca nut detection but also identifies key challenges in model implementation, particularly related to dataset quality, annotation methods, model architecture selection, and class imbalance in training data. Moving forward, dataset improvements using COCO-style annotation, dataset expansion with augmentation techniques, and the adoption of more optimal models could be strategic steps to further enhance the automated detection and classification of areca nuts at an industrial scale.

2. Methods

In this study, an optimized Faster R-CNN model is proposed for areca nut detection and classification, integrating Haar-like Features, Integral Image processing, and an optimized anchor box configuration. The methodology aims to address dataset limitations, including low-resolution images, inconsistent lighting, complex backgrounds, and annotation inaccuracies, which affect detection performance. The proposed approach enhanced feature extraction and object localization, allowing for better classification and bounding box regression.

As illustrated in Figure 1, the methodology follows a structured pipeline. Initially, Haar-like Features are extracted, focusing on edge, line, and center-surround features, to enhance texture recognition. Next, an Integral Image representation is applied to accelerate feature computation and improve object differentiation. The processed image is then passed to a ResNeXt-101 backbone, which extracts high-level feature maps. The Region Proposal Network (RPN) generates potential bounding box candidates, which are refined through ROI pooling and classification layers. Finally, the model outputs bounding box coordinates and classification scores, which are evaluated using train loss, accuracy, recall, F1-score, and mean average precision (mAP).

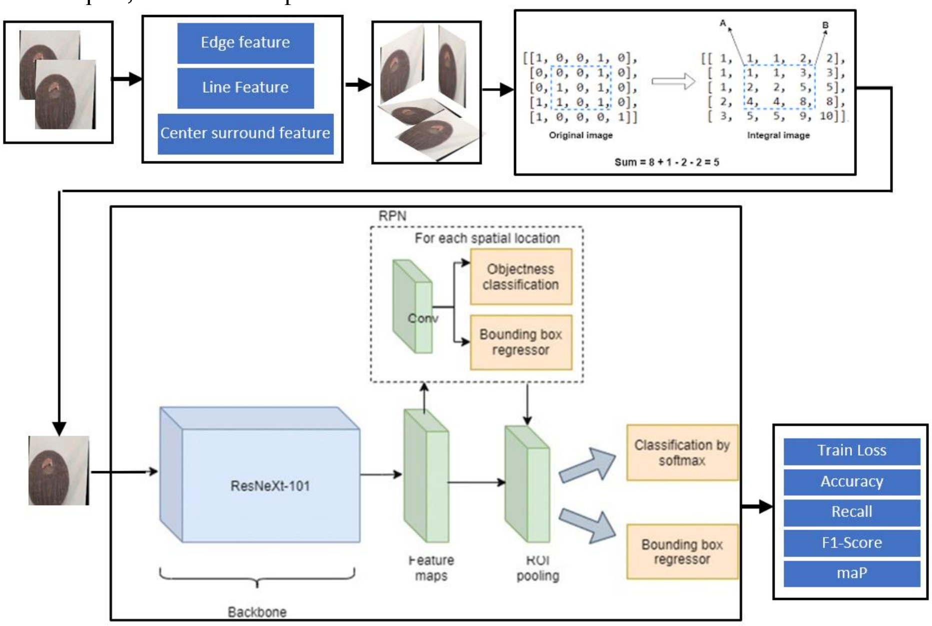

Figure 1. The proposed Faster R-CNN pipeline enhanced with Haar-like feature extraction and integral image computation. The system flows sequentially from input preprocessing to object detection and evaluation.

From Figure 1, it can be seen the integration of Haar-like Features and Integral Image processing in the Faster R-CNN pipeline aims to enhance detection accuracy and classification reliability for areca nut detection. By improving feature extraction and object localization, this approach mitigates dataset limitations, leading to better model generalization and performance evaluation. Figure 1 above presents the experimental setup and results, evaluating the model's effectiveness against conventional approaches.

The next section is about several critical issues that significantly impact model performance of areca nut. As seen in Figure 2, the dataset suffers from low image resolution, inconsistent lighting conditions, and cluttered backgrounds, making feature extraction challenging. Additionally, bounding box annotations appear misaligned or incomplete, leading to incorrect object localization and reduced Intersection over Union (IoU) scores. The presence of black padding and rotated perspectives further disrupts the uniformity of the dataset, affecting the model's ability to learn robust representations. These dataset limitations contribute to lower recall and precision values, emphasizing the need for higher-quality images, better annotation precision, and dataset augmentation to improve the overall detection and classification accuracy. Here are the issues of areca nut dataset in this research.

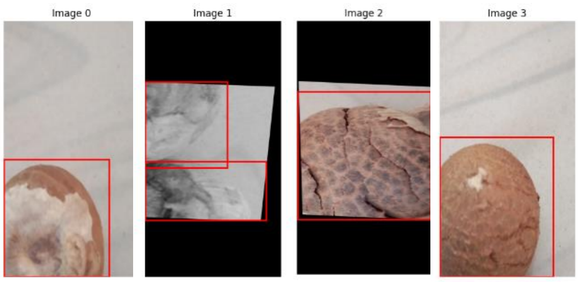

Figure 2. Sample images from the Areca Nut dataset, showing variations in angle, lighting, and background conditions.

Table 1 Issues of Areca Nut Dataset

Issue	Impact on Model Performance	Suggested Improvement
Low Image Quality	Poor feature extraction	Use high-resolution images
Lighting Variations	Model struggles with detection in different conditions	Normalize lighting or use data augmentation
Background Clutter	Misclassification due to unwanted features	Use clean, uniform backgrounds
Labelling Errors (Bounding Box Inaccuracy)	Reduces IoU and mAP scores	Ensure precise annotation using COCO format
Dataset Size (Small Number of Images)	Overfitting and poor generalization	Increase dataset size (COCO-like dataset)
Class Imbalance	Model favors majority class, poor recall	Balance the number of samples per class

From Table 1, it is evident that several dataset-related issues significantly affect model performance. Low image quality limits feature extraction, while lighting inconsistencies cause appearance variations that hinder generalization. Background clutter leads to misclassification, and inaccurate bounding boxes. Additionally, the small dataset size increases the risk of overfitting, and class imbalance skews predictions toward dominant categories. Addressing these limitations through improved image quality, accurate annotations, and balanced data distribution is essential to enhance detection outcomes.

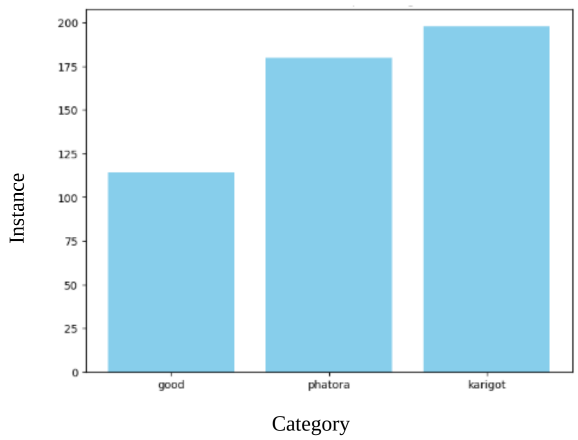

Figure 3. Instance Category Distribution of Arecanut for Detection

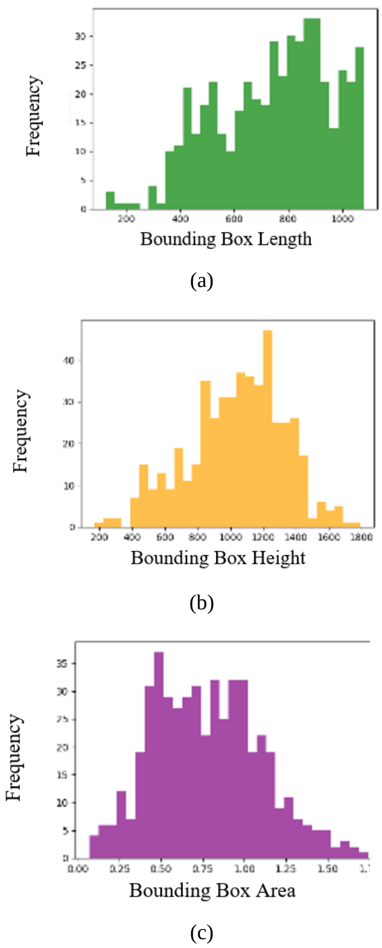

Figure 4 The (a) Pixel Length, (b) Pixel Height, (c) Pixel Area of Bounding Box Distribution of Arecanut Dataset for Detection

The bar chart in Figure 3 illustrates the distribution of the areca nut dataset, comparing the number of samples categorized as good, phatora, and karigot. The results indicate variations in classification counts, with karigot having the highest frequency, followed by phatora, while good quality areca nuts are the least represented.

The histogram plots illustrate in Figure 4 is the bounding box distribution in the areca nut dataset for object detection. The first plot represents the bounding box length, the second shows the bounding box height, and the third depicts the bounding box area distribution. The variations in pixel values indicate size inconsistencies, which may impact detection accuracy and require anchor box optimization for better model performance. The dataset distribution is visualized in Figure 4.

Figure 5 illustrates the distribution of aspect ratios in the areca nut dataset for object detection. The majority of bounding boxes have an aspect ratio close to 1, indicating near-square shapes, while a few samples have significantly higher values. This imbalance may affect the region proposal network (RPN) and suggests the need for optimized anchor box aspect ratios to improve detection accuracy.

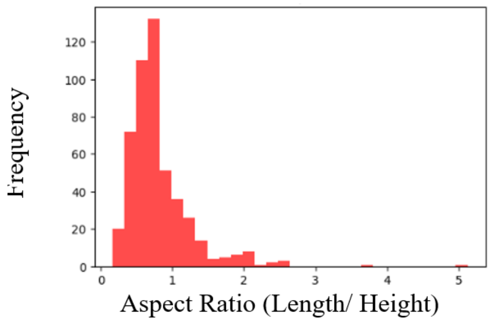

Figure 5 The Aspect Ratio Distribution of Arecanut Dataset.

This section presents the modified Faster R-CNN code, incorporating Haar-like Features and Integral Image to enhance object detection. The modifications include key functions such as load_model(), which loads the model and applies Haar-like Features & Integral Image, train_one_epoch() for training the model for one epoch, validate() for model validation, save_checkpoint() to store model checkpoints, and plot_loss() to visualize the loss curve during training. The integration of Haar-like Features & Integral Image is applied during pre-processing before inputting data into Faster R-CNN. Additionally, the code has been optimized for modularity and readability, making it easier to understand and implement. Here is the stage of pre-processing using Haar-like Features and Integral Image in Pseudocode 1.

Pseudocode 1: Pre-Processing using Haar-like Features and Integral Image

```
BEGIN CONFIGURATION
    SET DEVICE to CUDA if available, otherwise use CPU
    SET LEARNING_RATE to 0.005
    SET NUM_EPOCHS to 30
    SET PRINT_EVERY to 10
    SET CHECKPOINT_PATH to "faster_rcnn_checkpoint.pth"
    SET BEST_MODEL_PATH to "best_faster_rcnn.pth"
    SET LOG_FILE to "training_log.json"
END CONFIGURATION
DEFINE FUNCTION apply_haar_integral(img)
    INPUT: img (input image)
    OUTPUT: integral_img (Integral Image), objects (detected
objects)
    BEGIN
        CONVERT img to grayscale and STORE as gray
        COMPUTE Integral Image from gray and STORE as
integral_img
        LOAD Haar Cascade Classifier and STORE as haar_cascade
        DETECT objects using haar_cascade on gray
            SET sc
```

In the Pre-Processing stage, images are converted to grayscale to simplify feature extraction, followed by the computation of Integral Image to accelerate Haar-like feature calculations. The model utilizes a Haar Cascade Classifier to detect relevant patterns in areca nuts, producing output in the form of converted images and a list of detected objects. This process aims to enhance the model's ability to recognize patterns before being processed by Faster R-CNN. The process of load model of Faster R-CNN can be seen in Pseudocode 2.

Pseudocode 2: Load Model Faster R-CNN

```
BEGIN FUNCTION load_model()
    INITIALIZE model as Faster R-CNN with ResNet-50 FPN
backbone, pretrained on COCO dataset
    COMPUTE num_classes as total categories in train_dataset +
1 (for background)
    GET number of input features for classification layer
    REPLACE model's classification head with FastRCNNPredictor
using num_classes
    TRANSFER model to DEVICE
    RETURN model
END FUNCTION
```

The Load Model Faster R-CNN stage aims to prepare the model architecture by modifying the classification layer to match the number of classes in the dataset. The model used is the Faster R-CNN with a ResNet-50 FPN backbone, pretrained on the COCO dataset. It is then adjusted to accommodate the specific number of areca nut classes, transferred to the appropriate device (GPU or CPU), and made ready for training. Here is Pseudocode 3 about the training.

Pseudocode 3: Training

```
FUNCTION train_one_epoch(model, train_loader, optimizer):
    SET model to training mode
    INITIALIZE total_train_loss = 0
    FOR batch_idx, (imgs, targets) in train_loader:
        APPLY Haar-like Features & Integral Image to imgs
        CONVERT imgs and targets to DEVICE
        COMPUTE loss from model
        RESET and UPDATE optimizer with backpropagation
        ADD loss to total_train_loss
        IF batch_idx % PRINT_EVERY == 0: UPDATE progress bar
```

In the model training stage, the model enters training mode, where each input image is processed using Haar-like Features and Integral Image before being fed into the model. This process involves a forward pass to compute the loss and a backward pass to update the model weights using backpropagation. During each iteration, the loss value is recorded and displayed to monitor training progress. With the addition of Haar-like Features, the model is expected to better recognize areca nut features. For the validation can be seen in Pseudocode 4.

Pseudocode 4: Validation

```
BEGIN FUNCTION validate(model, val_loader)
    SET model to training mode
    INITIALIZE total_val_loss to 0
    INITIALIZE valid_batches to 0
    DISABLE gradient computation
    FOR each batch (imgs, targets) in val_loader DO:
        CONVERT each img to NumPy array and apply Haar-like
Features & Integral Image
        CONVERT processed images to tensor format and move to
DEVICE
        CONVERT targets to DEVICE format
        IF all target boxes are empty THEN:
            CONTINUE to the next batch
        COMPUTE loss_dict from model using imgs and targets
        IF loss_dict is a dictionary AND is not empty THEN:
            COMPUTE total loss as sum of all values in
loss_dict
            ADD loss value to total_val_loss
            INCREMENT valid_batches by 1
```

The Model Validation stage is conducted to evaluate the model's performance on unseen data. Similar to training, each image is tested using Haar-like Features and Integral Image, but this time without updating the model weights. If a batch contains no object annotations, it is skipped to prevent errors in loss computation. The validation results are represented as the average loss, calculated based on the number of valid batches available. The Pseudocode 5 is showed about testing.

```
Pseudocode 5: Testing

BEGIN FUNCTION run_inference(model, image_path)
    READ image from image_path
    CONVERT image color format from BGR to RGB
    processed_img, detected_objects =
apply_haar_integral(image)
    TRANSFORM processed_img into a tensor
    ADD batch dimension and MOVE tensor to DEVICE
    DISABLE gradient computation
    GET predictions from model using img_tensor
    RETURN ori
```

After the model is trained, the Model Testing (Inference) stage is conducted to evaluate its ability to detect new objects. The test images are first converted to grayscale and processed using Haar-like Features and Integral Image before being transformed into tensor format for model processing. Inference is performed by feeding the images into the model, which then generates predicted bounding boxes and probability scores for each detected object. For the pseudocode of image detection can be seen in Pseudocode 6.

```
Pseudocode 6: Image Detection
FUNCTION visualize_results(image, predictions,
detected_objects, threshold=0.5):
    CREATE figure (8,6) and display image
    SET ax for plotting
    FOR (x, y, w, h) in detected_objects:
        DRAW green rectangle (Haar detection)
    FOR each box in predictions:
        IF score ≥ threshold:
            DRAW red rectangle (Faster R-CNN)
    REMOVE axes, SET title, DISPLAY plot
END FUNCTION
```

The image detection stage aims to display the prediction results obtained from Faster R-CNN. The bounding boxes detected using Haar-like Features are shown in green, while those detected by Faster R-CNN are displayed in red. This allows for a direct comparison of how effectively Haar-like Features assist the model in detecting areca nuts. The evaluation stage can be seen in Pseudocode 7.

```
Pseudocode 7: Evaluation
BEGIN EVALUATION METRICS
    COMPUTE precision as (true positives) / (true positives +
false positives)
    COMPUTE recall as (true positives) / (true positives +
false negatives)
    COMPUTE f1-score as 2 * (precision * recall) / (precision
+ recall)
     COMPUTE specificity as (true negatives) / (true negatives
+ false positives)
    COMPUTE mAP using the average precision across all classes
    COMPUTE accuracy as (true positives + true negatives) /
(total samples)
END EVALUATION
```

Finally, the model evaluation stage is conducted to measure the overall performance of the model using various metrics, including Precision, Recall, F1 Score, Specificity, mAP (Mean Average Precision), and Accuracy. Precision evaluates the proportion of correct detections among all positive detections, while Recall measures how many actual objects were successfully detected by the model. The F1 Score balances precision and recall, whereas Specificity indicates how well the model avoids false negative detections. mAP assesses the model’s detection performance across different probability thresholds, while Accuracy measures the proportion of correct predictions out of the total predictions made.

3. Results and Discussions

The following figure presents a comparative analysis of the training performance between the baseline Faster R-CNN and the optimized Faster R-CNN with Haar-like Features & Integral Image. The graphs illustrate key evaluation metrics, including train loss, accuracy, precision, recall, F1-score, and mean average precision (mAP) across training epochs.

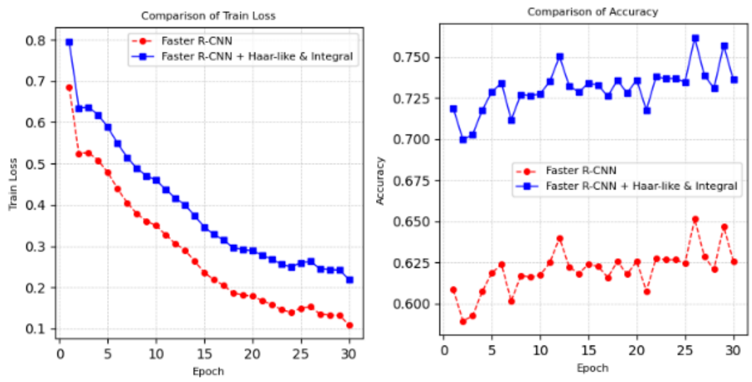

Figure 6. Comparation of (a) Train Loss, (b)Accuracy of Areca Nut Detection with Faster R-CNN using Haar-like Features and Integral Image.

Figure 6(a) compares the train loss of Faster R-CNN with and without Haar-like Features and Integral Image for areca nut detection. The standard Faster R-CNN (red) shows a faster decline in train loss, reaching around 0.1-0.15, while the modified model (blue) stabilizes at a higher loss of 0.2-0.25. This suggests that Haar-like features add complexity, slowing convergence. However, the additional features may improve detection accuracy, making further evaluation necessary to determine the best trade-off between loss reduction and model performance. Figure 6(b) compares the accuracy of Faster R-CNN with and without Haar-like Features and Integral Image. The modified model (blue) consistently achieves higher accuracy, stabilizing around 0.72-0.75, while the standard Faster R-CNN (red) remains lower, around 0.60-0.65. This suggests that the additional feature extraction improves object recognition, leading to better classification performance. Despite a slightly higher training loss, the modified model demonstrates superior accuracy, making it a more effective approach for areca nut detection.

Figure 7(a) compares the precision of Faster R-CNN with and without Haar-like Features and Integral Image. The modified model (blue) consistently achieves higher precision, fluctuating between 0.75 and 0.90, while the standard Faster R-CNN (red) remains lower, around 0.60-0.75 with more instability. This indicates that the additional feature extraction helps reduce false positives, making the modified model more reliable in detecting areca nuts accurately. Figure 7(b) compares the recall of Faster R-CNN with and without Haar-like Features and Integral Image. The modified model (blue) consistently achieves higher recall, ranging between 0.35 and 0.45, while the standard Faster R-CNN (red) remains lower, around 0.25-0.35 with more fluctuations. This indicates that the additional feature extraction improves the model's ability to detect more true positives, making it more effective in identifying areca nuts.

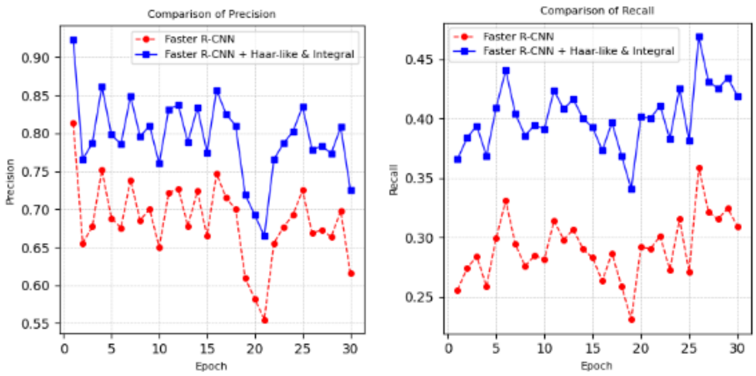

Figure 7. Comparation of (a) Precision, and (b)Recall of Areca Nut Detection with Faster R-CNN using Haar-like Features and Integral Image.

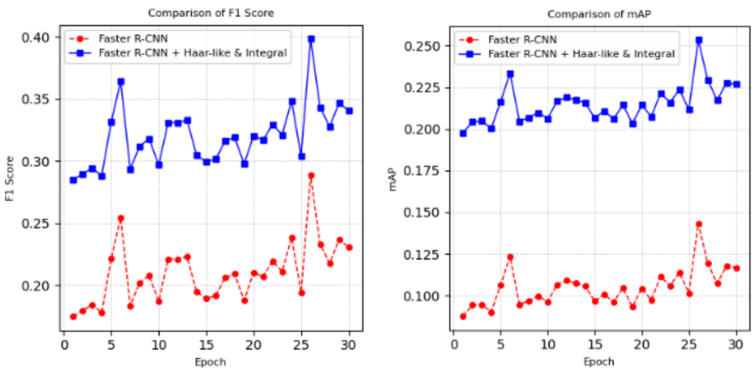

Figure 8 (a). Comparation of (a) F1-Score, and (b)MaP of Areca Nut Detection with Faster R-CNN using Haar-like Features and Integral Image.

Figure 8(a) presents a comparative analysis of F1 Score between the standard Faster R-CNN (red) and the proposed modification incorporating Haar-like Features and Integral Image (blue). The enhanced model consistently achieves a higher and more stable F1 Score, ranging from 0.30 to 0.40, while the baseline model fluctuates between 0.20 and 0.30. This performance gap reflects the effectiveness of the proposed feature extraction mechanism in balancing precision and recall. Furthermore, the smoother curve of the modified model suggests better generalization capability and reduced overfitting tendencies, likely due to the additional spatial-contextual encoding provided by Haar-like filters. These findings confirm that the integration of traditional vision-based features with deep learning enhances robustness in small, imbalanced datasets like the areca nut set. Figure 8(b) compares the mean Average Precision (mAP) of Faster R-CNN with and without Haar-like Features and Integral Image. The modified model (blue) consistently achieves a higher mAP, stabilizing around 0.20-0.25, while the standard Faster R-CNN (red) remains lower, around 0.10-0.13 with more fluctuations. This indicates that the additional feature extraction improves overall detection performance, making the modified model more effective in accurately identifying areca nuts.

The results in Table 2, demonstrate a significant improvement in the performance of the Faster R-CNN model after integrating Haar-like Features and Integral Image, particularly in accuracy, precision, recall, F1-score, and mean average precision (mAP). The optimized model achieved a 12% increase in accuracy (from 0.64 to 0.76), a 15% improvement in precision (from 0.70 to 0.85), and an increase in recall from 0.32 to 0.45, leading to a higher F1-score (from 0.26 to 0.38). Additionally, the mAP showed a significant boost from 0.13 to 0.25, indicating better object localization and classification capabilities. These improvements confirm that the proposed enhancements successfully improve the feature extraction process and detection efficiency. Despite these gains, the training loss increased from 0.70 to 0.75, suggesting that the modified model requires more computational resources and training iterations to converge effectively. This higher loss could also be attributed to the increased complexity introduced by Haar-like Features and Integral Image, which may add more noise to the feature maps if not properly optimized.

Table 2. The Result of Areca Nut Detection and Classification Using Faster R-CNN

Metric	Faster R-CNN	Faster R-CNN + Haar-like & Integral
Train Loss	0.70	0.75
Accuracy	0.64	0.76
Precision	0.70	0.85
Recall	0.32	0.45
F1 Score	0.26	0.38
mAP	0.13	0.25

While the proposed method demonstrates improvements in accuracy and robustness, certain dataset-related constraints remain. Factors such as low image resolution, varying lighting conditions, and occasional annotation inconsistencies may have contributed to minor detection errors and a slightly reduced recall. Moreover, class imbalance, particularly the underrepresentation of specific areca nut conditions, could limit the model's generalizability. Nonetheless, the method's ability to maintain strong performance despite these challenges underscores its robustness and potential for deployment in real-world agricultural settings.

Additionally, class imbalance, where certain areca nut conditions are underrepresented, may have affected the model's ability to generalize, causing a disparity between precision and recall. To further enhance performance, dataset improvements should be prioritized. Higher-resolution images, better annotation strategies using COCO-style datasets, and increased sample diversity can help the model learn more robust features. Additionally, optimizing anchor box configurations and applying more targeted data augmentation can help mitigate class imbalance issues. By addressing these dataset challenges, future iterations of the model could achieve even greater detection accuracy and stability in real-world applications.

The detection results in Figure 7 show improved object localization and feature recognition using Faster R-CNN with Haar-like Features and Integral Image, but also highlight dataset limitations. The model achieves better predictions but still struggles with misaligned detections and false positives, likely due to low-resolution images, inconsistent lighting, complex backgrounds, and annotation errors. From the performance evaluation table 2, the optimized model improved accuracy (76% vs. 64%), precision (85% vs. 70%), recall (45% vs. 32%), and mAP (0.25 vs. 0.13), confirming the benefits of Haar-like Features and Integral Image in enhancing texture detection. However, increased training loss (0.75 vs. 0.70) suggests added computational complexity. Additionally, class imbalance contributed to lower recall in detecting defects. Despite these challenges, the proposed enhancements significantly improve fine-grained feature detection in areca nuts. Future improvements should focus on better dataset quality, precise annotations, and class balancing, along with further anchor box and model optimization to reduce false detections and enhance performance. Table 3 presents a comparative summary between the proposed Faster R-CNN approach and several state-of-the-art (SOTA) object detection models, focusing on aspects such as dataset suitability, architectural design, optimization strategy, application target, and detection speed.

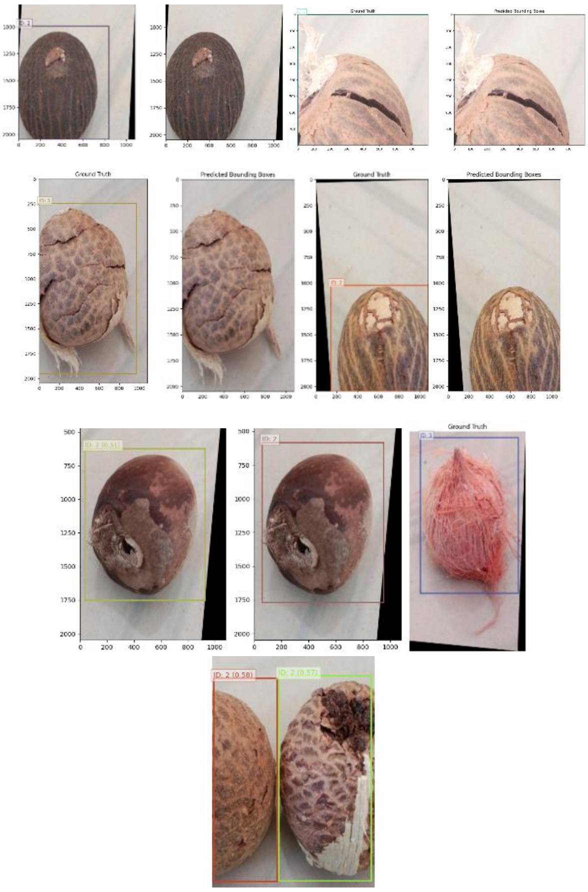

Figure 7. Arecanut Detection Result of Faster R-CNN with Haar-like & Integral

As summarized in Table 3, while SOTA models such as YOLOv7 and DETR offer superior detection speed and performance under large-scale and diverse datasets, the proposed Faster R-CNN model demonstrates significant strength in handling small, local datasets. The integration of Haar-like features and manual anchor box optimization contributes to a more robust performance in constrained agricultural settings, such as areca nut detection. Although the FPS is relatively lower, the model prioritizes accuracy and contextual suitability, making it practical for field-level deployments where computing resources are limited but precision is critical

Table 3.Comparative Summary between Proposed Method and State-of-the-Art Object Detection Models

Aspect	Faster R-CNN (This Work)	SOTA Models (YOLOv7, ViT[4], [5], [6], [29])
Dataset Suitability	Effective for small and locally constrained datasets	Require large and diverse datasets
Architectural Innovation	Haar-like feature integration and integral image	CNN or Transformer-based architectures
Anchor Optimization	Manually tuned based on dataset distribution	Default or auto-tuned anchor boxes
Target Applications	Suitable for agriculture-specific and local contexts	General-purpose multi-class object detection
Detection Speed (FPS)	Moderate (10–12 FPS on GPU)	High (30–70 FPS), requires higher compute resources

4. Conclusions

Overall, the results demonstrate that the optimized Faster R-CNN model significantly outperforms the baseline, with a 12% improvement across key metrics, accuracy, precision, recall, F1-score, and mean average precision (mAP). The integration of Haar-like Features and Integral Image proves effective in enhancing feature extraction and classification accuracy, contributing to a more robust and accurate object detection system. Although the proposed enhancements introduce slightly higher training loss due to added complexity, this trade-off is well-justified by the substantial gains in detection performance. While the model shows strong generalization capabilities, some dataset-related constraints remain. Factors such as low image resolution, inconsistent lighting, complex backgrounds, and occasional annotation inaccuracies may influence detection consistency. In particular, class imbalance and the limited representation of certain areca nut conditions may affect recall performance. Nonetheless, the model's ability to achieve significant improvements despite these challenges underscores its resilience and practical potential. To further advance this work, future efforts should focus on improving dataset quality through high-resolution image acquisition, precise annotations, and consistent formatting (e.g., COCO-style). Enhancing class balance, optimizing anchor box configurations, and exploring advanced augmentation techniques may also contribute to increased model robustness and generalization. With these refinements, the proposed approach holds strong potential as a reliable solution for automated areca nut detection and classification, offering meaningful benefits in agricultural and industrial contexts.

Acknowledgements

This work was supported by a research grant from the Dinamika Bangsa Foundation. Thank you for the material and non-material support that has been provided.

References

[1] M. J. Hasan *et al.*, "Nuclei Segmentation in ER-IHC Stained Histopathology Images using Mask R-CNN," in *2022 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, Nov. 2022, pp. 1–4. doi: 10.1109/ISPACS57703.2022.10082832.

[2] M. Balipa, P. Shetty, A. Kumar, and B. R. Puneeth, "Arecanut Disease Detection Using CNN and SVM Algorithms," presented at the International Conference on Artificial Intelligence and Data Engineering, AIDE 2022, 2022, pp. 64–67. doi: 10.1109/AIDE57180.2022.10060130.

[3] M. R. Borroek, E. Rasywir, Y. Pratama, and M. Istoningtyas, "Analysis on Knowledge Layer Application for Knowledge Based System," *2018 International Conference on Electrical Engineering and Computer Science …*. 2018.

[4] L. S. Nair, R. P. R, G. Sugathan, K. V. Gireesh, and A. S. Nair, "Mitotic Nuclei Detection in Breast Histopathology Images using YOLOv4," in *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Jul. 2021, pp. 1–5. doi: 10.1109/ICCCNT51525.2021.9579969.

[5] S. Pandey, K. -F. Chen, and E. B. Dam, "Comprehensive Multimodal Segmentation in Medical Imaging: Combining YOLOv8 with SAM and HQ-SAM Models," in *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, Oct. 2023, pp. 2584–2590. doi: 10.1109/ICCVW60793.2023.00273.

[6] H. Zhang *et al.*, "An Improved Lightweight Yolo-Fastest V2 for Engineering Vehicle Recognition Fusing Location Enhancement and Adaptive Label Assignment," *IEEE J. Sel. Top. Appl. Earth Observations Remote Sensing*, vol. 16, pp. 2450–2461, 2023, doi: 10.1109/JSTARS.2023.3249216.

[7] P. M. Naik and B. Rudra, "Classification of Arecanut X-Ray Images for Quality Assessment Using Adaptive Genetic Algorithm and Deep Learning," *IEEE Access*, vol. 11, pp. 127619–127636, 2023, doi: 10.1109/ACCESS.2023.3332215.

[8] V. Arun Karthik, J. Shivaprakash, and D. Rajeswari, "Disease Detection in Arecanut using Convolutional Neural Network," presented at the Proceedings - 3rd International Conference on Advances in Computing, Communication and Applied Informatics, ACCAI 2024, 2024. doi: 10.1109/ACCAI61061.2024.10602152.

[9] C. A. Rahmat, H. Permatasari, E. Rasywir, and Y. Pratama, "Penerapan K-Means Untuk Clustering Kondisi Gizi Balita Pada Posyandu," *Jurnal Media Informatika Budidarma*, vol. 7, no. 1. pp. 207–213, 2023.

[10] A. N. N. Azmi, S. Khairunniza-Bejo, M. Jahari, and I. Yule, "Early Detection of Plant Disease Infection Using Hyperspectral Data and Machine Learning," in *IoT and AI in Agriculture*, T. Ahamed, Ed., Singapore: Springer Nature Singapore, 2023, pp. 423–446. doi: 10.1007/978-981-19-8113-5_21.

[11] F. Fahmi, D. Trianda, U. Andayani, and B. Siregar, "Image processing analysis of geospatial uav orthophotos for palm oil plantation monitoring," *J. Phys.: Conf. Ser.*, vol. 978, p. 012064, Mar. 2018, doi: 10.1088/1742-6596/978/1/012064.

[12] I. Mohd Shukri, M. A. Izzuddin, R. Mohd Hefni, and A. S. Idris, "Geostatistics of Oil Palm Trees Affected by Ganoderma Disease in Low and High Planting Density," *IOP Conf. Ser.: Earth Environ. Sci.*, vol. 540, no. 1, p. 012065, Jul. 2020, doi: 10.1088/1755-1315/540/1/012065.

[13] H. Fatakdawala *et al.*, “Expectation–Maximization-Driven Geodesic Active Contour With Overlap Resolution (EMaGACOR): Application to Lymphocyte Segmentation on Breast Cancer Histopathology,” *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 7, pp. 1676–1689, Jul. 2010, doi: 10.1109/TBME.2010.2041232.

[14] W. Li, H. Fu, L. Yu, and A. Cracknell, “Deep Learning Based Oil Palm Tree Detection and Counting for High-Resolution Remote Sensing Images,” *Remote Sensing*, vol. 9, no. 1, p. 22, Dec. 2016, doi: 10.3390/rs9010022.

[15] T. Seesaard, N. Goel, M. Kumar, and C. Wongchoosuk, “Advances in gas sensors and electronic nose technologies for agricultural cycle applications,” *Computers and Electronics in Agriculture*, vol. 193, p. 106673, Feb. 2022, doi: 10.1016/j.compag.2021.106673.

[16] P. Singh, A. Verma, and J. S. R. Alex, “Disease and pest infection detection in coconut tree through deep learning techniques,” *Computers and Electronics in Agriculture*, vol. 182, p. 105986, Mar. 2021, doi: 10.1016/j.compag.2021.105986.

[17] C. Raghuram and M. Thenmozhi, “Short Review on Contrastive Learning-based Segmentation Techniques for Medical Image Processing,” in *2023 International Conference in Advances in Power, Signal, and Information Technology (APSIT)*, Jun. 2023, pp. 290–296. doi: 10.1109/APSIT58554.2023.10201707.

[18] A. K V, S. Krishna, and R. R. Paulose, “Computer-Aided Detection of Nuclei from Histopathology Image: Evaluating the Effectiveness of U-Net and Modified Networks for Nuclei Segmentation,” in *2023 3rd International Conference on Intelligent Technologies (CONIT)*, Jun. 2023, pp. 1–8. doi: 10.1109/CONIT59222.2023.10205653.

[19] P. Ahmadi, S. B. Mansor, H. Ahmadzadeh Araji, and B. Lu, “Convolutional SVM Networks for detection of Ganoderma Boninense At Early Stage in oil palm using UAV and Multispectral Pleiades Images,” *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, vol. X-4/W1-2022, pp. 25–30, Jan. 2023, doi: 10.5194/isprs-annals-X-4-W1-2022-25-2023.

[20] A. Gupta, B. S. Rajeshwari, B. Ambadas, J. Shreyas, and G. Bhuvan, “Analysis Of Areca Nut Leaf Pathology And Recommendation System Using Generative AI,” in *2024 Second International Conference on Networks, Multimedia and Information Technology (NMITCON)*, Bengaluru, India: IEEE, Aug. 2024, pp. 1–8. doi: 10.1109/NMITCON62075.2024.10698896.

[21] R. Dinesh and N. K. Bharadwaj, “Possible approaches to arecanut sorting / grading using computer vision: A brief review,” presented at the Proceeding - IEEE International Conference on Computing, Communication and Automation, ICCCA 2017, 2017, pp. 1007–1014. doi: 10.1109/CCAA.2017.8229971.

[22] F. Fachruddin, S. Saparudin, E. Rasywir, Y. Pratama, and B. Irawan, “Extraction of object image features with gradation contour,” *TELKOMNIKA*, vol. 19, no. 6, p. 1913, Dec. 2021, doi: 10.12928/telkomnika.v19i6.19491.

[23] A. J. Suresh and L. S. Nair, “Mitosis Detection In Breast Histopathology Image Using Ensemble Features Fed CNN Model,” in *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Jul. 2023, pp. 1–5. doi: 10.1109/ICCCNT56998.2023.10307867.

[24] J. Li, Y. Wu, X. Zheng, and S. Dong, “Supervoxels-based Self-supervised Few-shot 3D Medical Image Segmentation via Multiple Features Transfer,” in *2024 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, Dec. 2024, pp. 3422–3427. doi: 10.1109/BIBM62325.2024.10822846.

[25] N. A. Husin, S. Khairunniza–Bejo, A. F. Abdullah, M. S. M. Kassim, and D. Ahmad, “Study of the oil palm crown characteristics associated with Basal Stem Rot (BSR) disease using stratification method of point cloud data,” *Computers and Electronics in Agriculture*, vol. 178, p. 105810, Nov. 2020, doi: 10.1016/j.compag.2020.105810.

[26] M. Isnan, A. A. Hidayat, and B. Pardamean, “Indonesian Agricultural-crops Classification Using Transfer Learning Model,” *Procedia Computer Science*, vol. 227, pp. 128–136, 2023, doi: 10.1016/j.procs.2023.10.510.

[27] A. Y. Khaled, S. Abd Aziz, S. Khairunniza Bejo, N. Mat Nawi, D. Jamaludin, and N. U. A. Ibrahim, “A comparative study on dimensionality reduction of dielectric spectral data for the classification of basal stem rot (BSR) disease in oil palm,” *Computers and Electronics in Agriculture*, vol. 170, p. 105288, Mar. 2020, doi: 10.1016/j.compag.2020.105288.

[28] M. Maray *et al.*, “Artificial intelligence-enabled coconut tree disease detection and classification model for smart agriculture,” *Computers and Electrical Engineering*, vol. 104, p. 108399, Dec. 2022, doi: 10.1016/j.compeleceng.2022.108399.

[29] M. B. A. Gibril *et al.*, “Large-scale assessment of date palm plantations based on UAV remote sensing and multiscale vision transformer,” *Remote Sensing Applications: Society and Environment*, vol. 34, p. 101195, Apr. 2024, doi: 10.1016/j.rsase.2024.101195.