1. L.L. YIN[1], 2. M.N.Shah ZAINUDIN[1], 3. W.H.Mohd SAAD[1], 4. N.A. SULAIMAN[1], 5. M.I. IDRIS[1], 6. M.R. KAMARUDIN[1], 7. R. MOHAMED[2], 8. M.S.J.A RAZAK[3]

Centre for Telecommunication, Research and Innovation (CeTRI), Faculty of Electronics and Computer Engineering, Universiti Teknikal Malaysia Melaka, Melaka, Malaysia (1), Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Serdang, Selangor, Malaysia (2), 3MSJ Perwira Enterprise [202103095516 (SA0563088-W)], Duyung, 75460 Melaka, Malaysia (3)
ORCID: 1. 0000-0001-5621-9632

doi:10.15199/48.2023.08.15

Analysis Recognition of Ghost Pepper and Cili-Padi using Mask-RCNN and YOLO

Abstract. *Fruit harvesting robots have made headlines in the agricultural industry in recent years. A fruit recognition system would assist farmers or agricultural industry practitioners in lessening workloads while increasing crop yields. Due to the similar characteristics of chili fruits, approximating the chili according to their grades and identifying its maturity will be difficult. Furthermore, because of their different appearances and sizes, distinguishing between the fruits and the leaves becomes difficult. As a result, a real-time object detection algorithm called You Only Look Once (YOLO) and Mask-RCNN is investigates in order to distinguish the fruit from its plant based on its shape and colour. YOLO version 5 (YOLOv5) uses to define and distinguish the chili fruits and its leaves based on two characteristics; shape and colour. The CSPDarknet network serves as the backbone in YOLOv5, where feature extraction and mosaic augmentation has used to combine multiple images into a single image. Total 391 images has divided into two subsets: training and testing, with an 80:20 ratio. YoLov5 is notable for its ability to detect small objects with high precision in a short amount of time while Mask-RCNN has proven its ability to recognize a chili fruits with high precision above 90%. The classification is evaluated using precision, recall, loss function, and inference time.*

Streszczenie. *Roboty do zbioru owoców trafiły w ostatnich latach na pierwsze strony gazet w branży rolniczej. System rozpoznawania owoców pomógłby rolnikom lub praktykom z branży rolniczej w zmniejszeniu obciążenia pracą przy jednoczesnym zwiększeniu plonów. Ze względu na podobne cechy owoców chili przybliżenie chili według ich klas i określenie stopnia dojrzałości będzie trudne. Ponadto, ze względu na ich różny wygląd i rozmiary, odróżnienie owoców od liści staje się trudne. W rezultacie algorytm wykrywania obiektów w czasie rzeczywistym o nazwie You Only Look Once (YOLO) i Mask-RCNN jest badany w celu odróżnienia owocu od rośliny na podstawie jego kształtu i koloru. YOLO wersja 5 (YOLOv5) służy do definiowania i rozróżniania owoców chili i ich liści w oparciu o dwie cechy; kształt i kolor. Sieć CSPDarknet służy jako szkielet w YOLOv5, w którym wyodrębnianie cech i rozszerzanie mozaiki wykorzystano do łączenia wielu obrazów w jeden obraz. Łącznie 391 obrazów zostało podzielonych na dwa podzbiory: treningowe i testowe, ze stosunkiem 80:20. YoLov5 wyróżnia się zdolnością do wykrywania małych obiektów z dużą precyzją w krótkim czasie, podczas gdy Mask-RCNN udowodnił swoją zdolność rozpoznawania owoców chili z wysoką precyzją powyżej 90%. Klasyfikacja jest oceniana za pomocą precyzji, pamięci, funkcji utraty i czasu wnioskowania.* (***Analiza Rozpoznawanie Ghost Pepper i Cili-Padi przy użyciu Mask-RCNN i YOLO***)

Keywords: YOLO, CSPDarknet, CNN, Mast-RCNN, chili.
Słowa kluczowe: YOLO, przetwarzanie obrazu.

Introduction

Agriculture has always been a noteworthy economic and social sector in any country [15]. Worldwide, agriculture is a $5 trillion industry for its importance of providing food, raw materials and employment opportunities within the social community. As the economy grows, so did agriculture. Since agriculture is evolving, the procedures and techniques are also changing. The previous harvesting system does not have the recognition ability as a human does. A slow harvesting process leads to an inaccurate and inefficient result, while increasing the production cost. Hence, the process of detection and classification of fruits are bound to be different in comparison to the commonly approach. Two essential variables need to be consider when classifying agricultural products; weight and size. Researcher starts to undergo their study by improving the current practice in agriculture industries including the process of farming, fertigation supplies, harvesting and categorizing. Such information is crucial by creating a documentation and management record for farmers to undergo further analysis [16]. The invention of Internet of Things (IoT) has broadly utilized for improving fertigation and pesticide purposes. Somehow, some industries have started to implement an autonomous or semi-autonomous robot for practicing the harvesting process. However, the process of categorizing the fruits remain unchanged. Workers still conduct manual process of classification of fruit's class.

Human food supplies are gradually increasing nowadays. To meet the world's increasing population's food demand, horticulture must find new ways to increase fruit and vegetable production [1]. Fruit harvesting is an essential part of the development and management of farmlands. By using traditional approach, it is difficult for farmer or labor to detect the size of chili for estimating the maturity level. Small mistakes or missteps are usually unavoidable, especially for estimating the maturity of the chili because human eyes are prone to errors and inaccurate. Thus, the use of automated classification of maturity level is take into consideration. This approach represents an innovative, feasible, and economical alternative for farmers who require the accurate size of chili for maturity classification. Traditional manual harvesting necessitates a large number of farmworkers, resulting in a high production cost [2]. In order to address the issue of inefficient manual fruit harvesting, an intelligent and systematic automatic harvesting robot has introduced in recent years. To date, few numbers of related works reported for measuring the quality or categories of the fruits. Most reported works is able to prove its ability to define the fruit's quality types of fruits based on the captured images. Fruits images with small in sizes and having a complex characteristic such chili remain unanswered for further investigations. Deep learning is one of the most common examples of an innovation in agriculture industry. Deep learning is an Artificial Intelligence (AI) subfield that can learn from unstructured or unlabeled data and invented as a foundation of neural network architecture [19]. The use of AI has significantly proven its ability to solve a variety of applications including human activity [13], industries [12], smart home [11], etc.

In this paper, a deep learning method known as Convolutional Neural Network (CNN) has applied to carry out an analysis of fruit recognition system for chili fruits based on two characteristics: shape and color. CNN is a multi-layer neural network that has commonly used for image recognition and object detection. To address a rapid detection process, You Only Look Once (YOLO) algorithm

has introduced, which employs CNN to detect objects in real time. YOLO has demonstrated to be a high precision, fast algorithm for solving a wide range of scene detection tasks [17], as well as in the agriculture industries [3] such as in detection of apple [21-23], tomatoes [24-25] and kiwi fruits [26-27]. YOLOv5 is the fifth generation of YOLO, released in June 2020 as a natural progression from Glenn Jocher's YoLov3 Pytorch repository. YOLOv5 is an improved version of YOLOv4 that incorporates a mosaic augmentation technique to improve performance. The mosaic augmentation technique teaches the model how the object detection model can identify objects at a smaller scale than usual. A region-based fully convolutional network combines with the YOLOv5 algorithm to improve the detection of small targets in remote sensing images [4]. YOLOv5 uses to recognize and detect both fruits and leaves in images of fruit plants. YOLOv5 models include the YOLOv5-s (small version), YOLOv5-m (medium version), YOLOv5-l (large version), and YOLOv5-x (extra-large version) [5].

This article highlights a few contributions. A fruit recognition system is design and implement to recognize chili fruits based on their shape and color. Simultaneously, the collected 2D images has analyzed to determine their maturity category using YOLOv5. In terms of accuracy and efficiency, the analysis also compares the performance of YOLOv5 with the third version of YOLOv3 with different number of epoch. We also compare the performance of YOLOv5 with other version of CNN, Mask R-CNN.

Methodology

Fig. 1 depicts the general methodologies of a YOLOv5-based analysis of chili fruit detection and maturity estimation.

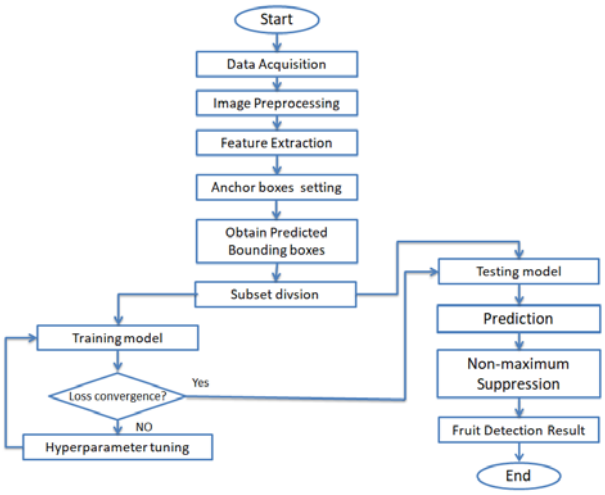

Fig. 1. Flowchart of an analysis of chili fruit detection and maturity estimation using YoLov5.

Data acquisition

Two types of peppers, ghost pepper and cili padi has used as an image for the experiment in this work. Under varying light intensity, the chili image is captured from various characteristics such as viewing direction, different colour (green, red, brownish), and shape. A 2D camera has used to capture 201 chili images. When pursuing this data augmentation process, factors such as backlighting, front lighting, long range, close range, and random occlusion must consider.

Image pre-processing

Image pre-processing is required to remove unwanted information from raw images or to clean them for better representation and quality. An image is standardized, unwanted noise is removed, and data augmentation and filtering are been carried out. The median filtering method uses to remove noise from images. The image then resized to a single dimension before it is use as input for the learning algorithm. The entire set of images has resized to 416 X 416 pixels. Data augmentation techniques such as scaling, rotation and other affine transformations has used to expand the dataset and expose the learning algorithm to a variety of image changes. As a result, 391 images has generated to evaluate the learning model's ability to detect an object in a variety of shapes and sizes. Fig. 2 depicts the data augmentation process used on a chili image.

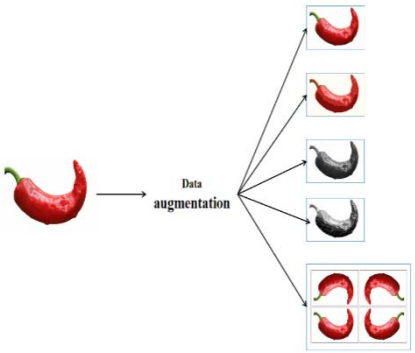

Fig. 2. Data augmentation process.

Feature extraction

As shown in Fig. 3, YOLOv5 network divides into three sections: the backbone, the neck, and the head. YOLOv5 built CSPDarknet as the Darknet's backbone by incorporating a cross stage partial network (CSPNet). The data fed into CSPDarknet for feature extraction before been fused into a path aggregation network (PANet). YOLO layer defines the detection results as an output.

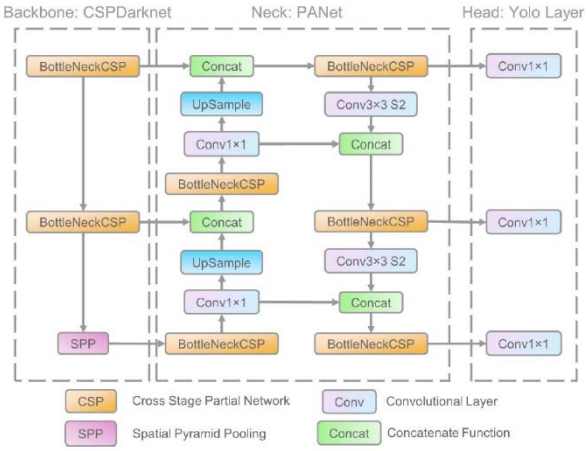

Fig. 3. YOLOv5 network architecture [6].

CSPNet uses to solve the problem of repeated gradient information in large-scale backbones by integrating gradient changes into convolution layers, reducing model parameters, and increasing FLOPS (floating-point operations per second). This not only improves inference speed and accuracy, but it also has the potential to reduce model size. YOLOv5 used PANet as a bottleneck to increase data throughput. To improve low-level feature propagation, PANet employs a new feature pyramid network (FPN) topology with an improved bottom-up approach. Furthermore, PANet improves the use of precise localization signals in lower layers, which can improve object location accuracy significantly. YOLO layer generates three different sizes of convolution layers to enable multi-scale prediction (18 X 18, 36 X 36, and 72 X 72). YOLO layer allows YOLOv5 to handle tiny, medium and large objects [6].

Anchor box

Anchor boxes, as shown in Fig. 4, are boundary boxes that represent the height and width. Learning anchor boxes based on analysing the distribution of bounding boxes in a dataset using K-means and genetic learning algorithms. Because the distribution of bounding box size and locations in the COCO dataset may differ significantly from the predefined bounding box anchors, this is critical for custom tasks. YOLOv5 automatically learns all YOLO anchor boxes when custom data is entered [7]. With their centres in the small cell, anchor boxes uses in fruit detection to detect and recognize a variety of objects.

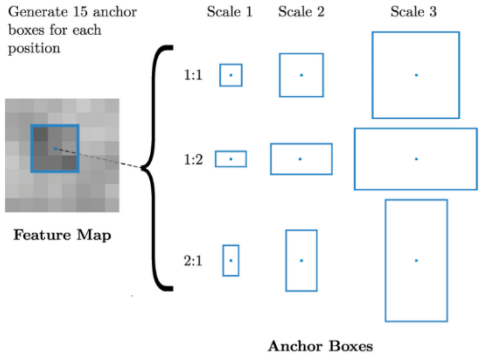

Fig. 4. Anchor boxes [7].

Bounding box prediction

As shown in Fig. 5, the YOLOv5 network predicts four coordinates for each generated bounding box: t_x, t_y, t_w, and t_h. If the cell is offset by (c_x, c_y) from the image's top left corner and the bounding box prior has width and height p_w, p_h, then the following predictions are made:

(1) $$b_x = \sigma(x) + c_x$$

(2) $$by = \sigma(t_y) + c_y$$

(3) $$b_w = p_w e$$

(4) $$b_h = p_h e$$

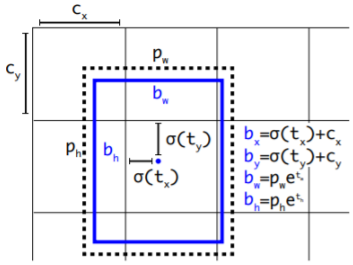

Fig. 5. Bounding boxes with dimension priors and location prediction [8].

YOLOv5 predicts an object score for each bounding box using logistic regression. This should be 1 if the bounding box prior overlaps a ground truth object by more than any other bounding box prior. The prediction is ignore if the bounding box prior is not the best but overlaps a ground truth object by more than a certain threshold.

Subset division

The dataset randomly divided into two subset groups: training and testing. In this experiment, 80% of the sample has chosen at random for training, while the remaining 20% has used to assess the capability of the trained model. In this experiment, various image categories such as ripe chili, unripe chili, and different shape and size chili are used.

Intersection over Union (IoU) and loss function

During the fruit detection process, anchor box is create and match. To estimate and determine the size of anchor boxes, K-means clustering has applied. K-means clustering is a clustering method that divides a training set into clusters of instances that are similar to one another. In these cases, this method uses to compute the object's centroids. The highest overlap centroids are divides by the non-overlapping centroids, and this process is repeats for each anchor box. That process is known as IoU. As shown in Eq. 5, IoU is a standard for identifying object precision that is used to compare the predicted bounding box to the actual bounding box. IoU is a normalised index with the values [0,1][9]. The object is detected if the IoU is greater than 50%. Otherwise, if the IoU is less than 40%, no object is detected.

(5) $$\text{IoU} = \frac{area\,(box(Predicted) \cap box\,(Truth))}{area\,(box\,(Predicted) \cup box\,(Truth))}$$

In addition, loss function will measure to determine the value of a set of parameters. The difference between the network output and the actual output will compare by the loss function. This function is uses to improve positioning while measuring the accuracy of an object. As shown in Eq. 6, the loss function in YOLOv5 is measure in three categories: bounding box positioning error, confidence error, and classification error. The L_{box} represents the bounding box positioning error, L_{cls} represents the confidence error, and L_{obj} represents the classification error. The bounding box positioning error is calculate when there is a difference between the predicted bounding box coordinates of the anchor boxes and the actual coordinates. Furthermore, the cross-entropy uses to calculate the confidence error, which reflects the probability that the target frame contains the target. The classification error is calculated when the bounding box detects the target in the current box. Furthermore, the precision, recall, and mean average precision (mAP) will measured, as shown in Eqs. 7-9.

(6) $$Loss = Lbox + Lcls$$

(7) $$p = \frac{Tp}{Tp + Fp}$$

(8($$R = \frac{Tp}{Tp + FN}$$

(9) $$mAP = ip(k) \triangle R(k)$$

where: C – number of categories, N – number of IoU thresholds, k – mIoU threshold, p(k) – precision, R(k) – recall.

Hyper-parameter tuning

Hyper-parameter tuning refers to the process of fine-tuning an optimum value of those required parameters in order to ensure the algorithm runs smoothly and produces excellent results. Three parameters must be define in the training process for this experiment: input image size, learning rate, and number of epochs. Typically, hyper-parameter tuning will accomplish in two ways. The first step is to conduct a random search, followed by a small-sized range grid search for final tuning.

Non-maximum suppression (NMS)

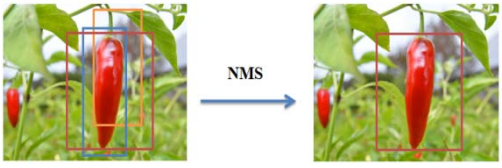

Fig. 6. The convergence curve of the total loss functions for different YOLOv5 models.

The next step in object detection is Non-Maximum Suppression (NMS), which uses to select the best bounding box for the object as shown in Fig. 6. In YoLov5, NMS chooses the best bounding box to define an object based on the value IoU greater than 50%. When the IoU value is greater than 50%, NMS will remove all bounding boxes.

Experimental analysis and discussion

YOLOv5 is available in four versions, as mentioned in the previous section: YOLOv5-s (small), YOLOv5-m (medium), YOLOv5-l (large), and YOLOv5-x (extra-large) (extra-large version). The optimizer stochastic gradient descent (SGD) uses as optimization. The batch size is 8, and the number of iteration epochs for training YOLOv5-s, YOLOv5-m, and YOLOv5-l is set to 400. Due to resource constraints, this experiment makes use of an existing server Graphical Processing Unit (GPU) in Google Colab. As a result, due to processor limitations and algorithm complexity, YOLOv5x employs a small number of epochs. As a result, 200 epochs are define to train the YOLOv5x. Total 331 of the 391 images used for training, with the remainder reserved for testing. The experiment will measured in terms of loss and mAP. This indicator is useful in object detection for determining the quality and accuracy of the model. Fig. 7 depicts the model-specific total loss function.

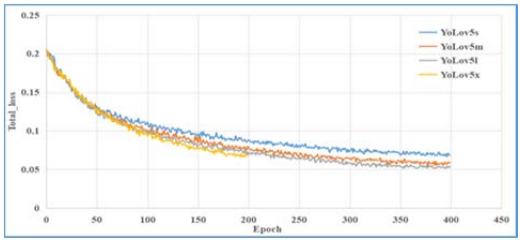

Fig. 7. The convergence curve of the total loss functions for different YOLOv5 models.

As the number of iterations increases and the loss decreases, the YOLOv5 algorithm curve gradually converges. YOLOv5-s converges and records the slowest and largest loss, followed by YOLOv5-m and YOLOv5-l. YOLOv5-x, on the other hand, has the fastest convergence and the smallest loss value. Because YOLOv5-x has a slow calculation speed, thought to be unsuitable for real-time applications. As a result, despite producing better results than other YOLOv5 models, YOLOv5-x has largely abandoned. YOLOv5-l's loss convergence is rapid and has the smallest loss value. As a result, YOLOv5-1 considered suitable for use in solving object detection problems with low processing requirements.

Comparison with various YOLO models

*YOLO*v5-s, *YOLO*v5-m, and *YOLO*v5-l applied for further analysis due to the limitations mentioned in the previous section. As a result, *YOLO*v5-x has excluded from the following experiment. Several evaluation metrics, such as precision, recall, and mAP, has applied to evaluate the experimental analysis. Table 1 summarizes the performance results for various YOLOv5 model versions.

Table 1. The performance results on different version of YOLOv5 detection models.

Models	Precision (%)	Recall (%)	mAP_0.5 (%)
YOLOv5-s	73.5	63.6	66.6
YOLOv5-m	75.5	65.8	67.0
YOLOv5-l	**78.2**	**67.8**	**69.2**

YOLOv5-l outperformed YOLOv5-s and YOLOv5-m in terms of precision, recall, and mAP. YOLOv5-l had 4.7% and 2.7% higher precision than YOLOv5-s and YOLOv5-m, respectively. YOLOv5-l has a recall of 67.8 percent, which is 4.2% and 2% higher than YOLOv5-s and YOLOv5-m, respectively. Furthermore, the mAP of YOLOv5-l is 2.6% and 2.2% higher than that of YOLOv5-s and YOLOv5-m, respectively. Overall, YOLOv5-l has reported the best performance for object detection and the highest accuracy when compared to YOLOv5-s and YOLOv5-m.

The integration of YOLOv5 and CNN will tested for object detection across an entire image. After that, the process divides into parts and predicts the bounding box and probability of each component. The predicted probability uses to calculate the weight of each bounding box. YOLOv5 makes predictions after passing through the neural network once in the forward direction. Following the completion of NMS, the detected objects has passed. Fig. 8 and 9 depict the outcome of inference on images divided into two categories: ripe chilies and unripe chilies. As shown in Fig. 8, the algorithm detects and recognizes ripe chili with greater than 93% accuracy. Above 80% accuracy has been achieved in detecting unripe chili and differentiating those two categories using different bounding boxes.

Fig. 8. Detection results of mature chili.

Fig. 9. Detection results of immature chili.

Hyper-parameter analysis

An epoch is a period of time during which each sample in the training dataset has the chance to update the internal model parameters. Each epoch consists of one or more batches. The number of epochs is usually large, ranging from hundreds to thousands, to allow the learning algorithm to run until the model's error sufficiently minimized. The most outstanding performance has recorded by YOLOv5-l, according to this work. (78.2% of precision). The number of training images used in this section has reduced to 100 due to a lack of GPU usage and allocation period. The experiment with a large number of images has tested, but due to the limitations mentioned, the training stops before the maximum number of epochs reached. Fig. 10–12 depict the convergence curve of the total loss function for various number of epochs during YOLOv5-l model training.

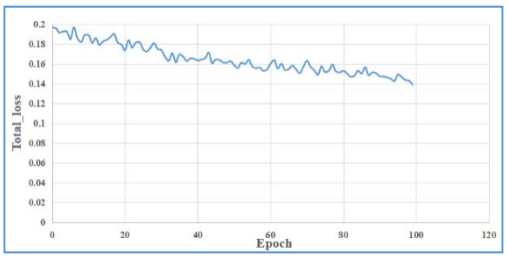

Fig. 10. The convergence curve of the total loss function for 100 epoch.

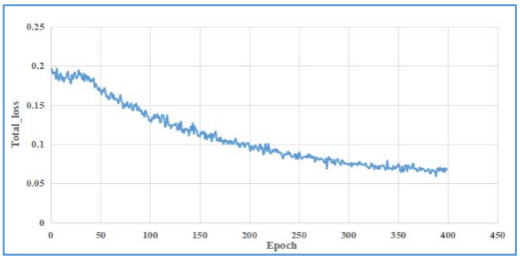

Fig. 11. The convergence curve of the total loss function for 400 epoch.

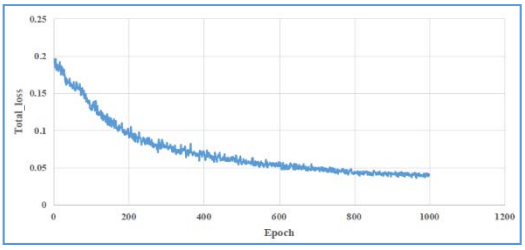

Fig. 12. The convergence curve of the total loss function for 1000 epoch.

In this experiment, the number of epochs for the YOLOv5-l model is set to 100, 400, and 1000. Fig. 10 depicts the overall result of the loss function obtained in YOLOv5-l after 100 epochs. Fig. 11 and 12 depict the loss over 400 and 1000 epochs, respectively. It is clear that when a small number of epochs (100) is define, it results in a high loss. Meanwhile, using a larger number of epochs reduces the loss slightly. When the model is iterate 800 times, the loss becomes more stable and approaches zero, as shown in Fig. 12. To summarize, the number of epochs chosen significantly relate to the value of the loss. Table 2 displays the precision and mAP obtained in the YOLOv5-l model for various epoch numbers. The model with 1000 epochs received the highest precision and mAP with 88.5% and 59.2% respectively. The precision was 39.9% and 20.8% higher than the training using 100 and 400 epochs respectively. While the smallest number of epoch used resulted in a 26.4% increase in mAP and a 6.7 percent increase in mAP.

Table 2. The Precision and mAP result for different number of epoch.

No. of Epoch	Precision (%)	mAP_0.5 (%)
100	48.6	32.8
400	67.7	52.5
1000	88.5	59.2

Comparison of YOLOv5, YOLOv3 and Mask-RCNN

As previously stated, the YOLO and Mask R-CNN algorithms are notable for their effectiveness in solving any object detection problem. This section discusses the differences between YOLOv5, YOLOv3, and Mask R-CNN. In this section, 331 images trained on YOLOv3 and YOLOv5 with 200 epochs each. Due to the limitation mentioned in the previous section, only 200 images with a maximum of 10 epochs used to train the Mask R-CNN. Analysis of speed, precision, and mAP 0.5 used to evaluate the performance of all models. Table 3 displays the testing time (in milliseconds), precision, and mAP 0.5 for each model.

Table 3. The performance results of different deep learning detection models.

	Testing Time (ms)	Precision (%)	mAP_0.5 (%)
YoLov3	102.4	75.0	65.1
YoLov5	63.6	78.2	69.2
Mask R-CNN	12000.0	95.0	75.0

Mask R-CNN is a Faster R-CNN extension that generates pixel-level masks for each detected object. Because of the advantages of Mask R-CNN, such as segmentation, it appears to be a good opportunity to use this method in this case. In contrast to traditional object detection algorithms, instance segmentation provides more information, allowing the object to localize without the use of a bounding box. Mask R-CNN achieved higher precision and mAP than YOLOv3 and YOLOv5, which achieved 95% precision and 75% mAP, respectively. Precision obtained is approximately 20% and 16.8% higher than YOLOv3 and YOLOv5, respectively; and mAP recorded 9.9 percent and 5.8% higher than YOLOv3 and YOLOv5, respectively. In terms of accuracy, YOLOv5 outperformed YOLOv3 by a significant different. YOLOv5 had a precision and mAP of 78.2%and 69.2%, respectively, which were 3.2% and 4.1% higher than YOLOv3.

Although the trained model Mask R-CNN has the highest precision and mAP score in predicting the objects, it has the longest inference time. Mask R-CNN recorded a total inference time of around 12000ms, making this model impractical for use in real-time as YOLO models. YOLOv3 and YOLOv5 have the fastest inference times, with 102.4ms and 63.6ms, respectively. Furthermore, YOLOv5 takes less time to process a single image in a standard deep learning machine than YOLOv3. In terms of overall performance, it is clear that YOLOv5 outperforms YOLOv3 and Mask R-CNN. As a result, YOLOv5 is regards as superior not only for detecting an object, but also for detecting small objects with complex representations such as chili fruits.

Discussion and conclusion

In this work, images are collects and analyze using various parameters to improve the model's accuracy. A deep learning method has investigated in order to perform automatic classification and detection of chilies in the chili plant. The YOLO model can detect the presence of chilies and distinguish between their fruits and leaves. Furthermore, it can estimate the maturity category of the chili fruit by distinguishing between ripe and unripe fruits. To detect chili fruits, the YOLOv5 model has applied to the recognition system. YOLOv5 can distinguish the category of chili fruit with high accuracy in a short period. The two-dimensional images of the chili fruit captured and stored as a dataset. The images then divided into two subsets: training and testing, with an 80:20 partition. Furthermore, other YOLO versions, such as YOLOv3 and Mask R-CNN, used to compare precision and mAP with YOLOv5. YOLOv5 outperformed YOLOv3 in terms of precision and mAP. It can also make inferences faster than Mask R-CNN. As a result, we believed that the proposed YOLOv5 model could be apply in the development of an autonomous fruit-picking robot.

A few criteria must considered in order for improving the accuracy of chili identification. More datasets are required to better identify the chili with its present is overlapping as an individual entity. Data consists of examples or cases from various domains that describe the problem that needs to solve. In our case, an input data as image made up of deep learning examples, each of which has an input element that will fed into the model and the model is expects to predict that input as its categories. As a result, broadening the dataset with different characteristics is a good way to improve the wide range of the chili identification process [10]. Chili images will captures at different times of the day, in various weather conditions, with various lighting conditions, and from different angles to help expand their characteristics. Furthermore, improving

the dataset quality will considers for improving the performance of the YOLOv5 recognition system. On the other hand, some of the features may be irrelevant in determining the object classes; thus, feature selection thought to aid in improving detection accuracy [14]. Another suggestion is to analyses the performance of the YOLOv5 using images captured by a dual-camera, also known as a stereo vision camera. This is because this camera provides more information about the depth of the feature vector, allowing for more accurate object detection [18]. The distance between the camera and the object could be useful in real-time environment conditions. The time required for model training with a large dataset and a large number of epochs will be longer. It is suggest that a GPU with more memory will uses to reduce the time required to train the model. Multi label classification, on the other hand, is also can be apply to define the maturity and category of chili fruits. As a result, the proposed algorithm can classify and recognize the maturity of chili fruits as well as the class of chili fruits as green, red, or brownish using multi label classification problems where the algorithm is able to classify various types of classes at the same time in parallel [11]. AI or machine learning has also be used to investigate or diagnose faults in machines such as fruit picking robots [12].

Acknowledgement

This work funded by the Centre for Research and Innovation Management (CRIM), Universiti Teknikal Malaysia Melaka (UTeM).

***Authors**: L.L.Yin, Email: m022010040@student.utem.edu.my; M.N.Shah Zainudin, Email: noorazlan@utem.edu.my; W.H.Mohd Saad, Email:wira_yugi@utem.edu.my; N.A.Sulaiman, Email: noor_asyikin@utem.edu.my; M.I.Idris, Email: idzdihar@utem.edu.my ; M.R.Kamarudin, Email: raihaan@utem.edu.my. Faculty of Electronics and Computer Engineering, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100, Durian Tunggal, Melaka, Malaysia; R.Mohamed, Email: raihanimohamed@upm.edu.my. Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, UPM Serdang, 43400, Seri Kembangan, Selangor, Malaysia ; M.S.J.A Razak, Email: solokfertigasi@gmail.com . 3MSJ Perwira Enterprise [202103095516 (SA0563088-W)], Duyung, 75450 Melaka, Malaysia.*

REFERENCES

[1] J. Gené-mola, V. Vilaplana, J. R. Rosell-polo, J. Morros, J. Ruiz-hidalgo, and E. Gregorio, Multi-modal deep learning for Fuji apple detection using RGB-D cameras and their radiometric capabilities, *Computers and Electronics in Agriculture,* (2019), vol. 162, pp. 689-698.

[2] G. Wu, Q. Zhu, M. Huang, Y. Guo, and J. Qin, Automatic recognition of juicy peaches on trees based on 3D contour features and colour data, *Biosystems Engineering,* (2019), vol. 188, pp. 1-13.

[3] A. Kuznetsova, T. Maleva, and V. Soloviev, Detecting Apples in Orchards Using YOLOv3 and YOLOv5 in General and Close-Up Images, *LNCS*. (2020), vol. 12557, pp. 233-243.

[4] W. Wu, H. Liu, L. Li, Y. Long, X. Wang, Z. Wang, J. Li and Y. Chang, Application of local fully Convolutional Neural Network combined with YOLO v5 algorithm in small target detection of remote sensing image, *PLoS One*, (2021), vol. 16, no. 10.

[5] F. Zhou, H. Zhao, and Z. Nie, Safety Helmet Detection Based on YOLOv5, *Proc. 2021 IEEE Int. Conference on Power Electronics, Computer Applications* (2021).

[6]R. Xu, H. Lin, K. Lu, L. Cao, and Y. Liu, A forest fire detection system based on ensemble learning, *Forests*, (2021), vol. 12, no. 2.

[7]M. Ferguson, R. Ak, Y.T.T. Lee, and K.H. Law, Detection and segmentation of manufacturing defects with convolutional neural networks and transfer learning, *Smart Sustain. Manuf. Syst.*, (2018), vol. 2, no. 1.

[8]J. Redmon, and A. Farhadi, *YOLO v.3*, Tech Report (2018).

[9]Q. Song, S. Li, Q. Bai, J. Yang, X. Zhang, Z. Li, and Z. Duan, Object detection method for grasping robot based on improved yolov5, *Micromachines*, (2021), vol. 12, no. 11.

[10] M.N.S. Zainudin, N. Hussin, W.H.M. Saad, S.M. Radzi, Z.M. Noh, N.A. Sulaiman, and M.S.J.A.Razak, A framework for chili fruits maturity estimation using deep learning convolutional neural network, *Przeglad Elektrotechniczny*, (2021), vol. 11, no. 2021.

[11] R. Mohamed, T. Perumal, M.N. Sulaiman, N. Mustapha, and M.N.S. Zainudin, Modeling activity recognition of multi resident using label combination of multi label classification in smart home, *International Conference on Applied Science and Technology*, (2017).

[12] N.A. Sulaiman, M.P. Abdullah, H. Abdullah, M.N.S. Zainudin, and A.M. Yusop, Fault detection for air conditioning system using machine learning, *IAES International Journal of Artificial Intelligence*, (2020), vol. 9, no. 1.

[13] Y.J. Kee, M.N.S. Zainudin, M.I. Idris, R.H. Ramlee, and M.R. Kamarudin, Activity Recognition on Subject Independent Using Machine Learning, *Cybernatics and Information Technologies*, (2020), vol. 20, no. 3.

[14] B. Venkatesh and J. Anuradha, A Review of Feature Selection and its Methods, *Cybernatics and Information Technologies*, (2019), vol. 19 no. 1.

[15]M. L. Praburaj, Role of Agriculture in the Economic Development of a Country, *Int. J. Commer.* (2018) vol. 6, no. 3, pp. 2.

[16]M. Mraz, P. Findura, O. Urbanovicoba, I. rigo, P. Bajus, T. Drozdz and P. Keilbasa, Development of the web application by the information system for data processing and documentation on selected farm in agricultural production, *Przeglad Elektrotechniczny*, (2020), vol. 1, no. 218, pp. 218-221.

[17]U. Nepal, and H. Eslamiat, Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs. *Sensors.* (2022), vol .22, no. 2, pp. 464.

[18]M.N.S. Zainudin, M.S.S.S. Azlan, L.L.Yin, and W.H.Mohd Saad, Analysis on Localization and Prediction of Depth Chili Fruits Images Using YOLOv5, International Journal of Advanced Technology and Engineering Exploration. (2022), vol. 9, no. 97, pp. 1786-1801.

[19]R.Y. Choi, A.S. Coyner, J. Kalpathy-cramer, M.F. Chiang, and J.P. Campbell, Introduction to machine learning, neural networks, and deep learning, *Translational Vision Science & Technology*, (2020), vol. 9, no. 2, pp. 1-14.

[20]S. Kumar, A. Balyan, and M. Chawla, Object detection and recognition in images, International Journal of Engineering Development and Research, (2017), vol. 5, no. 4, pp. 1029-34.

[21]Y. Tian, G. Yang, Z. Wang, H. Wang, E. Li, and Z. Liang, Apple detection during different growth stages in orchards using the improved YOLO-V3 model, *Computers and Electronics in Agriculture*, (2019), vol. 157, no. 417-26.

[22] Y. Tian, G. Yang, Z. Wang, E. Li, and Z. Liang, Detection of apple lesions in orchards based on deep learning methods of cyclegan and yolov3-dense, Journal of Sensors, (2019), vol. 2019, pp. 1-14.

[23] A. Kuznetsova, T. Maleva, and V. Soloviev, Using YOLOv3 algorithm with pre-and post-processing for apple detection in fruit-harvesting robot, Agronomy, (2020), vol. 10, no. 7, pp.1-19.

[24] J. Liu, and X. Wang Tomato diseases and pests detection based on improved YOLO V3 convolutional neural network, Frontiers in Plant Science, (2020), vol. 11, no. 1, pp. 1-12.

[25] M.O. Lawal, Tomato detection based on modified YOLOv3 framework, Scientific Reports, (2021), vol. 11, no. 1, pp.; 1-11.

[26] L. Fu, Y. Feng, J. Wu, Z. Liu, F. Gao, Y. Majeed, Fast and accurate detection of kiwifruit in orchard using improved YOLOv3-tiny model, Precision Agriculture, (2021), vol. 22, no. 3, pp. 754-76.

[27] J. Yao, J. Qi, J. Zhang, H. Shao, J. Yang, X. Li, A real time detection algorithm for Kiwifruit defects based on YOLOv5, Electronics, (2021), vol. 10, no. 14, pp.1-13.