Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/issn/15375110

Research Paper

Detection of maturity stages of coconuts in complex background using Faster R-CNN model

Subramanian Parvathi[*], *Sankar Tamil Selvi*

Department of ECE, National Engineering College, K.R. Nagar, Kovilpatti, 628503, Tamil Nadu, India

ARTICLE INFO

Article history:
Received 6 May 2020
Received in revised form
3 December 2020
Accepted 8 December 2020
Published online 2 January 2021

Keywords:
Coconut image acquisition
Coconut detection
Faster R–CNN
Deep learning

Coconuts are commonly harvested by judging their maturity based on colour, shape, timeframe, shaking sound, and other growth characteristics of changes as they grow. Currently, solutions involving image-processing techniques have substantial challenges involving the identification of the maturity stages of coconuts. Accordingly, an improved faster region-based convolutional neural network (Faster R–CNN) model is proposed for the detection of two important maturity stages for coconuts in complex backgrounds. The detection of the maturation stages of coconuts for harvesting without human intervention involves challenges because of the complexity of the environment and the similarity between fruits and their backgrounds. Images of coconut and mature coconut bunches were collected from coconut farms. These images were augmented using rotation and colour transformation techniques. These augmented images were used along with original images during model training. The Faster R–CNN algorithm with the ResNet-50 network was used to enhance the detection score of nuts with two major maturity stages. Following training, the detection performance was tested with a dataset that included real-time images as well as Google images. The test results showed that the detection performance achieved using the improved Faster R–CNN model was greater than that for other object detectors such as the single shot detector (SSD) you only look once (YOLO-V3) and Region-based Fully Convolutional Networks (R–FCN). The promising results obtained from this study provided the motivation to develop an application tool for detecting coconut maturity from real-time images on farms.

1. Introduction

For coconut production, India ranks third in the world in terms of the geographical area used and the number of coconuts produced with Tamil Nadu accounting for 31% of the total production in India. Coconut harvesting is carried out for drinking purposes and for commercial nut production. Nuts harvested with the maturity after 12 months are used for commercial use, and after 7–9 months for drinking purposes. The challenges involved in coconut harvesting include labour supply and high labour cost due to crop height and its complex trunk structure. Although the demand for coconuts is increasing, farmers face the prospect of severe financial

* *Corresponding author.*
E-mail addresses: srathi2611@gmail.com (S. Parvathi), stsece@nec.edu.in (S. Tamil Selvi).
https://doi.org/10.1016/j.biosystemseng.2020.12.002
1537-5110/

Nomenclature

ANN	Artificial neural networks
AP	Average precision
CHT	Circular Hough transform
CNN	Convolutional neural network
CPU	Central processing unit
CUDA	Compute unified device architecture
DL	Deep learning
DSLR camera	Digital single-lens reflex camera
F1 score	Harmonic mean of the precision and recall values
Faster R–CNN	Faster region-based convolutional neural network
GA	Genetic algorithms
GPU	Graphics processing unit
IoU	Intersection-over-union
mAP	Mean average precision
OpenCV	Open computer vision library
PASCAL VOC	PASCAL visual object classes
RAM	Random access memory
R–FCN	Region-based fully convolutional networks
RPN	Region proposal network
SSD	Single shot detector
SVM	Support vector machine
YOLO	You only look once

losses. Hence, smart solutions are required to judge the maturity of the nuts to overcome this problem and increase production.

The technology involved in object detection in real-time is based on image processing applications. The development of a fast and reliable coconut detection tool using image analysis with machine vision involves challenges with regarding the colour, size, texture, shape, illumination, and reflectance properties of the nut. Another vital issue pertains to occlusions and shadows in the tree crown, which can significantly effect target detection.

Currently, automation in agriculture in general has made the continuous monitoring of crop growth possible. This has increased crop quality and offers the possibility of reduced labour costs by using vision-based techniques to detect and picking objects and using visual information for harvesting (Bac et al., 2014; Tzounis et al., 2017; Vasconez et al., 2019; Zhao et al., 2016). Robot end-effector manipulation for the picking of fruits or vegetables is the most critical and challenging task for agricultural robots. Accurate detection of fruit is the primary step for manipulating the end effector, because the robot must identify the location of the fruit for further action. The detection of fruit location is difficult in some cases because the area of the fruit and the background have the same visual appearance. Recently, technique of deep learning (DL) has experienced considerable developments in both its scope and application in agriculture. When compared with existing image processing techniques used in agriculture, DL will improve learning capability with an increase in detection accuracy and precision.

The workflow of the paper is as follows. Section 2 describes related works, and section 3 presents the materials and methods for coconut detection using DL with database creation. Section 4 describes the methodology for coconut detection using Faster R–CNN. The experimental results are discussed in section 5, and a performance evaluation of the coconut detection is analysed in section 6. Finally, conclusions are drawn in section 7.

2. Related works

Intelligent harvesting has the advantage of monitoring crop growth based on information from images. Illumination variations, similar and complex backgrounds, the distribution of fruits/vegetables, overlapping and occlusions, distances, and viewing angles of the camera are the most critical factors that affect the process of object detection. DL is an essential technique for solving many problems using machine vision and image analysis in agriculture. According to a survey by (Kamilaris & Prenafeta-Boldú, 2018) on the use of DL in agriculture, it offers better performance than existing popular image processing techniques. Further, a reliable and advanced DL technique made considerable progress in object classification and detection in agricultural applications (Krizhevsky et al., 2017; Russakovsky et al., 2015).

DL technology has been widely used in the agriculture areas such as in leaf disease classification (Amara et al., 2017; Sladojevic et al., 2016), fruit detection (Bargoti & Underwood, 2017; da Costa et al., 2020; Mureşan & Oltean, 2018), plant classification and identification (Dyrmann et al., 2016; Grinblat et al., 2016), fruit grading and counting (Song et al., 2014), and precision agriculture (Patrício & Rieder, 2018; Zhu et al., 2018). Segmentation and classification are the starting points for target detection of an image. For example, (Behroozi-Khazaei & Maleki, 2017) developed a robust algorithm for colour-based grape cluster segmentation using artificial neural networks (ANNs) and genetic algorithms (GAs) from a collected image dataset for segmenting grape clusters, leaves, and other objects in an image under limited lighting conditions. Target segmentation in a complex environment has been carried out for apple blossoms in a complex background by (Dias et al., 2018) using a convolutional neural network (CNN) and support vector machine (SVM).

DL-based object detection methods such as the “you only look once” (YOLO) method proposed by (Redmon et al., 2016; Redmon & Farhadi, 2017, 2018), bring target classification and localisation together as a regression problem. YOLO directly performs object detection using regression in the image and the state-of-the-art version (YOLO V3) has high detection accuracy for large-sized objects. Apple detection at different growth stages in orchards using the YOLO V3 DenseNet method was presented by (Tian et al., 2019), classifying young apples, expanding apples, and ripe apples, with excellent results. In their research, 160 images from each class were collected and augmented to create training sets with 4800 images. The authors achieved an F1 score (i.e. the harmonic mean of the precision and recall values) of 0.817, an intersection-over-union (IoU) of 0.896, and the average detection time was 0.304s per frame at a resolution of

3000 × 3000 pixels. Ghoury et al. (2019) investigated disease detection in grapes and grape leaves using the Faster R–CNN and SSD MobileNet architectures. The Faster-R-CNN Inception v2 performed better with a classification accuracy of between 78% and 99% for all testing images. However, Faster R–CNN had a longer processing time with better accuracy. SSD_MobileNet V1 produced results with an accuracy of between 90% and 99% for images with less noise, more uniform backgrounds and sizes. In addition, a high percentage of misclassifications occurred in this model for noisy and complex environments with less accurate results. It was shown that the SSD model struggled with the detection of smaller objects, which led to misdetection of the target object. Thus, the SSD model may not be useful for real-time classifications in tasks with smaller and clustered objects, but the Faster-R-CNN model performs more effectively.

The Faster R–CNN (Ren et al., 2017) uses region proposal networks (RPNs) and classification networks to detect the target accurately. (Bargoti & Underwood, 2017; Sa et al., 2016; Wan & Goudos, 2020) presented a fruit detection method using Faster R–CNN for a variety of fruits with excellent results.

When compared with the standard object-detection methods used in agriculture, deep convolutional networks have produced progressive results for complex environments. The feature extraction could be enriched by increasing the number of layers in the convolutional network. (Simonyan & Zisserman, 2015) indicated that the recognition accuracy increases with an increase in the depth of the network. However, owing to the gradient explosion during backward propagation, network training is not smooth. A deep network can be trained through batch normalisation and dropout; however, a precision decline problem occurs and results degrade rapidly. To solve the problem of precision degradation in the network and limitations concerning the depth of the network, (He et al., 2016) proposed a deep residual ResNet model using an identity-mapping concept. This method solved the degradation problem by fitting a residual map with a multilayer network. In a stacked layer structure, when the residual is zero, the stacked layer only makes an identify map and network performance does not decrease. If the residual map is not zero, then the stacked layers learn new features based on the input characteristics and produce a better performance. Thus, the application of ResNet in the network enhances feature extraction and effectively solves the precision degradation problem, thereby improves the recognition performance of the neural networks.

When acquiring images at coconut farms, illumination conditions are not constant, the backgrounds are complex and similar, and the camera viewing angle and distance are inconsistent. Coconut bunches are densely distributed, overlap in the tree crown, and are occluded by leaves and other objects that are present in the tree crown. All of these problems are significant challenges in the detection of coconut bunches in the field. The shape, colour, and texture of coconut bunches in several maturity stages are different. Coconut takes 12 months to mature (Niral et al., 2017). The two major maturity stages of coconuts i.e. tender (7–9 months) and mature (12 months) are used in this work. The size and shape of the coconuts varies from the starting time until maturation, while coconuts in a bunch have the same maturity level. Usually, 15–50 coconuts occur in a bunch when the yield is good. Nuts aged 12 months are harvested at intervals of 30–45 days. On average, eight harvests can occur per year, as coconut palms produce inflorescence every month.

In this research, green-coloured and round-shaped tender nuts are represented as coconuts. Yellow/brown-coloured and ovoid-shaped nuts are described as mature coconuts. In each stage, the colour and volume of the nuts will change. The maturity of a coconut was detected using DL methods. The state-of-the-art Faster R–CNN algorithm was used to detect coconut bunches in real-time. The Faster R–CNN network with ResNet-50 was used to improve the detection performance of a coconut bunch. Images of coconuts in the different maturity stages, including tender coconut/coconuts and mature coconuts, were collected and used as an input data for training the neural network. The trained neural network was used for detecting coconuts in a complex background.

3. Materials and methods

Coconut detection and location identification are crucial, because coconuts are located at various positions in the tree crown and within a complex background. However, the differentiation of coconuts from the environment based on colour alone is difficult, because the presence of other objects such as leaflets and leaf stalks interfere with the coconut bunches in the tree crown. Hence, colour-based segmentation is insufficient to detect coconuts. Figure 1 (a) shows an input RGB image, and Fig. 1 (b) shows the results from a colour-based segmentation method. It is clear from the output that the coconuts are not segmented accurately since other objects have colours similar to those of the coconut bunches. Edge detection and the application of the circular Hough transform (CHT) are insufficient to detect all of the coconuts in a bunch owing to overlapping conditions. The output of the CHT method is illustrated in Fig. 2, which shows the false and failed detection of coconuts.

Thus, existing image processing techniques are unable to detect all varieties of coconuts owing to the complexity involved in the environment. The detection of coconut maturity stages using deep convolutional neural networks is presented in the following sections, starting with the development of a coconut image database for detection.

3.1. *Image preprocessing*

3.1.1. *Image acquisition*

Database creation is an essential step for object detection using machine vision and image analysis. In this study, image acquisition was conducted using a digital single-lens reflex (DSLR) camera (Nikon D810), a Samsung camera, and a mobile phone camera at coconut farms in Tamil Nadu, India. Images were acquired from various places during sunny and cloudy conditions (from morning to evening) and images varying in their size, shape, brightness, and illumination were used to create dataset.

The following points are considered during image acquisition to improve the performance of detection of maturity

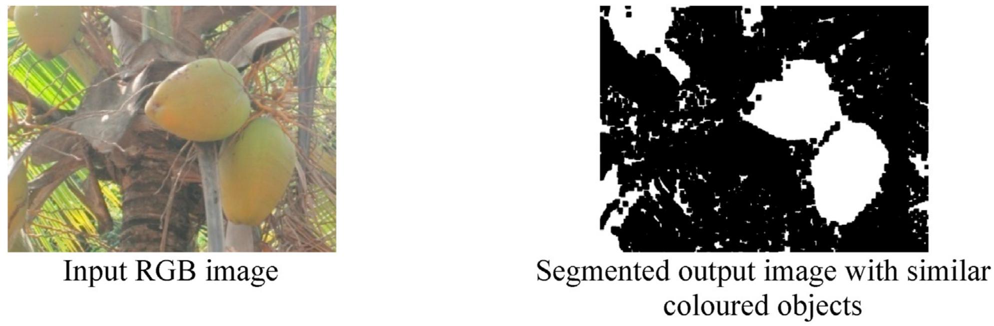

Fig. 1 – (a) Input RGB image (b) Segmentation output using morphological operations. Coconuts are presented in the tree crown with similar coloured objects. Colour-based segmentation is not sufficient since other objects in the tree crown interfere with coconut bunches.

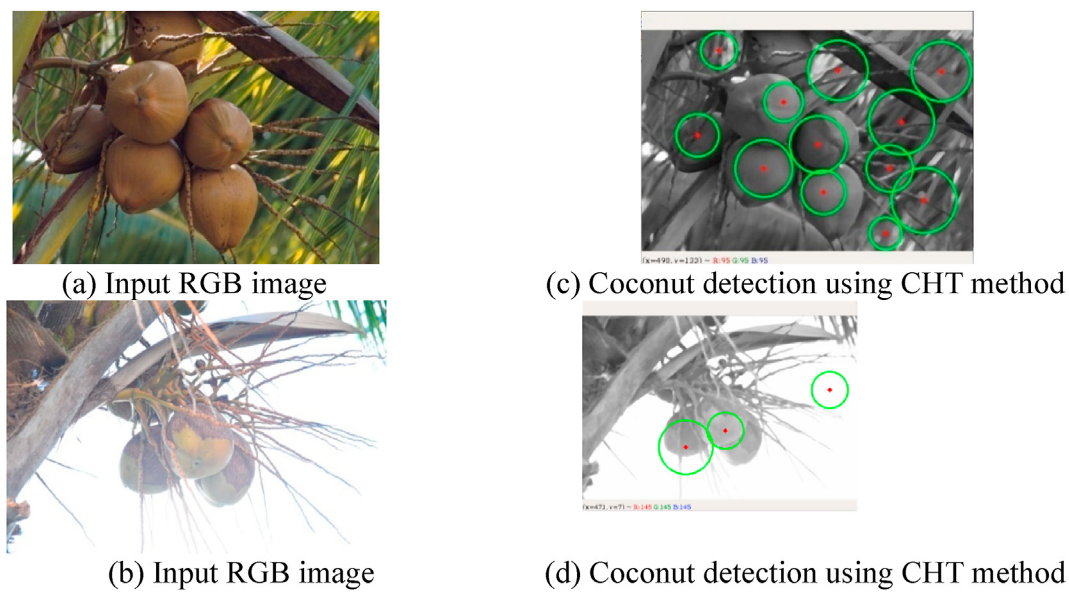

Fig. 2 – (a and b) Input RGB image (c and d) Output of CHT method for different images. CHT method failed to detect coconuts since they are overlapping with each other.

stages of coconuts with a sufficient mixture of training samples.

- Acquire images at different position angles to get pose variations of coconuts in the tree crown.
- Collection of images under different lighting conditions such as frontal lighting, backlighting, side lighting, and scattered lighting.

As sunlight passes through the tree canopy, more illumination variation occurs, and the object colour in the images is also affected. In Fig. 3, the coconuts in similar and complex background are illustrated.

From field visit, the coconut dataset has developed by acquiring images from various coconut farms in and around Tamil Nadu, India. From the dataset, two thousand images of coconuts with two major maturity stages are used for the coconut detection process with or without overlap with other objects. During image acquisition, some images were obtained from multiple viewing angles to improve detection performance. Among these 2000 sample images, 140 images were collected with numerous view angles.

These original images were then expanded using data augmentation methods to provide a training dataset that was used for training with DL models. The remaining images in the dataset and Google Images were used as the test dataset to verify the performance of the Faster R–CNN model.

3.1.2. *Image augmentation*

During image acquisition, the intensity and the angle with which the sunlight illuminates the target varies considerably. In this work, to improve the quality of the experimental dataset, the collected images were pre-processed using image augmentation techniques such as horizontal flip, image rotation by 90°, and colour transformation (saturation, hue, and exposure); all designed to ensure effective generalisation in the model for the detection of coconuts. Three randomly selected values were used to adjust the brightness of the

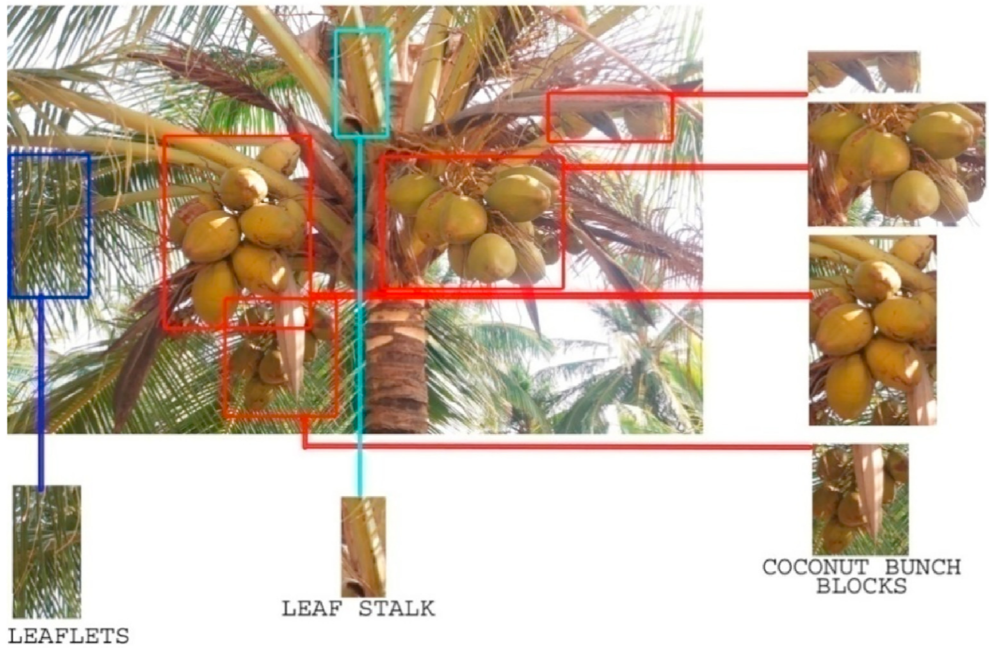

Fig. 3 – Coconuts in complex background with similar coloured objects in tree crown. The presence of coconuts with other objects which are similar in colour in the tree crown is illustrated in Figure 3.

original images. The newly adjusted images were then added to the training dataset to avoid the difficulty of drawing bounding boxes with low or high brightness values, and to obtain a clear view of the target edge during manual annotation. These images strongly influence the performance of coconut detection under different illumination conditions which can affect detection results in real-time. Hence, augmented images in the training dataset were blurred to handle all types of images to improve the robustness of the detection model.

In real-time environment, challenges such as incorrect focusing and camera-shake are captured but this could be avoided by acquiring images in multiple view angles to cover the entire coconut bunch.

3.1.3. *Image annotation and development of dataset*

The acquired images were numbered and manually annotated using LabelImg graphical image annotation tool which was written in Python. Bounding boxes were drawn for the coconuts of two categories. The images in the training set were then converted to PASCAL VOC format and YOLO format to compare the performance of the different algorithms. In the training set, the lengths of the images were rescaled to 516 pixels to maintain the original aspect ratio, and the widths were adjusted accordingly. During the creation of the training set, pictures with lower resolution and fewer dimensions were removed to maintain a dataset with good quality. In addition, images with moderate occlusions over the target area were selected to increase the detection performance. The complete image dataset is shown in Table 1 (after applying the augmentation methods).

4. Methodology

4.1. *Coconut detection using Faster R–CNN*

As stated earlier, the accurate detection of objects in complex backgrounds remains a challenging problem in computer vision and machine learning. Faster R–CNN was proposed to overcome this challenge by using an RPN. Faster R–CNN includes a convolutional layer, pooling layer, fully connected layer, and regression layer to classify the objects. The convolutional layer uses a semi-supervised learning and hierarchical feature extraction algorithm to obtain a feature map for the extraction of features. The convolutional layer acts as the feature extractor for RPN. The RPN applies sliding windows on these feature maps to obtain the object proposals along with their objectness scores.

In Faster R–CNN, the variable-length problem in the bounding boxes is avoided using fixed-size bounding boxes called anchors, which are placed uniformly throughout the image. RPN predicts the probability that an anchor is an object, and a bounding-box regressor adjusts the anchors to fit the object more effectively. After obtaining the relevant objects and their locations, the pooling layer was used to create fixed-size feature maps that correspond to each anchor. Finally, the fully connected layer uses this information to classify the objects and predict the bounding boxes of the identified objects.

In this model, object detection is accomplished in two stages: RPN and a classification network. The VGG-16 feature extractor was used in traditional Faster R–CNN and produces

Table 1 – Images generated by data augmentation methods.

Classes	Original image	Brightness adjustment	Colour transformations (Hue, saturation, and exposure)	Blur	Rotation	Horizontal flip	Total
Number of tender coconuts	1000	3000	3000	1000	1000	1000	10,000
Number of mature coconuts	1000	3000	3000	1000	1000	1000	10,000

better performance with good speed and accuracy when compared with other object detection methods (Huang et al., 2017).

4.2. *Proposed algorithm: Faster R–CNN with ResNet-50*

When compared with traditional object detection methods, the deep convolutional neural network offers significant advancement. (Simonyan & Zisserman, 2015) explained that by increasing the depth of the convolutional network, the accuracy of recognition showed considerable improvement. However, a simple stacked convolutional layer could not train the network smoothly owing to the gradient explosion during backward propagation. The accuracy became saturated and then degraded rapidly. This resulted in precision degradation owing to a greater number of training errors. To solve problems in deep neural networks such as precision degradation and network depth limits, (He et al., 2016) proposed a deep residual ResNet model by using identity mapping concept. This method solved the above issues by selecting a suitable residual map with a multilayer network. A building block of the residual learning module is shown in Fig. 4 with two weight layers.

Finally, a Faster R–CNN with ResNet-50 model is proposed in this paper to enhance the performance of coconut detection in two major maturation stages such as coconut and mature coconut. The improved Faster R–CNN model with a ResNet-50 feature extraction network is shown in Fig. 5.

A flowchart for the proposed coconut detection method is shown in Fig. 6.

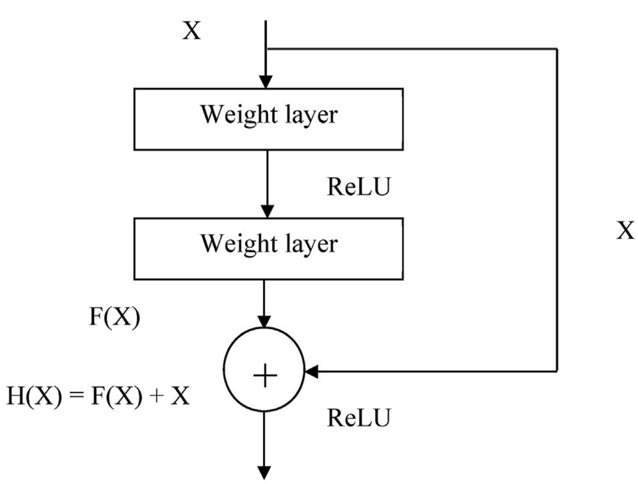

Fig. 4 – Building blocks of residual learning module. The residual map concept of Faster R-CNN is shown in Figure 4 with two weight layers.

5. Experimental results

In this section, the performance of the proposed method is evaluated and analysed. The recognition rate in this proposed method was good, with fewer images used for training and testing with variations in colour, size, shape, texture, illumination, and brightness. The proposed algorithm detected the two maturity stages as tender coconuts and mature coconut bunches in images using the Tensorflow Object detection algorithm and achieved a detection score of 99%. The detection models were trained and tested on Google Colab with Tesla K80 graphics processing unit (GPU). Parameters such as the momentum, initial learning rate, and weight were referred to in the original parameters of the Faster R–CNN model.

A series of experiments were performed with the Faster R–CNN model with the test images to verify the performance of the model with the collected image dataset and Google Images at different resolutions. Figure 7 presents the detection of different types of coconuts from the RGB image using the proposed algorithm and a sample output image is illustrated in Fig. 8.

The proposed coconut detection algorithm is also suitable for detecting coconuts from publicly available images such as those collected by the Google search engine. Figure 9 presents the output of coconut detection from Google Images.

From these experimental results, it can be observed that the coconut detection algorithm can detect coconuts from real-world environments (such as coconut farms), as well as from Google images. Nevertheless, there are some misidentifications in the images regarding coconuts with different maturity levels and different-coloured coconuts in real-time. These issues will be resolved in the future with the addition of more images in the database to improve learning with this vision-based technique to meet real-world challenges and to also make this model more suitable for all varieties of coconuts.

6. Performance evaluation analysis

The training and testing for the detection of coconut maturity stages such as coconuts and mature coconuts of this work is performed on the laptop with the configurations of Intel Core i7-CPU@ 4.5 GHz, 4GB GPU NVIDIA GeForce GTX 1650 with 16GB RAM (Random Access Memory), and Google Colab. Programs were written in Python with OPENCV and CUDA base on Microsoft Windows 10 operating system.

The performance of the object detection algorithm was evaluated by the well-used IoU metric and by average

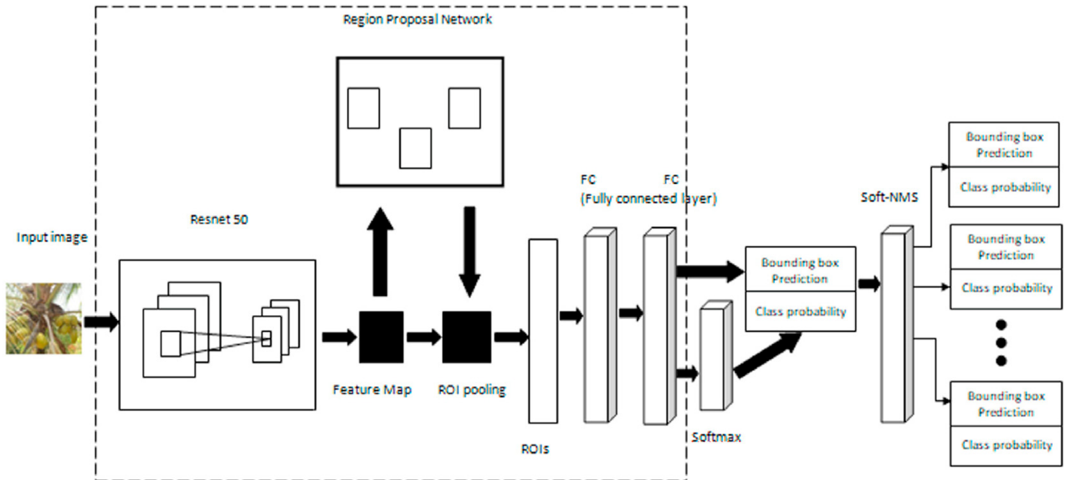

Fig. 5 – Improved Faster R-CNN model with ResNet-50 feature extraction network. Faster R-CNN with ResNet-50 feature extraction network is proposed to detect coconuts with two maturation stages

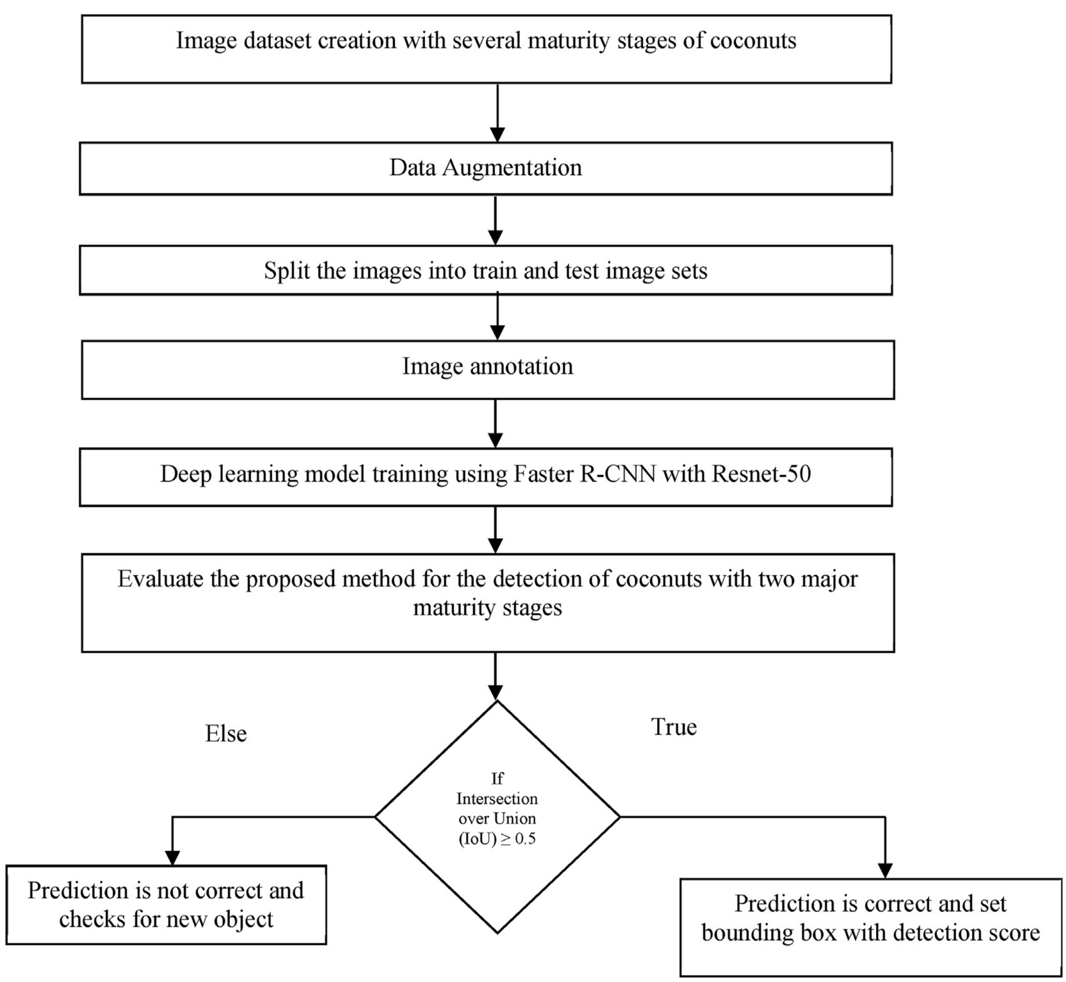

Fig. 6 – Flowchart for proposed coconut detection. The workflow of the coconut maturity stages detection is illustrated in Figure 6.

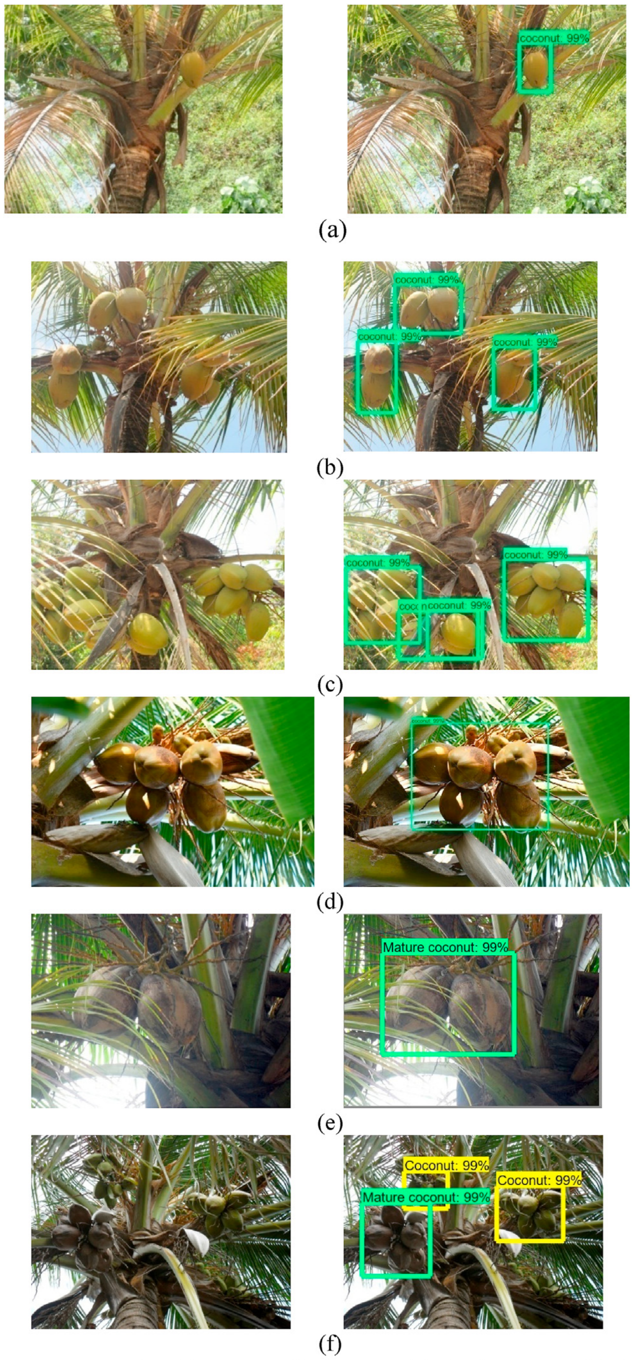

Fig. 7 – Detection of coconuts in different conditions: (a) with single and (b) cluster of coconuts (b) and (c) overlap and occlusion conditions (d) illumination variation (e) and (f) Detection and classification of coconut maturity stages. Coconuts are detected in different conditions in real-time environment.

precision (AP), and mean average precision (mAP). The performance results obtained using IoU and mAP for coconut detection are as follows:

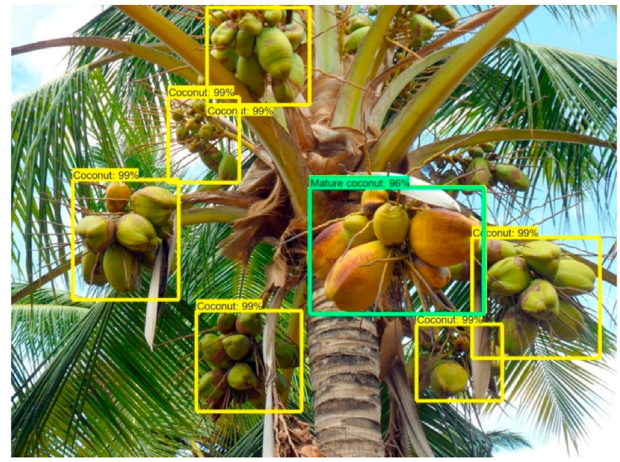

Fig. 8 – Detection of two maturity stages of coconuts using proposed coconut detection method. The proposed method successfully detect two maturity stages of coconuts in an image.

$$AP = \frac{1}{11} \sum_{Recall\_i} Precision(Recall\_i)$$

where $Recall\_i = [0, 0.1, 0.2, \ldots, 1.0]$

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union}$$

where Area of Overlap is the area of intersection of the predicted bounding box and the true bounding box. Area of Union occurs between the two bounding boxes.

The coconut images with two major maturity stages is tested with other feature extraction networks such as ResNet 101, ResNet 152, Inception V2, Inception ResNet V2 and NASNet to compare the real-time performance of Faster R–CNN with ResNet 50. The detection results proved that the performance of our proposed method is better when compared with other feature extraction networks. The detection results for Faster R–CNN with other feature extractors are illustrated in Fig. 10.

The mAP value is calculated by taking the mean AP over the coconut and mature coconut classes and overall IoU thresholds. The resultant values are listed in Table 2.

Based on the performance measure values, Faster R–CNN with the ResNet-50 the feature extraction network performed more effectively in the detection of different maturation stages of coconuts during testing. The models such as Faster R–CNN with Inception Resnet V2 and Nasnet performed well with good mAP values. Meanwhile, the detection speed was also better in Faster R–CNN with ResNet-50. The training and testing of the images was done through the CPU and GPU separately. During the evaluation process, there were misclassifications in some images owing to variations from scales, occlusion, and blurred objects. This issue will be resolved by adding more images with different background

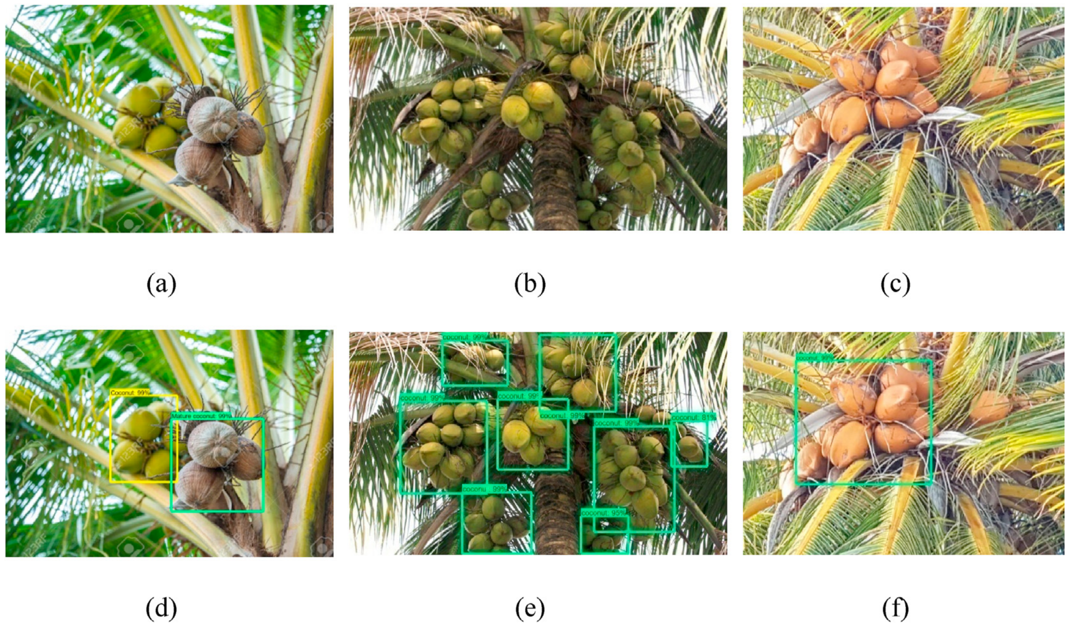

Fig. 9 – Visualisation results for detection of coconuts from Google images (a) with different maturity stages (b) bunch of coconuts in similar and complex backgrounds (c) detection of red coconut variety and (d), (e) and (f) detection, classification and labelling process using proposed method. Google Images are used to test the performance accuracy of the proposed coconut detection method.

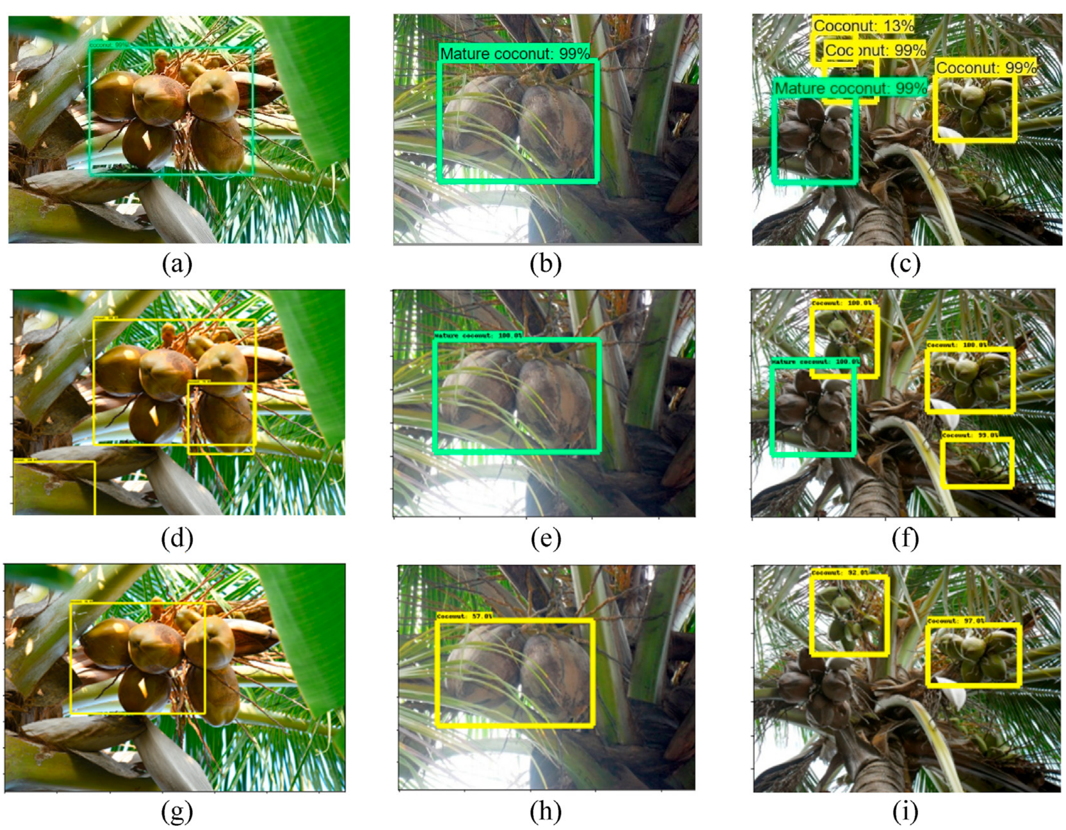

Fig. 10 – Detection results for major maturity stages of coconuts using different feature extraction networks with Faster R-CNN: (a–c) ResNet 50, (d–f) ResNet 101, (g–i) ResNet 152, (j-l) Inception V2, (m-o) Inception ResNet V2, and (p-r) NASNet . Comparison of detection results of Faster R-CNN with ResNet-50 feature extraction network with other feature extractors.

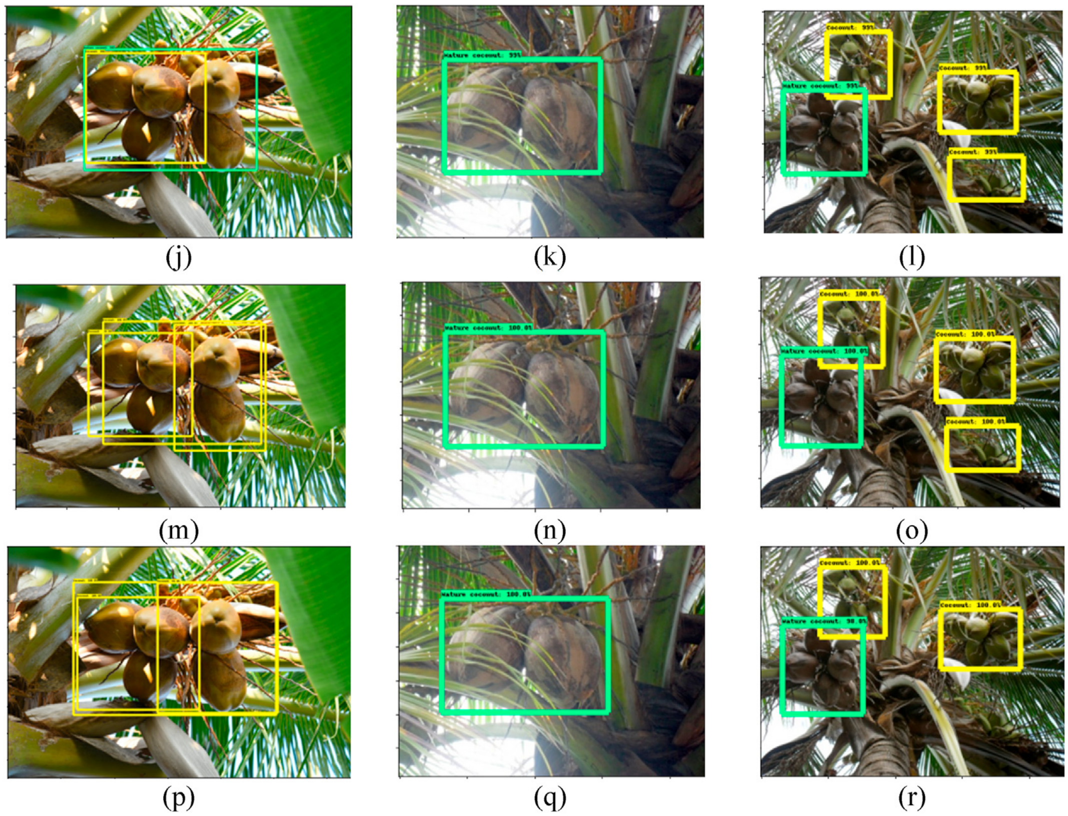

Fig. 10 – (*continued*).

Table 2 – Performance measure values for Faster R–CNN.

Model	Backbone	mAP@0.5IoU	Detection speed (s image^{-1})
Faster R –CNN	ResNet-50	0.894	3.124
	ResNet-101	0.823	3.969
	ResNet-152	0.810	4.132
	Inception V2	0.836	4.248
	Inception Resnet V2	0.855	10.357
	Nasnet	0.862	9.199

conditions for training to achieve better detection results during real-time operation.

The evaluation process of the ResNet-50 network for the classification loss, localization loss, RPN loss/localization loss, and objectness loss is presented in Fig. 11 and Table 3.

6.1. *Comparison of different algorithms*

Different maturation stages of coconuts were used in the training set to verify the performance of the proposed model. The Faster R–CNN model performance was compared with SSD, YOLO V3 and R–FCN to check the advantages of the proposed algorithm. The performance comparison of mAP values for the Faster R–CNN, SSD, YOLO V3 and R–FCN models are presented in Table 4.

The detection results for two major maturity stages of coconuts using Faster R–CNN, SSD, YOLO V3 and R–FCN models are shown in Fig. 12.

The YOLO V3 model was able to detect a single coconut in a complex background with less detection score, but it was unable to identify clustered coconuts that are positioned in various locations adequately in a complex image. Therefore, the YOLO V3 model has less ability to produce good results for the detection of maturity stages of coconuts in our work in real-time environment.

In the coconut crown, occlusions of other objects such as leaflets and leafstalks, as well as other coconut bunches in the tree crown, commonly occur. This can affect the performance of coconut detection. Based on this condition, inaccurate detection results were obtained during different maturation stages. The performance measures for Faster R–CNN with SSD, YOLO V3 and R–FCN are presented in Table 4. From Table 4, it is clear that the detection algorithms can detect the major maturity stages of coconuts in real-time with promising mAP values. The comparison results are given in Table 4 to explore the performance of proposed Faster R–CNN among other object detection algorithms.

The maturity stages of coconuts were tested with the proposed object detection algorithm and it was observed that the results were better when compared with other detection methods. Comparing the test results, the performance of YOLO V3 was low amongst all the other methods. The proposed coconut detection method can accurately detect the maturation stages from similar and complex background objects. The processing speed of YOLO is high when compared with other detection methods but it is unable to detect small and clustered coconuts with different maturity stages in this work. The experimental results indicated that the Faster

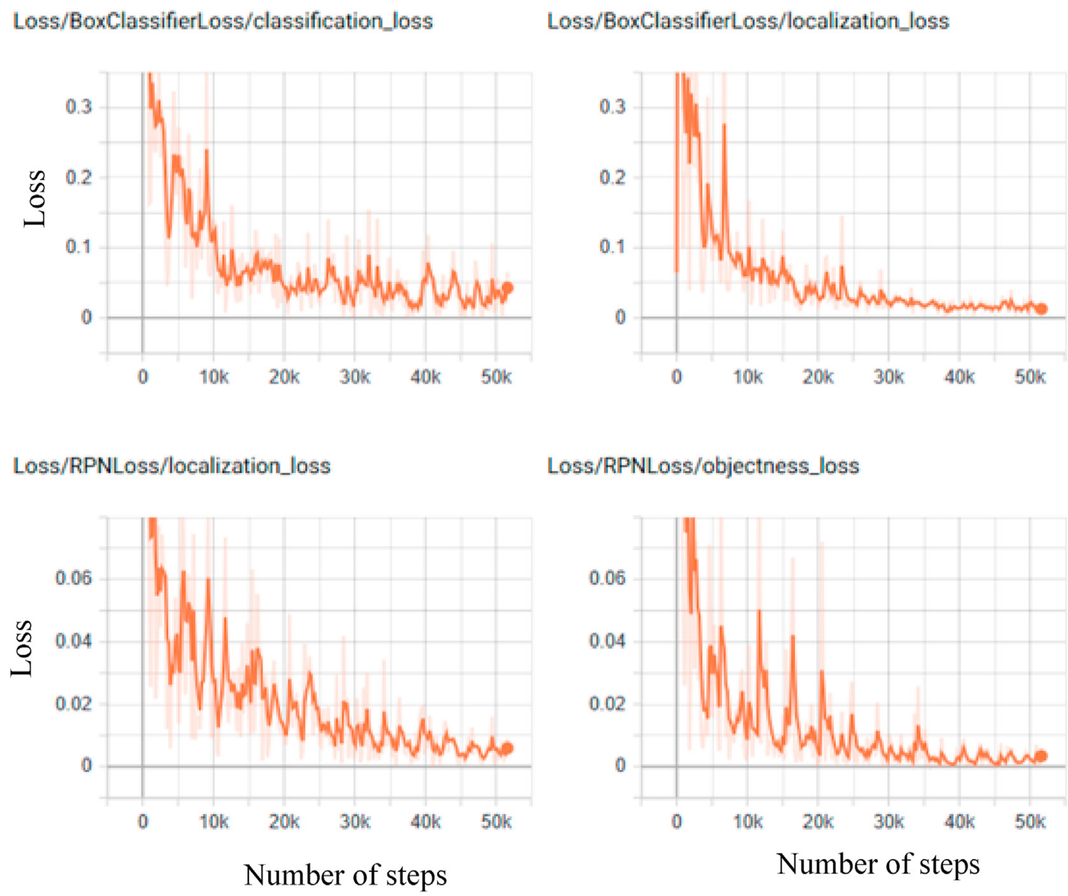

Fig. 11 – Performance evaluation of losses for Faster R-CNN with ResNet-50. The loss values of Faster R-CNN are illustrated in Figure 11 for performance evaluation.

Table 3 – Loss values for improved Faster R–CNN model.

Model	Classification loss	Localisation loss	RPN loss/localisation loss	Objectness loss
Faster R–CNN with ResNet-50	0.0510	0.0252	0.0110	0.0045

Table 4 – Performance results from different object detection algorithms.

Detection algorithm	mAP@0.5IoU	Detection speed (s image^{-1})
Faster RCNN	0.894	3.124
SSD	0.795	2.229
YOLO V3	0.654	1.625
R–FCN	0.768	7.585

R–CNN method is suitable to detect two major maturity stages of coconuts with high detection accuracy and it can be used as the basis for the harvesting of coconuts without human intervention.

During real-time detection, the camera can also capture images of coconut tree crowns that do not have coconuts. To check the real-time challenges at coconut farms, 40 images that did not contain coconuts were collected to test the performance of the proposed algorithm. Among them, 15 images contained leaves, 10 images contained backgrounds, and 15 images had tree crowns without coconuts. The detection method did not find any nuts in these images.

From the experimental results, it is clear that Faster R–CNN with the ResNet-50 model adequately detected the maturity stages of coconuts when compared with the SSD, YOLO V3 and R–FCN models. The results of this experiment further revealed that the Faster R–CNN model was the most suitable for detecting all coconuts in the tree crown particularly when there were small and clustered coconut bunches in a complex background. The SSD model struggled to detect small objects, but it performed better than YOLO with less object detection score. YOLO and R–FCN models struggled to identify two major maturity stages of coconuts that were positioned at various locations in the tree crown and coconuts that were grouped and close together.

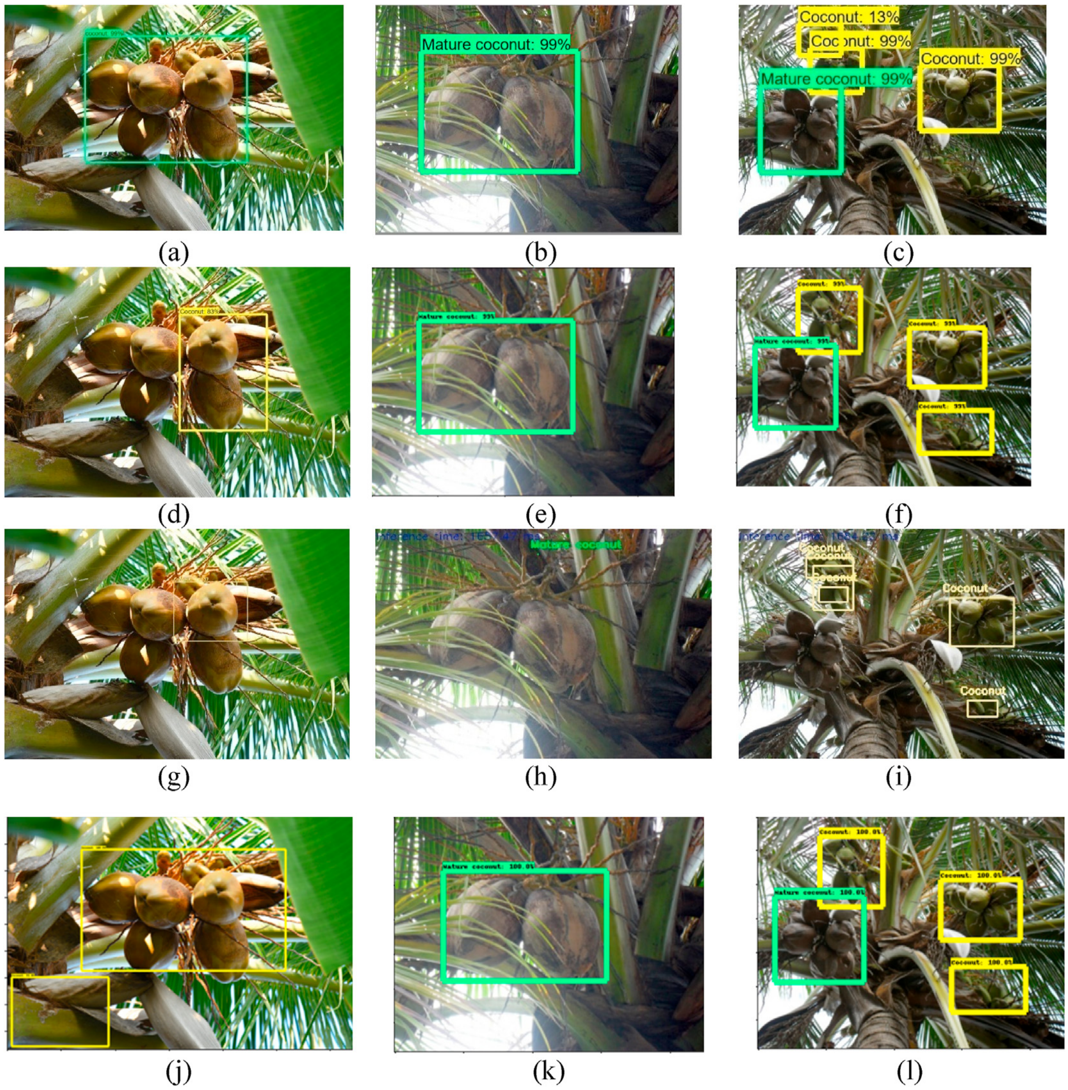

Fig.12 – Detection results for major maturity stages of coconuts using different algorithms: (a–c) Faster R-CNN, (d–f) SSD, (g–i) YOLO V3 and (j-l) R-FCN . Compare the detection results of Faster R-CNN method for the detection of two maturity stages of coconuts among other detection methods.

7. Conclusion

A state-of-the-art fruit detection model with Faster R–CNN was improved by using the ResNet-50 network for detecting coconuts in their two crucial maturation stages; tender for recovering milk and mature to harvest nuts. This proposed model was also able to identify tender coconuts and mature coconuts separately as well as located in the same image. The proposed method efficiently detected coconuts from images with many environmental challenges. Real-world problems were verified with our test dataset by checking the algorithm with real-time images captured in coconut farms as well as from Google Images. In the future work, cutting point of the coconut bunches has to be identified in real-time environment.

Declaration of competing interest

None.

Acknowledgments

This research did not receive any specific grants from funding agencies in the public, commercial, or not-for-profit sectors.

REFERENCES

Amara, J., Bouaziz, B., & Algergawy, A. (2017). A deep learning-based approach for banana leaf diseases classification. Lecture

Notes in Informatics (LNI). *Proceedings - Series of the Gesellschaft Fur Informatik (GI), 266*, 79–88.
Bac, C. W., Van Henten, E. J., Hemming, J., & Edan, Y. (2014). Harvesting robots for high-value crops: State-of-the-art review and challenges ahead. In *Journal of Field Robotics* (pp. 888–911). Wiley Periodicals, Inc. https://doi.org/10.1002/rob.21525.
Bargoti, S., & Underwood, J. (2017). Deep fruit detection in orchards. *Proceedings - IEEE International Conference on Robotics and Automation*, 3626–3633. https://doi.org/10.1109/ICRA.2017.7989417
Behroozi-Khazaei, N., & Maleki, M. R. (2017). A robust algorithm based on color features for grape cluster segmentation. *Computers and Electronics in Agriculture, 142*, 41–49. https://doi.org/10.1016/j.compag.2017.08.025
da Costa, A. Z., Figueroa, H. E. H., & Fracarolli, J. A. (2020). Computer vision based detection of external defects on tomatoes using deep learning. *Biosystems Engineering, 190*, 131–144. https://doi.org/10.1016/j.biosystemseng.2019.12.003
Dias, P. A., Tabb, A., & Medeiros, H. (2018). Apple flower detection using deep convolutional networks. *Computers in Industry, 99*, 17–28. https://doi.org/10.1016/j.compind.2018.03.010
Dyrmann, M., Karstoft, H., & Midtiby, H. S. (2016). Plant species classification using deep convolutional neural network. *Biosystems Engineering, 151*, 72–80. https://doi.org/10.1016/j.biosystemseng.2016.08.024
Ghoury, S., Sungur, C., & Durdu, A. (2019). *Real-time diseases detection of grape and grape leaves using faster R-CNN and SSD MobileNet architectures. International conference on advanced technologies, computer engineering and science (ICATCES 2019), 2019- April* (pp. 39–44). ICATCES 2019.
Grinblat, G. L., Uzal, L. C., Larese, M. G., & Granitto, P. M. (2016). Deep learning for plant identification using vein morphological patterns. *Computers and Electronics in Agriculture, 127*, 418–424. https://doi.org/10.1016/j.compag.2016.07.003
He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Identity mappings in deep residual networks. Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics), 9908 LNCS* (pp. 630–645). Springer.
Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., … Murphy, K. (2017). *Speed/accuracy trade-offs for modern convolutional object detectors. Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, 2017-January* (pp. 3296–3305). IEEE.
Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture, 147*, 70–90. https://doi.org/10.1016/j.compag.2018.02.016
Krizhevsky, B. A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM, 60*(6), 84–90.
Mureşan, H., & Oltean, M. (2018). Fruit recognition from images using deep learning. *Acta Universitatis Sapientiae, Informatica, 10*(1), 26–42. https://doi.org/10.2478/ausi-2018-0002
Niral, V., Rajesh, M. K., & Parthasarathy, V. A. (2017). *Breeding of horticultural crops, 1 -Part B* pp. 445–506). Plantation Crops (2017).
Patrício, D. I., & Rieder, R. (2018). Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. *Computers and Electronics in Agriculture, 153*(April), 69–81. https://doi.org/10.1016/j.compag.2018.08.001
Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE computer society conference on computer vision and pattern recognition, 2016-Decem* (pp. 779–788). https://doi.org/10.1109/CVPR.2016.91
Redmon, J., & Farhadi, A. (2017). YOLO9000: Better, faster, stronger. In *IEEE conference on computer vision and pattern recognition (CVPR2017)* (pp. 6517–6525). https://doi.org/10.1109/CVPR.2017.690. *April*.
Redmon, J., & Farhadi, A. (2018). *YOLOV3: An incremental improvement*. arXiv:1804.02767.
Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 39*(6), 1137–1149. https://doi.org/10.1109/TPAMI.2016.2577031
Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet large scale visual recognition challenge. *International Journal of Computer Vision, 115*(3), 211–252. https://doi.org/10.1007/s11263-015-0816-y
Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T., & McCool, C. (2016). Deep fruits: A fruit detection system using deep neural networks. *Sensors, 16*(8). https://doi.org/10.3390/s16081222
Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations. ICLR 2015 - Conference Track Proceedings*, 1–14.
Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. (2016). *Deep neural networks based recognition of plant diseases by leaf image classification*. Computational Intelligence and Neuroscience, 2016. https://doi.org/10.1155/2016/3289801
Song, Y., Glasbey, C. A., Horgan, G. W., Polder, G., Dieleman, J. A., & van der Heijden, G. W. A. M. (2014). Automatic fruit recognition and counting from multiple images. *Biosystems Engineering, 118*, 203–215. https://doi.org/10.1016/j.biosystemseng.2013.12.008
Tian, Y., Yang, G., Wang, Z., Wang, H., Li, E., & Liang, Z. (2019). Apple detection during different growth stages in orchards using the improved YOLO-V3 model. *Computers and Electronics in Agriculture, 157*, 417–426. https://doi.org/10.1016/j.compag.2019.01.012
Tzounis, A., Katsoulas, N., Bartzanas, T., & Kittas, C. (2017). Internet of Things in agriculture, recent advances and future challenges. *Biosystems Engineering, 164*, 31–48. https://doi.org/10.1016/j.biosystemseng.2017.09.007
Vasconez, J. P., Kantor, G. A., & Auat Cheein, F. A. (2019). Human–robot interaction in agriculture: A survey and current challenges. *Biosystems Engineering, 179*, 35–48. https://doi.org/10.1016/j.biosystemseng.2018.12.005
Wan, S., & Goudos, S. (2020). Faster R-CNN for multi-class fruit detection using a robotic vision system. *Computer Networks, 168*, 107036. https://doi.org/10.1016/j.comnet.2019.107036
Zhao, Y., Gong, L., Huang, Y., & Liu, C. (2016). A review of key techniques of vision-based control for harvesting robot. *Computers and Electronics in Agriculture, 127*, 311–323. https://doi.org/10.1016/j.compag.2016.06.022
Zhu, N., Liu, X., Liu, Z., Hu, K., Wang, Y., Tan, J., Huang, M., Zhu, Q., Ji, X., Jiang, Y., & Guo, Y. (2018). Deep learning for smart agriculture: Concepts, tools, applications, and opportunities. *International Journal of Agricultural and Biological Engineering, 11*(4), 32–44. https://doi.org/10.25165/j.ijabe.20181104.4475

WEB REFERENCES

https://www.123rf.com/photo_12380922_brown-and-green-coconut-at-tree.html. (Accessed 24 November 2020).

https://www.123rf.com/photo_70172826_coconut-tree-with-brown-coconut.html. (Accessed 24 November 2020).
https://www.coolearth.org/2017/07/geneius-coconut-protection/. (Accessed 24 November 2020).
https://www.shutterstock.com/image-photo/green-brown-coconut-on-tree-tropical-1289167426. (Accessed 24 November 2020).
https://ritefmonline.org/ghanas-coconut-industry-identified-as-an-untapped-gold-avenue/. (Accessed 24 November 2020).
https://gardeningkiwi.wordpress.com/20170703_140154/. (Accessed 24 November 2020).