Journal of Engineering Science and Technology
Vol. 18, No. 5 (2023) 2470 - 2477

DURIAN DETECTION AND COUNTING SYSTEM USING DEEP LEARNING

AMIR HILMI AHMAD AZIZI, FAUZUN ABDULLAH ASUHAIMI*, M SAHRIM, IZZUDDIN MAT LAZIM, AZWEEN MOHD ROZMI, WAN ZAKIAH WAN ISMAIL, JULIZA JAMALUDIN, IRNEZA ISMAIL, SHARMA RAO BALAKRISHNAN

[1]Faculty of Engineering and Built Environment,
Universiti Sains Islam Malaysia, Negeri Sembilan, Malaysia
*Corresponding Author: fauzun@usim.edu.my

Abstract

Artificial intelligence (AI) and computer vision (CV) advancements have paved the way for more efficient agricultural activities such as predicting and estimating fruit yield. Durian, a fruit native to tropical regions, necessitated using high-tech solutions to keep up with its rising global demand. This work aimed to apply the image analysis technique using deep learning to identify and estimate the number of durian fruits using image recognition. A new dataset was specifically constructed in this work, consisting of 500 images split for training and testing the object detection model. Various pre-trained object detection models such as YOLOv3, YOLOv4, YOLOv3 tiny, and YOLOv4 tiny are used for performance comparison on the newly constructed dataset. The best model is then chosen as the inference model for the drone-captured video dataset, assisted by the DeepSORT algorithm as the counting mechanism. Our investigations showed that the YOLOv4 model significantly performs best among all four state-of-art detection networks where it computes the highest mean average precision (mAP) performance with 96.02% accuracy on the constructed dataset. This work enables more efficient and precise durian cultivation with less labour and higher-quality yields.

Keywords: Agriculture, Artificial intelligence, Computer vision, Image analysis, Tropical fruit.

1. Introduction

With the rapid development of AI technologies such as Machine Learning (ML) and Deep Learning (DL), the role of computer vision (CV) in agriculture has received increased attention in recent years, which farmers and agronomists have adopted to boost efficiency in the sector [1-3]. The emergence of deep learning technology has offered a practical method for facilitating smart management and decision-making in many agricultural-motivated activities, such as product categorization, real-time plant health monitoring, labour free harvesting and crop growth monitoring [4]. Recent studies have shown that the implementation of AI-based systems in agricultural processes has resulted in significant improvements in crop yield, resource utilization, and operational efficiency[5].

In agricultural technology, apple have been the primary focus for the application of recent computing advancements due to their extensive production and consumption [6]. For instance, Chen et al. [7] proposed a deep learning-based automatic detection and counting mechanism for apple and orange in challenging environments. The study demonstrated the use of a fully convolutional network-based blob detector and the Caffe framework in its fruit detection and counting mechanism. Kuznetsova et al. [8] designed a robot vision system for apple detection using the YOLOv3 algorithm. Further, Liu et al. [9] proposed a method for apple detection method based on the features of shape and colour by utilizing the combination mechanisms of the histogram of oriented gradient (HOG) and simple linear iterative clustering (SLIC). More recently, Kang and Chen [10] employed a one-stage detector with Feature Pyramid Network (FPN) and Atrous Spatial Pyramid Pooling (ASPP) architecture, achieving a recall of 0.821 and an accuracy of 0.853 in real-life apple detection.

Researchers are also applying deep learning techniques in the detection of various other types of fruit and crops. To detect kiwifruits, Liu et al. [11] employed Faster R-CNN and VGG16 network to the input images and reported and an F1-score of 0.88. Neupane et al. [12] have reported a maximum of 96.4% detection accuracy by employing the Faster R-CNN network on custom high-resolution UAV image datasets for the detection and counting of banana trees. In their study on mango detection, Zheng et al. [13] employed a multi-task learning approach combining Faster R-CNN and Mask R-CNN, achieving an Average Precision (AP) of 0.947 at an Intersection over Union (IoU) threshold of 0.75. Zheng et al. [4] and Mureșan and Oltean [14] have presented their respective collections of agriculture-based large scale image datasets related to fruits, vegetables, and plant species classification, taken in various settings.

Although the earlier studies focused on developing fruit detection systems and agricultural image datasets, fewer attempts were made to explore the application of the machine learning technique on tropical fruits, specifically durian. Durian, the king of fruits, is a type of fruit that can be classified as rare and tropical due to its unique, thorny characteristics. Thus, this paper aimed to develop a durian detection and counting system based on YOLO and the Darknet framework trained on a custom-developed image dataset.

2. Methodology

This work aimed to apply the image analysis technique using deep learning to identify and estimate the number of durian fruits using image recognition. Figure 1 showed the flowchart of the process of developing and evaluation the proposed object detection model.

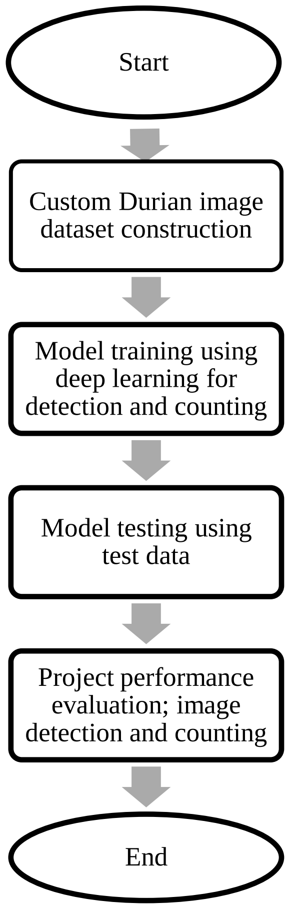

Fig. 1. Flowchart for the methodology in developing durian detection model.

2.1. Durian Dataset Construction

At the beginning of this study, a new dataset that contains a reasonable amount of durian images is constructed. A Google Extension called Download All Images is used to collect durian images on trees as a dataset. Images of durians on trees are retrieved from Google and filtered to include only jpg and jpeg file types to facilitate the later process of data annotation. Downloading several zip files of durian images yielded a total of 500 images for this study. Manual filtering of those images is performed to ensure that all durian images extracted from Google are of high quality. Additionally, prior to training the durian dataset, each image in the dataset was annotated for the durian detection process. The annotation process uses Labellmg graphical annotation tools to create and label durian bounding boxes in each image in the dataset. Figure 2 displayed examples of annotated durian from the constructed image dataset.

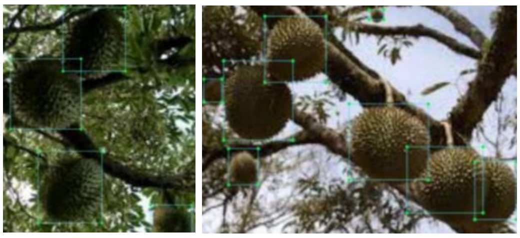

Fig. 2. Sample of annotated durian from the constructed image dataset.

2.2. Custom YOLO object detection model

The model is trained using Google Colab and a cloud-processed Tesla T4 GPU. The Darknet, CUDNN, and OpenCV are used as training frameworks and libraries in the pretrained YOLO models used in this experiment. The training process was started by training the YOLOv3 model with a custom-built dataset of 500 images. The following parameters were used for the training: batch=32, subdivisions=16, learning rate=0.00261, maximum batches=2000, steps=1400,1800 for one class of durian. To investigate the performance of the YOLO detection model training for the durian detection and counting system, some state-of-the-art models of YOLO were run as a comparison with the YOLOv3 model, including YOLOv4, YOLOv3 tiny and YOLOv4 tiny. All the models are trained accordingly for all dataset sizes using the same training parameters. The performance results of training for all four YOLO detection models are compared.

The last part of the deep learning process in this durian detection study is the performance evaluation of the trained model. All the trained YOLO models for this study are evaluated based on their model performance and finally the best model is selected to be implemented in the counting system by applying the Deep Simple Online and Real-time Tracking (DeepSORT) method to keep track of the detected durian counting system.

The networks' performance was evaluated by evaluating the object detection model parameters, which are the precision (Pr), recall (Rc), Fl-score, and mean average precision (mAP), as shown in the Eqs. (1), (2), (3) and (4) respectively.

$$Pr = \frac{TP}{TP+FP} \tag{1}$$

$$Rc = \frac{TP}{TP+FN} \tag{2}$$

$$F1 - score = \frac{2\times Pr\times Rc}{Pr + Rc} \tag{3}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{4}$$

where TP is true positive, FP is false positive, TN is true negative, FN is false negative, Pr is precision, Rc is recall, AP is average precision over all classes, and Mean Average Precision is the average of the AP of each class.

3. Results and Discussion

The aim of this study is to detect durian using good detection model based on deep neural network. To test the model's detection accuracy, a custom durian dataset with 500 images divided into 85% training sets and 15% testing sets was used with four YOLO pretrained models: YOLOv3, YOLOv4, YOLOv3tiny, and YOLOv4tiny. Performance analysis of the AP from IoU based on the output of the training process of all four YOLO models was used to predict the best model to be implemented to the Durian detection system.

Table 1 displayed the trained object detection models and their performance results. The YOLOv4 mode of training performs significantly better than the other three state-of-the-art detection networks, with a mean average precision (mAP) of 96.02% compared to 95.13% for YOLOv4tiny, 95.12% for YOLOv3, and 89.40% for YOLOv3tiny using the same 500 durian image dataset size. Besides, the YOLOv3tiny model has the lowest mAP of 89.40% among the four trained YOLO models.

Table 1. Accuracy result of YOLO models trained with Durian dataset.

Model	TP	FP	FN	Precision	Recall	F1-Score	mAP
YOLOv3	549	22	34	0.96	0.94	0.95	95.12%
YOLOv3tiny	495	47	88	0.91	0.85	0.88	89.40%
YOLOv4	552	35	31	0.94	0.95	0.94	96.02%
YOLOv4tiny	542	29	41	0.95	0.93	0.94	95.13%

The performance results of the evaluated various machine learning models suggested that that the specific architecture of the detection network plays a crucial role in determining the detection accuracy. For instance, the best detection accuracy performance inferred by the YOLOv4 model is attributed to the utilization of CSPDarknet53 as its main backbone network for image features training and extraction [15], significantly improving the detection accuracy over other evaluated models. Further, the YOLOv4 model achieves an improved extracted feature fusion module by designating PANet (Path Aggregation Network) as the neck network while maintaining the YOLOv3's head in its object detection network. On the other hand, the YOLOv4-tiny, YOLOv3, and YOLOv3-tiny architectures utilize inferior network architectures, namely CSPDarknet-53, Darknet-53, and Darknet-19 respectively, which results in a reduction in detection accuracy when compared to the YOLOv4 architecture.

Figure 3 displays the relationship between the loss function and mAP to the number of iterations of the selected model (YOLOv4). From the figure, it can be deduced that as the iteration number increases, the loss function rapidly decreases while the classification accuracy on the test data improves. The network's mAP has scored a high accuracy of 93% in the first 1000 iterations because the dataset has only one class of object, which is durian. Consecutively, the accuracy gradually increased until the 1200 iteration number, reaching 96%, before slowing down to the 1400 iteration number. After 1500 iterations, the network's accuracy ceased to improve, and development fluctuations can be observed thereafter. At this point, it's safe to say that the model has learned completely from the durian dataset and has started to overfit.

Figure 4 shows the snapshot of the video inferencing results using the developed durian detection and counting system based on YOLOv4 and the DeepSORT algorithm. The system has an inference speed of approximately 1.82 FPS. The proposed system was able to detect most of the durians in the sample video. Some durians, on the other hand, are hard to spot because their images are small, and their colours look the same as the green leaves around them.

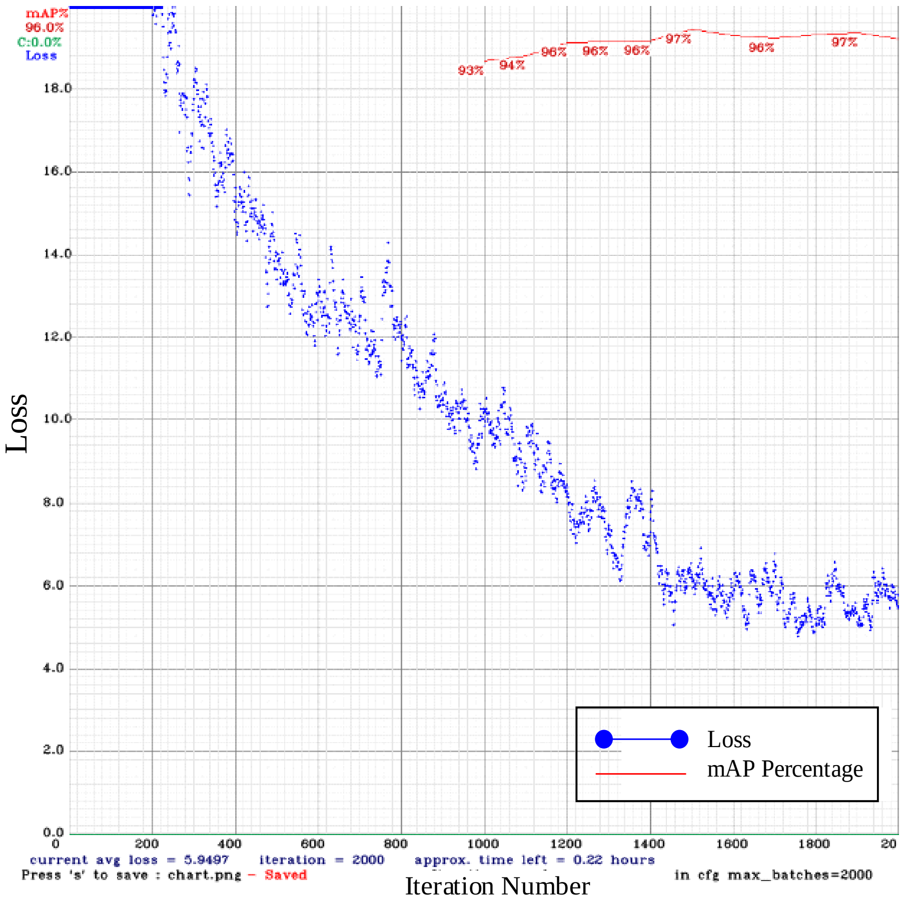

Fig. 3. mAP and loss chart of YOLOv4-based Durian detection model training.

Fig. 4. Snapshot of the video inferencing results of the durian detection and counting system

4. Conclusions

This study has developed and evaluated durian detection and counting system using deep learning approach. The YOLOv4 trained model with the highest mAP value of 96%, trained on a 500-image durian dataset, is determined to be the optimal model in this study. This work could be expanded by implementing the developed system on agricultural drones or robots.

Acknowledgement

This work was supported by grant PPPI/FKAB/0122/USIM/12222 under Universiti Sains Islam Malaysia (USIM). The study was conducted as part of an initiative of Research Excellence Consortium (JPT (BPKI) 1000/016/018/25 (57)), Consortium of Robotics Technology for Search and Rescue Operations (CORTESRO) sourced by Ministry of Higher Education Malaysia (MOHE). The researchers further thank USIM, grant number (43820, 71119), for providing resources in carrying out this study.

References

1. Tian, H.; Wang, T.; Liu, Y.; Qiao, X.; and Li, Y. (2019). Computer vision technology in agricultural automation - A review. *Information Processing in Agriculture*, 7(1), 1-19.
2. Devi, T.G.; Neelamegam, P.; and Sudha, S. (2017). Image processing system for automatic segmentation and yield prediction of fruits using open CV. *Proceedings of the 2017 International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC)*. Mysore, India, 758-762.
3. Bacco, M.; Berton, A.; Ferro, E.; Gennaro, C.; Gotta, A.; Matteoli, S.; Paonessa, F.; Ruggeri, M.; Virone, G.; and Zanella, A. (2018). Smart farming: Opportunities, challenges and technology enablers. *Proceedings of the 2018 IoT Vertical and Topical Summit on Agriculture - Tuscany (IOT Tuscany)*. Tuscany, Italy, 1-6.
4. Zheng, Y.Y.; Kong, J.L.; Jin, X.B.; Wang, X.Y.; Su, T.L.; and Zuo, M. (2019). Cropdeep: The crop vision dataset for deep-learning-based classification and detection in precision agriculture. *Sensors*, 19(5), 1058.
5. Pallathadka, H.; Mustafa, M.; Sanchez, D.T.; Sekhar Sajja, G.; Gour, S.; and Naved, M. (2023). Impact of machine learning on management, healthcare and agriculture. *Materials Today: Proceedings*, 80(Part 3), 2803-2806.
6. Čirjak, D.; Miklečić, I.; Lemić, D.; Kos, T.; and Pajač Živković, I. (2022). Automatic pest monitoring systems in apple production under changing climatic conditions. *Horticulturae*, 8(6), 520.
7. Chen, S.; Skandan, S.; Dcunha, S.; Das, J.; Okon, E.; Qu, C.; Taylor, C.; and Kumar, V. (2017). Counting apples and oranges with deep learning: A data-driven approach. *IEEE Robotics and Automation Letters*, 2(2), 781-788.
8. Kuznetsova, A.; Maleva, T.; and Soloviev, V. (2020). Using YOLOv3 algorithm with pre- and post-processing for apple detection in fruit-harvesting robot. *Agronomy*, 10(7), 1016.
9. Liu, X.; Zhao, D.; Jia, W.; Ji, W.; and Sun, Y. (2019). A detection method for apple fruits based on color and shape features. *IEEE Access*, 7, 67923-67933.

10. Kang, H.; and Chen, C. (2020). Fast implementation of real-time fruit detection in apple orchards using deep learning. *Computers and Electronics in Agriculture*, 168, 105108.
11. Liu, Z.; Wu, J.; Fu, L.; Majeed, Y.; Feng, Y.; Li, R.; and Cui, Y. (2020). Improved kiwifruit detection using pre-trained VGG16 with RGB and NIR information fusion. *IEEE Access*, 8, 2327-2336.
12. Neupane, B.; Horanont, T.; and Hung, N. D. (2019). Deep learning based banana plant detection and counting using high-resolution red-green-blue (RGB) images collected from unmanned aerial vehicle (UAV). *PLOS ONE*, 14(10), e0223906.
13. Zheng, C.; Chen, P.; Pang, J.; Yang, X.; Chen, C.; Tu, S.; and Xue, Y. (2021). A mango picking vision algorithm on instance segmentation and key point detection from RGB images in an open orchard. *Biosystems Engineering*, 206, 32-54.
14. Mureşan, H.; and Oltean, M. (2018). Fruit recognition from images using deep learning. *Acta Universitatis Sapientiae, Informatica*, 10(1), 26-42.
15. Nepal, U.; and Eslamiat, H. (2022). Comparing YOLOv3, YOLOv4 and YOLOv5 for autonomous landing spot detection in faulty UAVs. *Sensors*, 22(2), 464.