Analysis of Detection, Recognition and Distance Estimation of Chili using YOLOv5

M.N.Shah Zainudin, *Member, IAENG*, M.A.F.Ibrahim, N.Zarirah Nizam, W.H.M.Saad, M.R.Kamarudin, M.I.Idris, Mohd Safirin Karis, Sufri Muhamamad, and Raihani Mohamed

***Abstract*—In agricultural industries, autonomous robots have adopted to reduce labor-intensive tasks. Agriculture encompasses various activities such as soil cultivation, crop production, and animal rearing which done manually. It plays a crucial role in providing the world's food, textiles, construction materials, and paper goods. Harvesting is carry out manually using sharp tools like knives and scissors to cut the chilies from the plant. This process requires individual picking of each chili, consuming a significant amount of energy and time. In traditional chili harvesting, the process often requires a significant workforce especially for grading due to human eyes being prone to errors. In addition, characteristics of chili such as sizes, variations, texture and its localization are significantly different with other type of botanic vegetables. To address this, an investigation has conducted using You Only Look Once (YOLO) version 5-object detection to localize and classify chili variations. Datasets of 300 images with resolution of 640x640 pixels has utilized where 270 images used for training while 30 images used in testing. The foundation of Convolutional Neural Network (CNN) in YOLO, the proposed model successfully classified chili into three categories; green chili, red chili and rotten chili with detection accuracy above 93% in real-time implementation.**

***Index Terms*—Agricultural, YOLO, Object detection, CNN, Localization, Chili**

Manuscript received July 22, 2024; revised July 18, 2025.

This work was supported by Centre for Research and Innovation Management (CRIM), Universiti Teknikal Malaysia Melaka.

M.N.Shah Zainudin is a senior lecturer in Faculty of Artificial Intelligence and Cyber Security, Universiti Teknikal Malaysia Melaka, 76100, Hang Tuah Jaya, Melaka, MALAYSIA (corresponding author to provide phone: 606-270-2387; fax: 606-270-1045; e-mail: noorazlan@utem.edu.my).

M.A.F.Ibrahim is an undergraduate student in Universiti Teknikal Malaysia Melaka, 76100, Hang Tuah Jaya, Melaka, MALAYSIA (e-mail: afiq13579@gmail.com).

N.Zarirah Nizam is a senior lecturer in Faculty of Technology Management and Technopreneurship, Universiti Teknikal Malaysia Melaka, 76100, Hang Tuah Jaya, Melaka, MALAYSIA (e-mail: zarirah@utem.edu.my).

W.H.M.Saad is an associate professor in Universiti Teknikal Malaysia Melaka, 76100, Hang Tuah Jaya, Melaka, MALAYSIA (e-mail: wira_yugi@utem.edu.my).

M.R.Kamarudin is a lecturer in Universiti Teknikal Malaysia Melaka, 76100, Hang Tuah Jaya, Melaka, MALAYSIA (e-mail: raihaan@utem.edu.my).

M.I.Idris is a senior lecturer in Universiti Teknikal Malaysia Melaka, 76100, Hang Tuah Jaya, Melaka, MALAYSIA (e-mail: idzdihar@utem.edu.my).

M.S.Karis is a senior lecturer in Universiti Teknikal Malaysia Melaka, 76100, Hang Tuah Jaya, Melaka, MALAYSIA. (e-mail: safirin@utem.edu.my).

Sufri Muhammad is a senior lecturer Universiti Putra Malaysia, UPM Serdang, 43400, Serdang, MALAYSIA (e-mail: sufry@upm.edu.my).

Raihani Mohamed is a senior lecturer in Universiti Putra Malaysia, UPM Serdang, 43400, Serdang, MALAYSIA (e-mail: raihanimohamed@upm.edu.my).

I. Introduction

AGRICULTURE encompasses various activities such as soil cultivation, crop production, and animal rearing. It plays a vital role in supplying the world with food, textiles, construction materials, and paper products. This sector has been instrumental in advancing human civilization to its current heights, transforming societies from simple hunter-gatherer groups to more sophisticated ones. Agriculture also serves as a market for industrial goods, including agricultural machinery, equipment, and fertilizers. Global population growth is driving increased food demand, particularly across developing nations. This highlights the importance of paying close attention to the nutritional content and quality of the food produced. Furthermore, agriculture generates numerous off-farm operations, such as transportation and research programs aimed at enhancing agricultural and livestock activities. Over the course of human history, substantial technical innovations have applied to boost agricultural productivity despite limited resources [1].

Technology refers to the application of knowledge in a specific and repeatable manner to achieve practical objectives and becomes potential to assist the government in addressing the nation's food security concerns while easing the burden on available food supply [2]. It encompasses tangible elements like machinery and tools, as well as intangible components such as software. In the field of agriculture, technology plays a vital role in increasing output and labor productivity [3]. The advancement of technologies such as the Internet of Things (IoT) and unmanned aerial vehicles (UAVs) has significantly transformed and integrated with traditional agricultural practices. The integration of various wireless sensors and IoT devices has paved the way for numerous breakthroughs in crop development. These emerging technologies are now addressing various conventional agricultural challenges, including disease control, efficient irrigation, cultural practices, and responses to drought [4]. Semi-autonomous systems lie between fully autonomous systems and automated systems in terms of their level of self-containment and independence. Compared to completely autonomous systems, they are less distinct but nevertheless more independent and adaptable than automated systems. In a fully autonomous system, the human user is usually not involved unless necessary, while in a semi-autonomous system, there is shared control between the computer and

the human user. However, it is important to note that even fully autonomous systems must include some form of monitoring and direct human control as a precautionary measure in case of emergencies or if the system malfunctions [5].

The primary goal of the science of Artificial Intelligence (AI) is to enable robots and computers to perform tasks that would typically require human intellect. A computer has considered intelligent if it can execute cognitive tasks at a level comparable to a human. To achieve this, the computer must collect, organize, and apply human expert knowledge in a specific area to become intelligent. An effective data labeling procedure is essential for creating a good AI product. The performance of the model is highly influenced by both the quality and quantity of the training data. Proper and efficient data labeling is a crucial part of the training process. Data collection and identification play a vital role in training the model, with equal emphasis on the data quality [6]. Furthermore, AI also enables estimation and calculation of object distance using depth estimation. This process involves determining the distance of objects in images obtained from cameras, providing valuable information about the subject's location and surroundings.

Although the agriculture sector in food production has incorporated machines into many of its activities, certain tasks still heavily rely on human labor. For instance, the harvesting of bird's eye chili for production continues by using traditional methods, involving handpicking. This is because the unique characteristics of the fruit make it challenging to machine to complete the task effectively. Bird's eye chili plants produce small, tapered fruits, typically two or three per node. These botanic vegetables become green when unripe and turn red as they mature, becoming tiny, thin, and pointed with a pungent odor. Harvesting is carry out manually using sharp tools like knives and scissors to cut the chilies from the plant. This process necessitates choosing each chili by hand, which takes a lot of time and effort. Additionally, after harvesting, the chilies need to sort by human method to separate the matured ones from the unripe ones. The grading process might lead to inefficient using traditional approach due to the human tends to make mistakes. However, by implementing AI in the harvesting process, it is possible to differentiate the chilies by their condition directly from the plant through image labeling. This data can significantly improve the harvesting process, making it more efficient and accurate compared to traditional methods. The incorporation of AI can pave the way for a fully autonomous system, where machines handle the picking and selection processes with precision and speed [7]. Although the harvesting robot can reduce the labor cost, it does not have the ability to identify the grades as well as the localization of the chili. Hence, the use of distance information becomes a solution in this matter.

This article composing few contributions; YOLO version 5 is applied to recognize and localize the chili as well as differentiating between green, red and rotten chili. The model is also been tested to recognize the chili from recorded video by differentiating the chili variations. To assess the model's capability in real-world scenarios, the trained model was also tested on real-time video for chili detection. Section 2 presents the previous literature work, section 3 explains the proposed material and methods for experiments, section 4 discusses the analysis result and discussions from the experiment conducted followed by the last section end up for conclusion for the experiment.

II. Background Study

As a key cash crop in Asia and Africa, chili pepper production generates substantial revenue for farmers, ranging from smallholders to large agribusinesses. However, chili producers often face challenges related to pests and diseases, necessitating timely and informed decisions for a successful harvest. The study introduces a Decision Support Platform (DSP) tailored for chili cultivation, leveraging real-time disease and nutrient deficit detection to empower farmers with actionable insights for prompt decision-making. The proposed system combines IoT, cloud computing, and data analytics. The paper presents preliminary results on CNN-based chili classification as well as the structure and design of the suggested chili-DSP. The outcomes reveal that CNN provides precise predictions and effectively learns from the datasets. The authors suggest that their work can expands to larger datasets for real-time chili illness categorization. The chili-DSP aims to offer a comprehensive feature set and support to chili producers, enabling them to enhance production while reducing losses. The study focuses on two illnesses, namely powdery mildew and leaf spot, utilizing a datasets comprising 86 images, with 60% for training, 20% validation, and 20% testing. After 23 epochs, an early halt is implement, and the model achieves favorable performance metrics, including an accuracy of 87%, precision of 88%, recall of 92%, and an f1-score of 88% [8].

To overcome the time-consuming and inefficient manual grading process, an automatic grading system has developed to identify and categorize crack chili after de-stemming. The experiment utilized a CNN model to detect fractures, and the actuator has employed to provide appropriate control signals for classification. The system utilized TensorFlow as the database structure, OpenCV for image processing, and the Keras API for creating and training neural network models. In both static and operational scenarios, the system achieved high accuracy rates, with 97% and 95.3%, respectively. Remarkably, even after a 120-hour storage period, with the chili body wrinkled due to drying, the system still achieved a success rate of 93%. The results demonstrate the reliability and effectiveness of the model in real-world applications [9].

The classification of chilies based on the intersection of the calyx and the apex. The extracted chilies then further divided into different ripeness degrees using a CNN model. The program tracks the number of samples taken from the farms and categorizes the quantity of images based on various chili sizes. An analysis will conducted to evaluate

how well the chili's size will classified after size categorization. For this experiment, a CNN model with a three-layer operation, including a standard layer, a max-pooling layer, and a fully linked layer has utilized. The images of the chilies will fed into the CNN model to observe and learn the patterns of its categories, which include immature, moderately mature, and mature classifications. The preliminary experiment used a learning rate of 0.0001 and ran for 200 epochs. The optimization algorithm chosen was the Adam optimizer, which combines stochastic gradient descent with adaptive learning rate modification. In total, 240 photographs with equal-sized images of 195 × 260 pixels has used in the investigation. The findings indicate that the proposed model can reliably classify chilies into three maturity stages: immature, moderately developed, and mature [10].

In agriculture sector, automating processes that utilize image-processing techniques to categorize chili crops based on their color, shape, and texture is crucial. The authors developed a portable sorting device that utilizes Artificial Neural Networks (ANN) to separate chilies according to their color. Plot disagreement has used to assess the model's performance. Initially, the learning algorithm trained using a sample of 10 images of green chilies and 10 images of red chilies. Later, the algorithm's effectiveness has compared with a larger datasets comprising 40 samples of chili images. The design of the smart sorting machine was versatile enough to be applied in the agricultural industry, where a significant number of chili crops with various distinctive qualities need to be processed simultaneously. The results emphasize the importance of research in sorting mechanisms, even though there might be some additional cost involved [11].

The proposed system aims to establish fruit maturity grading through object identification by training neural network models. This will enable the system to differentiate between ripe and unripe fruits, and subsequently, robotic arms has utilized for harvesting. Traditionally, farming decisions have relied heavily on human expertise. In the article [12], the authors put forth a multi-layer perceptron model with Keras to predict the location and motion of a multi-axial robotic arm. The input to the neural network consists of pixel coordinates of the center of the target crop in the images after object recognition, while the output represents the movements of the robotic arms. To achieve object detection, a single-shot multi-box detector model has combined with a MobileNet version 2 CNN, which serves as the visual feature extraction model. The model then trained to detect and categorize crops from gathered images. Empirical data shows that the suggested model achieved a mAP of 84%, surpassing the performance of other models. Furthermore, the arm selection results demonstrated a mAP of 89% [12].

Another work on chili localization has done by using YOLOv5. In comparison with other versions of YOLO model, YOLOv5 has known its ability to produce an outstanding performance and recorded high efficiency in detection. The author reported mAP above 80% achieved to differentiate between red and green color chili. Even the performance of single chili slightly low, combination of various colors is still acceptable. Due to lighting condition and reflection of an artificial fruits, the detection of green chili is lower than red chili [13]. Another work has also reported for recognition of chili using an object detection algorithm. In this work, two different class of chili species, cili-padi and ghost pepper has utilized. The performance of detection is been compared by using two different algorithms; YOLOv5 and Mask-RCNN. Two features used including its shape and colors for differentiating the chili categories between mature and immature. YOLOv5-l is able to achieve 78% of precision, while YOLOv5-s and YOLOv5-m recorded 73% and 75% respectively. Mask-RCNN is achieved an outstanding performance with above 95% of precision. Yet, the time taken to infer the testing subset is definitely longer above 120000ms than YOLO models [14].

III. Materials and Methods

A. *Deep Learning*

Deep learning is a specialized category within the realm of machine learning algorithms that builds upon the foundational principles of machine learning, particularly utilizing neural networks, to tackle highly intricate and complex problems. While machine learning has demonstrated its efficacy in solving relatively simple to moderately complex tasks, it may struggle to deliver high-performance results when dealing with exceptionally intricate challenges. Deep learning has emerged as a transformative solution, harnessing recent theoretical breakthroughs and technological advancements to address these longstanding limitations across diverse application domains. For instance, it has found applications in cutting-edge fields like self-driving cars, image recognition on social media platforms, and language translation, where it excels in handling intricate problems and generating accurate outcomes. Deep learning is a specialized domain within the field of machine learning that is dedicates to developing algorithms capable of learning and understanding both intricate and fundamental data abstractions. These abstractions can be challenging or even impossible for traditional machine learning algorithms to grasp. Deep learning models draw inspiration from diverse fields like neuroscience and game theory, often mirroring the underlying organization of the human nervous system. The future holds the promise of software becoming less rigidly hard-coded, allowing for more comprehensive and versatile solutions to various problems.

Deep learning algorithms possess the remarkable ability to learn complex patterns, making them adept at prediction and classification tasks. Deep learning models commonly comprise layers of neurons, which are nonlinear units utilized for processing input data. Each layer in these models operates at different levels of abstraction. Deep neural networks recognized by their substantial number of hidden layers, as the inputs and outputs of these layers may

not be straightforwardly comprehensible beyond their relationship with the preceding layer. The distinctiveness of an architecture determined by the inclusion of various layers, and the functions within the neurons of these layers dictate the diverse applications of a specific model. While users can customize these layers, their core functions are application-specific, offering superior flexibility compared to traditional machine learning methods for regression and classification tasks. The inclusion of multiple layers in the model enables it to process inputs in a manner that progresses from simple features to more intricate structures. The ultimate aim of these models is to perform tasks with reduced reliance on explicit guidance. This promise of addressing both supervised and unsupervised learning challenges is one of their significant advantages [16].

Deep learning has proven to be highly effective due to its capacity to process large datasets. Hidden layer methods, especially in pattern recognition, have become more popular than classical techniques. One notable deep neural network model is the Convolutional Neural Network (CNN) [17-18]. CNNs excel at tasks such as recognizing handwritten numbers, identifying cancer types, and facial recognition. However, training deep learning models requires extensive datasets and significant computational power. CNNs is one of the deep neural network that frequently used to analyze visual data. Numerous applications has shown promising results including image and video for recognition, segmentation, and classification in medical fields as well as natural language processing. In essence, CNNs are customized multilayer perceptrons where is the network is a fully connected in which each neuron in a layer links to all other neurons in a layer above it [19]. A network input has formed by multiplying number of heights with number of input, input channels and input width then will fed into a CNN. Another fascinating problem that computer vision faces is object recognition in addition to conventional image classification.

Full names of authors are preferred in the author field, but are not required. Put a space between authors' initials.

B. Convolutional Neural Network

Unlike traditional machine learning algorithms that operate linearly, deep learning algorithms such CNN are organized in a hierarchy of increasing complexity and abstraction. In the area of image recognition, deep neural networks CNN has demonstrated to produce an outstanding performance [20]. CNN consist of neurons that undergo a learning process to optimize themselves. Each neuron performs operations such as scalar products and nonlinear functions to process information. A single perceptive score function, which serves, as the weight from the input raw image vectors to the final output of the class score to represent the complete network. The last layer includes loss functions related to the classes, and conventional techniques created for standard ANN are still applicable.

The key distinction between CNN and traditional ANN lies in the widespread use of CNN in the field of pattern detection within images. This makes it possible to integrate architecture made to manage particular image features, improving the network's adaptability for image-focused activities and minimizing the amount of parameters needed to build the model. Traditional ANN models often face challenges when dealing with the computational complexity required for processing image data, which is considers one of their major limitations. For standard machine learning benchmarks involving handwritten digits with relatively low image dimensional of just 28x28 pixels, ANN can be effective. However, when dealing with extensive input, such as 64x64 colored images, the number of weights in a single neuron of the first hidden layer significantly increases to 12,288. This substantial growth in the number of weights highlights the drawbacks of employing traditional ANN models, as it would require much larger networks to handle such input data effectively [21].

C. You Look Only Once

Individuals are extremely adept at quickly identifying items in an image, knowing where they are, and understanding their relationships with ease. Because of our quick reactions and accurate eyesight, humans can perform complicated activities like driving with little conscious thought. Creating object identification algorithms that are simultaneously fast and accurate could lead to a variety of uses, including the development of responsive and all-purpose robotic systems, assistive devices that can give users real-time scene information, and computer-controlled vehicles that do not need specialized sensors. In modern object detection systems, various techniques employs to identify and assess objects within an image. For instance, classifiers uses to detect objects and evaluate them at different sizes and locations within a test image. Before applying a classifier to these proposed boxes, other methods like as R-CNN generate possible bounding boxes for objects in an image using region proposal algorithms. Following classification, bounding boxes are refined, duplicate detection are eliminated, and the boxes are given new scores based on additional information in the scene during post-processing.

You Only Look Once (YOLO) is a unique method that reframes object detection as a single regression problem. YOLO, which processes the entire image at once, instantly predicts bounding box coordinates and class probabilities. This results in faster and more efficient detection performance, as it eliminates the need for region proposals and allows for training on complete images. Compared to conventional object detection techniques, YOLO models offer several advantages. One of the key strengths of YOLO is its speed and efficiency. It simplifies the detection process by treating it as a prediction model. During testing, YOLO only needs to run the neural network on a new image to make predictions for detection. The base network

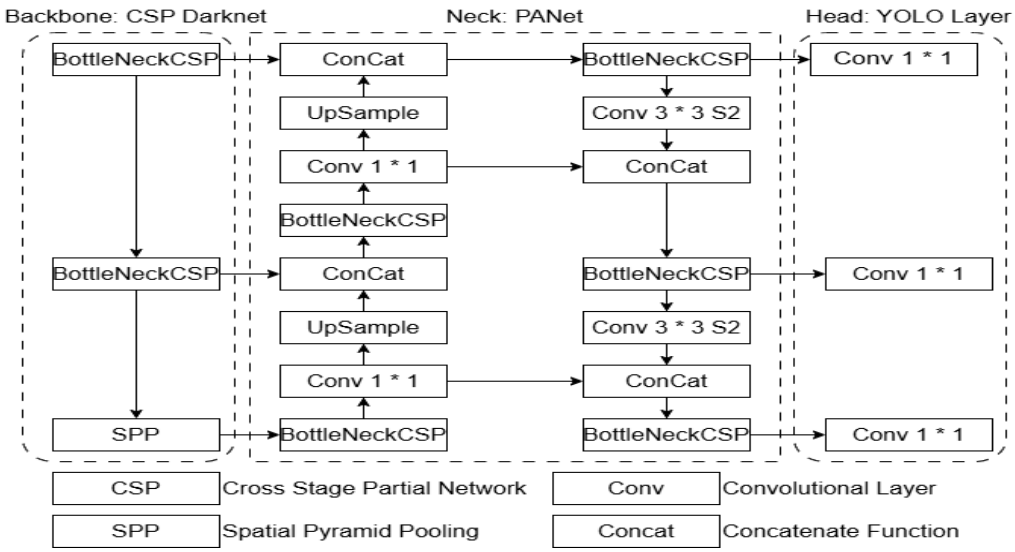

Fig. 1. YOLOv5 architecture

can achieve a speed of 45 frames per second on a Titan X GPU without batch processing, while the fast version can go even faster, reaching over 150 frames per second. This remarkable speed allows YOLO to handle real-time live streams with a latency of less than 25 milliseconds. Moreover, YOLO achieves a mean average accuracy that is more than double that of comparable real-time object detection systems.

As shown in Fig. 1, the YOLOv5 network is composed of three parts: the head, the neck, and the backbone [22]. YOLOv5 integrated a cross stage partial network (CSPNet) to create CSPDarknet, which serves as the foundation of the Darknet. An input is supply into CSPDarknet for feature extraction then it will gives an input for PANet to fuse the data. Finally, the detection results from the YOLO layer pass through optimized convolutional layers that minimize parameters and floating-point operations per second (FLOPs) via gradient techniques. YOLOv5 implements CSPNet to efficiently handle gradient information flow in its large-scale backbone network, eliminating gradient redundancy while maintaining feature richness. This could result in a smaller model size in addition to increasing inference accuracy and speed. PANet served as a bottleneck for YOLOv5 in order to boost data throughput.

The proposed architecture integrates an enhanced Feature Pyramid Network (FPN) with a bottom-up approach into PANet to strengthen low-level feature propagation. This modification simultaneously improves localization signal extraction from lower layers and enhances object positioning accuracy. For multi-scale prediction, the system generates three distinct convolutional layers (18×18, 36×36, and 72×72), enabling YOLOv5 to effectively detect objects across varying scales (small, medium, and large) through its specialized YOLO layer.

A key advantage of YOLO lies in its unified architecture for end-to-end object detection. The system processes entire images through a single neural network that concurrently predicts bounding boxes and class probabilities, eliminating the need for separate region proposal stages [23]. This holistic approach enables real-time performance by analyzing the complete image context during both training and inference, allowing YOLO to capture not just visual features but also valuable contextual relationships between objects. For instance, the fast R-CNN is a top detection technique, but YOLO outperforms it by making fewer than half as many background mistakes. By taking a global view of the image, YOLO is able to achieve more accurate and efficient object detection results. Furthermore, YOLO excels at learning generalization representations of objects. When evaluate in artistic photographs and trains in real-world images, YOLO outperforms top detection techniques such as DPM and R-CNN. Its ability to generalize well across different contexts and handle unexpected inputs makes it less likely to fail when used in unfamiliar situations. This adaptability and robustness make YOLO a highly effective and versatile object detection model [24].

YOLOv5 stands out as a simpler and more reliable deep learning system compared to other alternatives. Notably, it achieves significantly faster performance and demands fewer processing resources while delivering equivalent outcomes. YOLOv5's architecture builds on YOLOv4, employing CSPDarknet as its encoder, complemented by the inclusion of Path Aggregation Network (PANet). Additionally, YOLOv5 replaces the Leaky ReLU and Hard-swish activation used in YOLOv4 with the SiLU activation function. These improvements contribute to YOLOv5's efficiency and effectiveness in object detection tasks [25]. The YOLOv5 architecture has chosen because of its lightweight design, which enables users to train the model with minimal processing resources and thereby reduces costs. Additionally, its compact size enables its deployment on mobile devices. However, there are both advantages and disadvantages concerning the memory usage of YOLOv5. It is 88% more compact and 180% faster than YOLOv4, achieving an impressive frame rate of 140 Frames per Second (FPS) compared to YOLOv4's 50 FPS. Despite these improvements, the precision difference between YOLOv4 and YOLOv5 is minimal, around 0.003,

rendering them almost identical in performance. YOLOv5 is introduce with four different versions (s, m, l, and xl), where larger models offer more tunable parameters and potentially better performance. However, it is essential to consider that larger models with more parameters may lead to longer training times. For real-time detection, smaller or medium-sized models that are more suitable [26].

IV. Experimental Results and Analysis

A. Data Collection

The goal of this work is to localize, recognize and distinguish the green, red and rotten chili from the plant. 300 images from various categories of chili with various angles of direction is collected from 2D mobile devices camera. The image has divided into two different subsets; training and testing. 270 images are used for training while the rest 30 images will reserved for testing. A background that is all white in order to exclude any possible outside distractions and create a stark contrast. The chili has rotated at various angles before the images has taken. The datasets has also taken at various camera-to-subject distances. These variants were included, making the datasets more complete and addressing various conceivable conditions found in real-world applications. Fig. 2 depict the sample of image chili plant.

Fig. 2. Sample chili image

B. Image Labeling

After drawing the bounding outlines around the objects, the appropriate class label as in Fig. 3 has assigned to each of them. YOLO supports multiple classes, allowing objects with designated with class names such as “green chili”, “red chili” and “rotten chili”. Once the image annotation is complete, the annotations must be save in the specific format required by YOLO. Usually, YOLO utilizes a text file for each image, where each line represents an object annotation and contains information like the class identifier, bounding box coordinates, and other relevant details. The annotated images and their corresponding annotation files has organized within a hierarchical directory structure. Each image should have a separate annotation file with the same name but different file extensions, for instance, “image1.jpg” and “image1.txt”.

Fig. 3. Sample labeling image

C. Model Training and Testing

YOLO architecture is widely regarded as the primary choice for object detection, with several versions available, such as YOLOv3, YOLOv4, and YOLOv5. To use the YOLO model, pre-trained weights and their corresponding configuration file are required. The configuration file specifies important details about the model's architecture, such as the number of classes to detect dimensions of anchor boxes, input size, and other parameters. By using pre-trained weights, the YOLO model is initializes with valuable features learned from a large datasets, making it a form of transfer learning. During training, the YOLO model adjusts its parameters to improve object detection accuracy. The instruction procedure typically involves adjusting hyper-parameters, such as learning rate, weight decay, and sample size, to find a configuration that yields better training results. Experimentation and model performance monitoring on the validation set may be necessary throughout this procedure. The model's performance on the validation set is evaluates during training. Metrics such as precision, recall, and mean mAP used as indicators to assess the accuracy of the model's object detection capabilities. By fine-tuning the model and monitoring its performance, the YOLO model can achieve excellent results in object detection tasks.

A portion of the annotated datasets is set aside specifically for testing purposes. This testing datasets contains images that unused during the training or validation stages. This ensures that the custom YOLO model is evaluate on completely unseen data, providing a more accurate representation of its performance in real-world scenarios. Once the custom YOLO model is train, its weights are loaded, ensuring they match the model's architecture and configuration used during training. The model's configuration for inference is prepare by importing the model architecture and adjusting parameters like input size and class identifiers. During the testing phase, the model iterates through the images in the testing datasets and performs inference on each image. For each object found in the images, the model predicts bounding boxes, class labels, and confidence scores. Typically, a list of bounding boxes, class labels, and confidence scores generated along with these predictions. The predicted bounding boxes, class labels, and confidence scores is

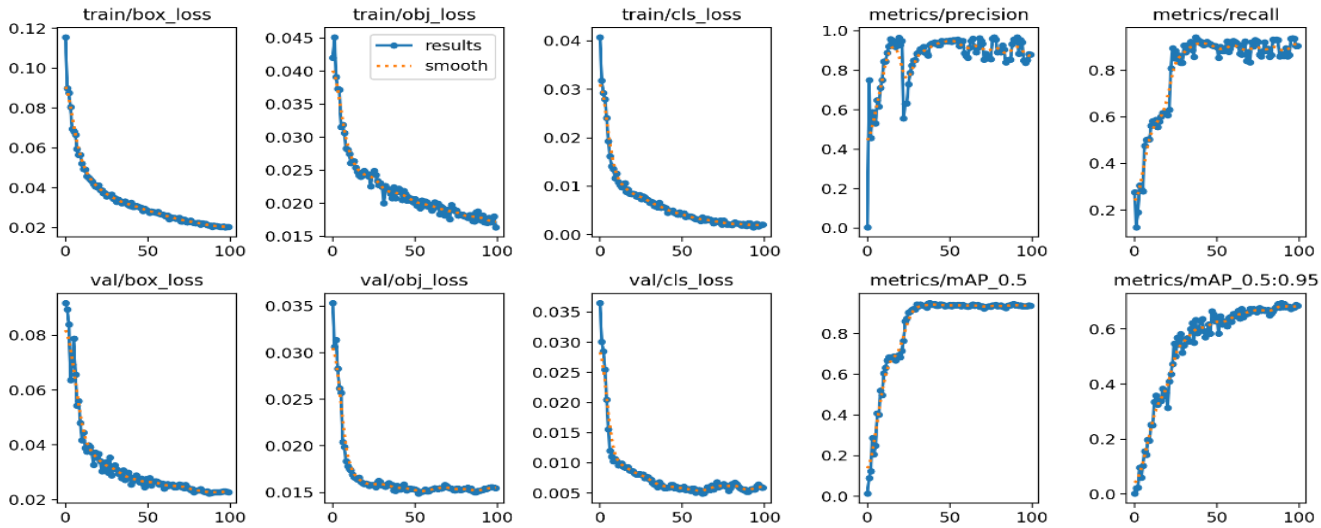

Fig. 4. Training and validation performance for chili detection

impose on the example images to assess the model's performance. This visualization helps in comprehending how well the model is detecting objects. Testing a customized YOLO model is crucial in assessing its effectiveness in detecting objects in unseen data. By evaluating its performance on a distinct testing datasets, potential areas for improvement can be identify, and informed deployment decisions can be made. In this experiment, we label the green chili as 0, red chili as 1 and rotten chili as 3.

The training and validation performance graphs as in Fig. 4 for chili fruit detection model reveal both its strengths and areas for optimization. The box loss curves indicate effective bounding box regression, with both training and validation losses converging near 0.02, suggesting accurate localization with minimal overfitting. Notably, the validation loss plateaus after epoch 50, signaling diminishing gains from continued training. Meanwhile, the classification loss trends towards zero, indicating that the model effectively distinguishes between different chili categories. However, intermittent fluctuations in validation loss imply occasional misclassifications, particularly in the rotten chili class. This is likely due to limited and visually ambiguous samples, which reduce the model's discriminative power. Incorporating additional rotten chili images with varied visual features could significantly reduce this ambiguity and improve classification accuracy. The reported performance metrics support these observations. The model achieves high precision (~0.95) and recall (~0.90), reflecting its ability to minimize false positives and false negatives effectively. However, the mAP@0.5:0.95 score plateaus at approximately 0.05, indicating difficulty in maintaining high precision under stricter IoU thresholds.

Table I shows the performance of detection for all three categories chili using YOLOv5 using testing subsets. These results show that the model is highly effective at identifying the correct chili class when it detects an object (high precision), and even better at detecting most chili objects present (very high recall). However, the relatively lower mAP@0.5:0.95 score reflects the model's decreasing localization accuracy at stricter IoU thresholds, indicating a need for bounding box refinement. For fair comparison, all types of chili (green, red and rotten) applied with the same numbers of testing sample with 30 images. As we can observed, an average precision of green and red chili achieves an outstanding performance above 95% of accuracy. Green chili shows excellent detection and classification performance with both precision and recall above 95%, and the highest mAP@0.5 value. This suggests the model is very consistent in both identifying and localizing green chilies. The mAP@0.5:0.95 also indicates strong robustness under stricter evaluation, likely due to clear color contrast and sufficient sample representation.

Red chili detection maintains high accuracy, though slightly lower than green chili, especially in recall. This could be attributed to overlapping instances or less distinguishable shapes, resulting in some missed detections. Nevertheless, the model performs reliably in most test cases, as evidenced by the strong mAP@0.5. Since we are using the real chili plant for this experiment, it is challenging to have a good sample of rotten chili due to the chili conditions. Rotten chili exhibits the lowest precision (76%) among the classes, meaning that some predictions made as "rotten" are false positives. However, the high recall (90%) indicates that the model was able to detect most actual rotten chilies. This class also achieves a respectable mAP@0.5 and mAP@0.5:0.95, despite being underrepresented. This highlights potential overfitting or confusion with similar visual features, and suggests the need for more training samples and better augmentation for this class. The chili needed to be harvest for few days before it becomes rotten. About 76% of accuracy has obtained where it considered the lowest performance from all three classes. In average, 94% of mAP is recorded which is considerably outstanding in differentiating various kind of chili categories.

TABLE I
ACCURACY OF DETECTION FOR VARIOUS CHILI CATEGORIES

Class	Precision	Recall	mAP@0.5	mAP@0.5:095
Green	0.954	0.952	0.974	0.725
Red	0.919	0.868	0.938	0.663
Rotten	0.766	0.900	0.900	0.670
Average	0.880	0.906	0.937	0.686

The graph as in Fig. 5 illustrates the relationship between model confidence scores and precision across the three-chili classes (green, red, rotten) and their collective performance. The X-axis shows the confidence threshold (from 0 to 1), while the Y-axis shows precision, i.e., the proportion of correct predictions out of all predictions made at a given confidence level. The blue line represents the average precision over all classes. It reaches 100% precision at a confidence threshold of 0.976, indicating that at high confidence, the model predictions are extremely accurate. The green chili class maintains very high precision (≥95%) across most of the confidence range. This indicates the model is very confident and accurate in detecting green chilies. The curve flattens near the top, meaning the model rarely misclassifies green chilies, even at moderate confidence. Red chili also shows strong precision performance, with values consistently above 90% throughout the range. A slight dip at the end near 1.0 confidence suggests a few misclassifications at very high confidence, possibly due to visual similarity with green chilies or class imbalance. The rotten chili class has the weakest precision performance across all confidence levels. It starts low (≈40–50%) and slowly increases, peaking below 90%. This curve shows a more gradual slope and higher variance, indicating that the model is less confident and more error-prone when predicting this class. The likely cause is insufficient training data or visual ambiguity (e.g., color similarity with leaves, partial rotting).

D. Experiments on Various Conditions

In order to ensure the model is able to detect the chili from various conditions, different sources of input has utilized. Fig. 6 shows the sample of detection using image that has been captured beforehand.

Fig. 6. Detection of chili images

The model is capable to differentiate between green and red chili in the plant with average above 90% of accuracy. We also tested the model by measuring its ability to detect the presence of chili from the recorded video. The model demonstrates promising performance, even when the video background is not a plain white background. This finding suggests that the model is robust in handling diverse background conditions and displays its effectiveness in accurately detecting and classifying chili fruits. This finding suggests that the model is robust in handling diverse background conditions and displays its effectiveness in accurately detecting and classifying chili fruits. On average, the red chili records an accuracy above 90% due to its distinct red color, making it easily distinguishable from the leaves. However, some green chili instances do not yield promising results. This is likely because the green chili's color closely resembles that of the plant leaves, making it challenging the model to differentiate between them accurately. Fig. 7 displays the sample of experimental result of chili detection from recorded video.

Fig. 7. Detection of chili images from real-time video

The last part of experiment is by utilizing the model on real-time experimental conditions. In this part, we use low-resolution camera using webcam to evaluate the model performance for real-time situations. As we know, when it comes to the real farming, there are few aspects need to be tackle such as weather conditions, lighting, clutter, etc. The live testing conditions and lower resolution of the webcam can affect the model's performance to some extent, but it still demonstrates the ability to distinguish chili from their plants. Because of the camera's limitations and lower image quality, the objects placed close to the webcam, leading to a higher accuracy of detection for both chili categories, with an average accuracy of above 96%. Fig. 8 shows the sample of experimental result of chili detection for real-time conditions.

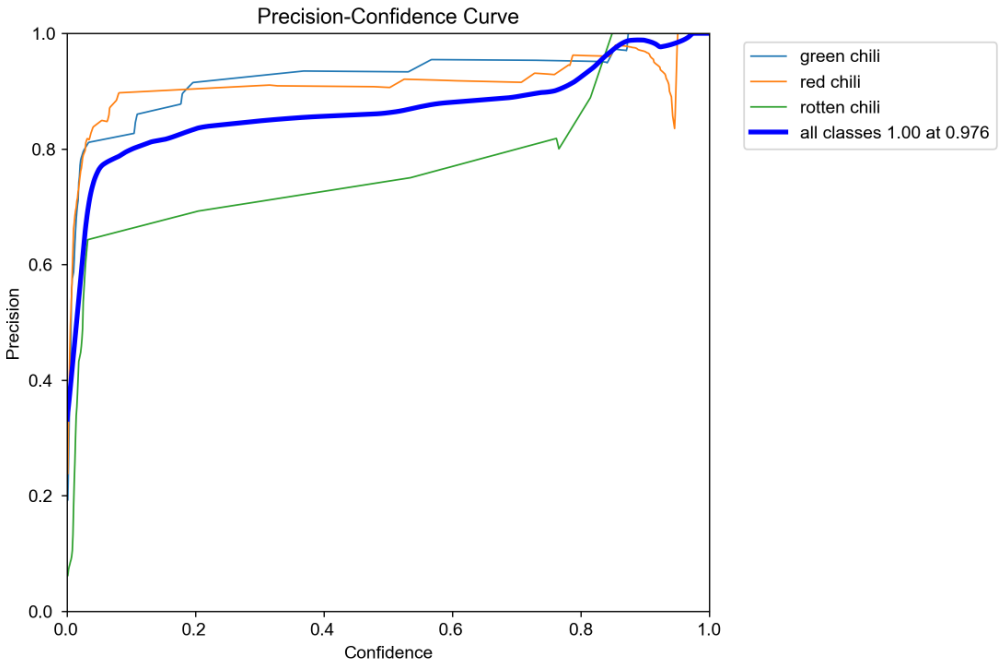

Fig. 5. Relationship between model confidence scores and precision across the three-chili classes (green, red, rotten)

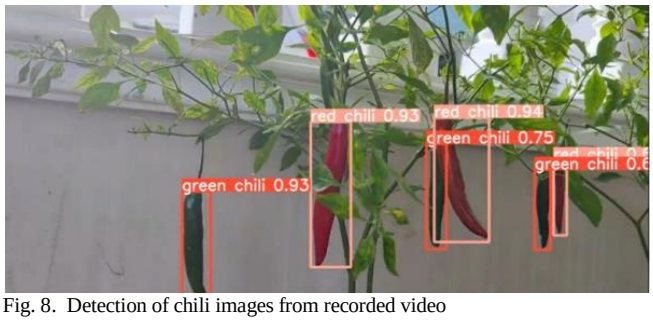

Fig. 8. Detection of chili images from recorded video

and stability under varying conditions, guiding the refinement process toward achieving optimal detection accuracy.

TABLE II
COMPARISON OF ROBOFLOW DATASET

YOLO	Total images	Epochs	Precision	Recall	mAP	Time (h)
YOLOv5	4476	100	0.926	0.866	0.932	1.047
YOLOv7	4476	68	0.971	0.940	0.974	3.674

E. Experiments on Different Version YOLO

In addition, we also tested the chili datasets sourced from Roboflow, providing a robust foundation with 4,376 pre-labeled images. These images include a diverse range of chili types, specifically red and green varieties, capturing the variability of chili shapes, colors, and orientations. The dataset underwent augmentation techniques to enhance its diversity, which improves the model's robustness by exposing it to a broader set of visual conditions.

Additionally, 100 images of chili plants has captured from various angles using the Intel RealSense Camera D455 to test the model's compatibility with RGB image inputs from this specific camera. These 100 images manually labeled using Roboflow's labeling tools to ensure precision in the dataset and to assess the effectiveness of the Intel RealSense Camera D455. Throughout the training phase, several parameters, such as the number of epochs and dataset size, were iteratively fine-tuned to optimize model performance. Adjustments to these parameters systematically applied to maximize the model's detection and localization capabilities for chili objects. Performance metrics, including precision, recall, and mAP, tracked to assess accuracy across different parameter configurations, as presented in Table II. This comparative analysis of training metrics provides insight into the model's behavior

Comparative performance results show that the YOLOv5 algorithm demonstrates high efficiency and accuracy under different training conditions. With an expanded dataset of 4,476 images over 100 epochs, YOLOv5 achieved an improved precision of 92.6%, although this required a longer training time of 1.047 hours. In comparison, YOLOv7, trained with 4,476 images over 68 epochs, reached the highest precision of 97.1% but necessitated a substantially longer training duration of 3.674 hours. Although YOLOv7 demonstrated superior precision, its extended training time highlights a trade-off between model accuracy and computational efficiency. In this study, YOLOv5 has selected as the preferred model for object detection due to its balanced performance, achieving over 90% precision with a considerably shorter training time than YOLOv7. YOLOv5's capability to deliver high accuracy in less time is especially advantageous in applications where computational resources are constrained, making it both cost-effective and efficient for GPU-based processing [27-28]. Fig. 9 to 11 visually illustrate the detection of YOLOv5 across different chili categories, highlighting the model's accuracy in detecting and localizing chili fruits across diverse image types. These figures provide a comprehensive visualization of YOLOv5's performance, reinforcing its suitability for real-world object

detection tasks by highlighting the precision and reliability of its bounding box predictions in various detection scenarios.

Fig. 9. Detection of single chili (red and green)

Fig. 10. Detection of multiple chilies (red and green)

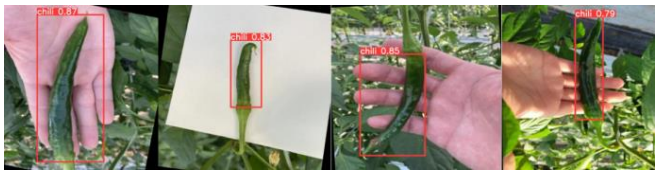

Fig. 11. Detection of harvested chili

Despite the similar training time per epoch for these lighter YOLO architectures, YOLOv7 required nearly twice the total training duration to reach convergence and the added computational depth in its architecture. This extended training time, while providing higher model accuracy, underscores the more intricate structure of YOLOv7 compared to YOLOv5, which more streamlined for efficient training. A notable observation is the improvement in inference speed demonstrated by YOLOv7 over YOLOv5, consistent with expectations for lighter YOLO models in real-time applications. YOLOv7's optimized architecture enables faster inference speeds critical for real-time applications, though these performance gains necessitate longer training periods compared to previous versions. As the complexity of object detection networks rises, so too does the time required for precise identification, highlighting a necessary balance between network sophistication and operational speed. The decision to select an appropriate YOLO architecture becomes essential for applications prioritizing real-time tracking and high bounding-box accuracy. Selecting between YOLOv5 and YOLOv7 requires an evaluation of application needs, computational resources, and acceptable trade-offs between processing efficiency and detection precision, underscoring the importance of aligning network complexity with specific project requirements to achieve optimal outcomes.

F. Distance Estimations

The evaluation of the model's performance in distance estimation reveals that the model performs with a high degree of accuracy within a range of 40 cm to 90 cm. Within this distance range, the model estimates closely align with actual distances, indicating a strong ability to interpret spatial parameters effectively and project bounding boxes that closely match the true object size. This minimal error range suggests that the model's calibration and algorithmic approach to distance measurement are highly reliable within this specific field of operation. The accuracy in this range likely stems from a combination of well-tuned parameters and precise object recognition, highlighting the model's suitability for applications requiring detailed spatial estimations within short to moderate distances. For distances less than 40 cm, however, the model exhibits a tendency to underestimate the true measurement, leading to a smaller bounding box projection around the subject. This behavior may be due to the increased parallax and perspective distortions typically encountered at very close ranges, which can introduce spatial ambiguities that challenge the model's algorithms. Additionally, sensors and camera limitations may contribute to the underestimation at close distances, as finer details become harder to capture accurately, affecting the model's overall depth perception.

In contrast, when distances exceed 90 cm, the model demonstrates a trend of overestimating the actual distance. This overestimation results in the projection of a larger bounding box than the actual object dimensions warrant. Factors contributing to this overestimation might include diminished resolution and decreased sensitivity in distinguishing depth at greater distances, which can influence the model's ability to capture the relative size of objects. As distance increases, depth information becomes less precise, and any minor error in estimation can become magnified, leading to noticeable discrepancies. Fig. 12 visually captures an image taken during the distance estimation testing, displaying the detection of a chili fruit and the corresponding bounding box as projected by the model. This figure provides a clear illustration of how the model's estimations visually translate into spatial representations. It also serves as a reference for understanding the interaction between the bounding box and the actual object when the model operates within varying distances.

For further quantify the model's performance, Table III presents detailed comparisons between estimated distances and actual measurements. The table highlights individual data points, highlighting where the model's estimates align or diverge from true values. Through these data points, the error percentage has calculated, offering a concrete measure of the model's accuracy across the entire distance range. This error analysis allows for an objective assessment of the model's capacity to generalize its distance estimation and provides insights into specific areas for improvement. The observed variations in accuracy underscore the need for adjustments or calibrations in scenarios requiring accurate detection at distances outside the model's optimal range. Enhancing the model's algorithms to improve accuracy for both closer and farther distances could involve integrating additional calibration data or refining feature extraction techniques. These adjustments help reduce detection errors at close and far distances, expanding the model's accurate detection range.

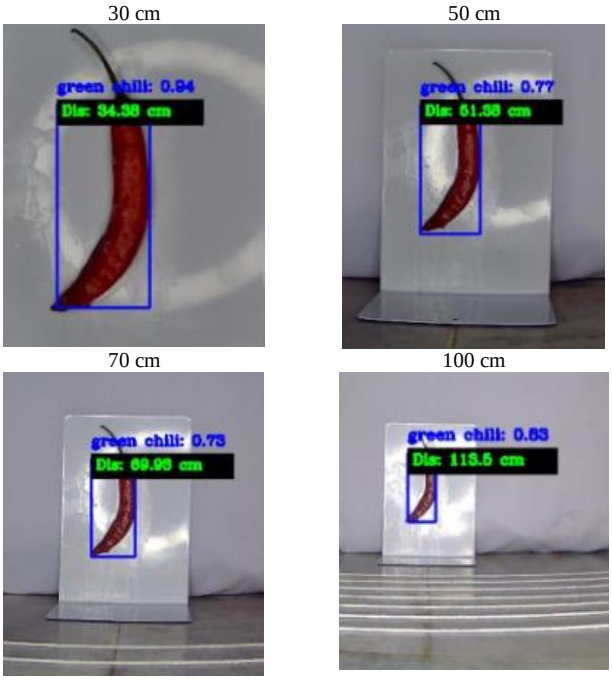

Fig. 12. Distance estimation of chili image

TABLE III
DISTANCE ESTIMATION WITH ACTUAL DISTANCE

Actual (cm)	Estimation (cm)	Error (%)
30	34	13.33
35	37.5	7.14
40	41.5	3.75
45	47	4.44
50	52	4.00
55	57	3.51
60	64	6.67
70	72	2.86
80	83	3.75
90	88	2.22
100	114	14.00

V. CONCLUSION

Regarding chili recognition and localization, the YOLOv5 model demonstrated high effectiveness in accurately detecting chili from their plants, achieving a precision score above 90%. The model displays excellent performance across various sources, including images, videos and live webcam. The evaluation of green, red, and rotten chili detection accuracy accomplishes by analyzing the model's parameters after training. The objective has successfully achieved with the model achieving a mAP score almost 94% during the testing phase. High precision at high confidence (especially >0.9) confirms the model's strong discriminative ability when it is certain. The green and red chilies can be reliably detected even at lower confidence levels, which is critical for real-time agricultural use. The rotten chili reflects the need for data augmentation and better labeling—this class introduces the greatest uncertainty and should be prioritized for improvement. If deployed in production (e.g., robotic picking or sorting systems), the confidence threshold could be tuned to ~0.8–0.9 to balance high precision with sufficient recall, especially for green and red chili detection. In conclusion, the investigation demonstrated the effectiveness of YOLO-based algorithms in accomplishing the desired objectives. Improved production, lower labor costs, and increased efficiency in chili farming made possible by the effective recognition, localization, and accuracy analysis of chili. This work is a part of our invention in developing an agricultural robot which is able to replace traditional approach including detection, picking and grading.

The limitation is commonly observed in agricultural datasets due to fruit overlap, occlusion, and irregular shapes. Enhancing the input resolution or adopting more advanced architectures, such as YOLOv8 with anchor-free detection heads, may offer improvements in localization precision across varying IoU thresholds. For projection, we will test the image with high resolution to ensure the proposed model able to execute as what we have done in laboratory conditions. However, when it involves bigger sizes of images or videos, features selection might useful to increase the effectiveness and efficiency of the model [29]. We also planning to expand our work to evaluate the detection of fruits variations that not limited to chili with the effect of light intensity distribution [30] and using RGB color intensity [31]. The use of latest version of YOLO models is necessary to expand this work using UAV images [32]. Depth cameras use intensity analysis to calculate an object's distance from a perspective while also giving details on the object's shape, location, classification, and real-world distance. Unlike standard cameras, depth cameras include an additional pixel value that represents the object's distance from the camera. This depth information displays alongside the image. Various depth cameras are able to produce pixels various aspects: red, green, blue, and depth; this achieve by incorporating an RGB color space with a depth system. Utilizing this additional information allows for precise determination of the object's exact location and its distance from the camera, which is particularly useful in processes such as picking or sorting.

REFERENCES

[1] Melissa McDaniel, Santani Teng, Erin Sprout, Hillary Costa, Hillary Hall, Jeff Hunt, Diane Boudreau, Tara Ramroop and Kim Rutledge, "Agriculture is the art and science of cultivating the soil, growing crops and raising livestock, The Art and Science of Agriculture." National Geographic Society, 6 January 6, 2023. Available: https://education.nationalgeographic.org/resource/the-art-and-science-of-agriculture/

[2] The Edge Market. "IR 4.0 Agricultural Technology Can Help Government to Manage Food Security Issues," Ministry of Communication and Digital, 2 June, 2023. Available: https://www.komunikasi.gov.my/en/public/news/

[3] Jean-Jacques Saloman, "What is technology? The issue of its origins and definitions," History and Technology, vol. 1, no. 1984, pp113-156, 1984.

[4] Nawab Khan, Ram L. Ray, Ghulam Raza Sargani, Muhammad Ihtisham, Muhammad Khayyam and Sohaib Ismail, "Current progress and future prospects of agriculture technology: Gateway to sustainable agriculture," Sustainability, vol. 13, no. 9, pp1-31, 2021.

[5] William R. Norris and Albert E. Patterson, "Automation, Autonomy, and Semi Autonomy: A Brief Definition Relative to Robotics and Machine Systems," Essay, Illinois Library, 2019, pp1-3.

[6] Janosch Baltensperger, Pasquale Salza and Harald C. Gall, "Continuous Deep Learning: A Workflow to Bring Models into Production," Software Engineering, Cornell University, 2022.
[7] Mayuree Krajayklang, Andreas Klieber and Peter R Dry, "Colour at harvest and post-harvest behaviour influence paprika and chilli spice quality," Postharvest Biology and Technology, vol. 20, no. 3, pp269-278, 2000.
[8] Olakunle Elijah, Sharul K. A. Rahim, Emmanuel A. Abioye, Muazu Jibrin Musa, Yahaya Otuoze Salihu and Abubakar Abisetu Oremeyi, "Decision Support Platform for Production of Chili using IoT, Cloud Computing, and Machine Learning Approach," 2022 IEEE Nigeria 4th International Conference on Disruptive Technologies for Sustainable Development (NIGERCON), 5-7 April, 2022, pp1-5.
[9] Quoc-Khanh Huynh, Chi-Ngon Nguyen, Hong-Phuc Vo-Nguyen, Phuong Lan Tran-Nguyen, Phan-Hung Le, Dang-Khanh-Linh Le, Van-Cuong Nguyen, "Crack Identification on the Fresh Chilli (Capsicum) Fruit Destemmed System," Journal of Sensors, vol. 2021, pp1-10, 2021.
[10] M.N.Shah Zainudin, Najihah Husin, W. H. Mohd Saad, S. Mohd Radzi. Z.Mohd Noh, N. A. Sulaiman and M. S. J. A. Razak, "A framework for chili fruits maturity estimation using deep convolutional neural network," Przegląd Elektrotechniczny, vol. 97, no. 12, pp77-81, 2021.
[11] M.F. Abdul Aziz, Wan Mohd Bukhari Wan Daud, M.N. Sukhaimie, Tiffany Azhar Izzuddin, M. A. Norasikin, A. F. A. Rasid and N.F. Bazilah, "Development of smart sorting machine using artificial intelligence for chili fertigation industries," Journal of Automation Mobile Robotics and Intelligent Systems, vol. 15, no. 4, pp44-52, 2021.
[12] Gwo-Jiun Horng, Min-Xiang Liu and Chao-Chun Chen, "The smart image recognition mechanism for crop harvesting system in intelligent agriculture," IEEE Sensors Journal, vol. 20, no. 5, pp2766-2781, 2019.
[13] M.N.Shah Zainudin, M.S.S. Shahrul Azlan, L.L. Yin, W.H. Mohd Saad, M.I. Idris, Sufri Muhammad and M.S.J.A. Razak, "Analysis on localization and prediction of depth chili fruits images using YOLOv5, International Journal of Advanced Technology and Engineering Exploration, vol. 9, no. 97, pp1786-1801, 2022.
[14] L.L. Yin, M.N.Shah Zainudin, W.H.Mohd Saad, N.A. Sulaiman, M.I. Idris, M.R. Kamarudin, R. Mohamed and M.S.J.A. Razak, "Analysis recognition of ghost peper and cili-padi using Mask-RCNN and YOLO," Przegląd Elektrotechniczny, vol. 99, no. 8, pp92-97, 2023.
[15] Ying Da Wang and Martin J Blunt, Ryan T Armstrong and Peyman Mostaghimi, "Deep Learning in Pore Scale Imaging and Modeling," Earth-Science Reviews, vol. 215, no. 2021, pp1-32, 2021.
[16] Keiron O'Shea and Ryan Nash, "An introduction to convolutional neural networks, Neural and Evolutionary Computing," Cornell University, 2015, pp1-11.
[17] Wentong Wu, Han Liu, Lingling Li, Yilin Long, Xiadong Wang, Zhuohua Wang, Jinglun Li and Yi Chang Y, "Application of local fully Convolutional Neural Network combined with YOLOv5 algorithm in small target detection of remote sensing image," PLoS One, vol. 16, no. 10, pp1-15, 2021.
[18] Rene Y Choi, Aaron S Coyner, Jayashree Kalpathy-Cramer, Michael F Chiang and J Peter Campbell, "Introduction to Machine Learning, Neural Networks, and Deep Learning," Transl Vis Sci Technol, vol. 9, no. 2:14, pp1-10, 2020.
[19] Joanne Quinn, Joanne McEachen, Michael Fullan, Mag Garder and Max Drummy, "Dive into Deep Learning: Tools for Engagement," SAGE Publications, 2019, pp1-296
[20] Arief Rais Bahtiar, Albertus Joko Santoso, Pranowo and Jujuk Juhariah, "Deep learning detected nutrient deficiency in chili plant," 8th International Conference on Information and Communication Technology (ICoICT), 24-26 June, 2020, pp1-4.
[21] Joseph Redmon, Santosh Divvala, Ross Girshick and Ali Farhadi, "You only look once: Unified, real-time object detection," Proceedings of the IEEE conference on computer vision and pattern recognition, 27-30 June, 2016, pp779-788.
[22] Renjie Xu, Haifeng Lin, Kangjie Lu, Lin Cao and Yunfei Liu, "A forest fire detection system based on ensemble learning," Forests, vol. 12, no. 2, pp1–17, 2021.
[23] Marco Sozzi, Silvia Cantalamessa, Alessia Cogato, Ahmed Kayad and Francesco Marinello, "Automatic Bunch Detection in White Grape Varieties Using YOLOv3," YOLOv4, and YOLOv5 Deep Learning Algorithms, Agronomy, vol. 12, no. 2, pp1-17, 2022.
[24] Mateusz Choiński, Mateusz Rogowski, Piotr Tynecki, Dries P.J. Kuijper, Marcin Churski and Jakub W. Bubnicki, "A First Step Towards Automated Species Recognition from Camera Trap Images of Mammals Using AI in a European Temperate Forest," Lecture Notes in Computer Science, Springer Science and Business Media Deutschland GmbH, vol. 12883, pp299–310, 2021.
[25] Upesh Nepal and Hossein Eslamiat, "Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs," Sensors, vol. 22, no. 2, pp1-15, 2022.
[26] M.N.Shah Zainudin, Md Nasir Sulaiman, Norwati Mustapha and Thinagaran Perumal, "Activity recognition using one-versus-all strategy with relief-f and self-adaptive algorithm", 2018 IEEE Conference on Open Systems, ICOS, 21-22 November, 2018, pp31-36.
[27] Jingwei Zhao, Ye Tao, Zhixian Zhang, Chao Huang and Wenhua Cui, "Lightweight road damage detection network based on YOLOv5," Engineering Letters, vol. 32, no. 8, pp1708-1720, 2024.
[28] Ruiqing Shan, Xiaoxia Zhang and Shicheng Li, "A method of pneumonia detection based on an improved YOLOv5s," Engineering Letters, vol. 32, no. 6, pp1243-1254, 2024.
[29] Mohd Muzafar Ismail and Muhammad Noorazlan Shah Zainudin, "Numerical method approaches in optical waveguide modeling," Applied Mechanics and Materials, vol. 52-54, pp2133-2137, 2021.
[30] Eva Darko, Kamiran A. Hamov, Tihana Marcek, Mihaly Dernovics, Mohamed Ahres and Gabor Galiba, "Modulated light dependence of growth, flowering and the accumulation of secondary metabolities in chili," Front Plant Sci, 2022; vol. 13, no. 2021, pp1-15, 2022.
[31] Sri Wahjuni, Wulandari and Husna Nurafifah, "Faster RCNN based leaf segmentation using stereo images," Journal of Agriculture and Food Research, vol. 11, no. 100514, pp1-7.
[32] Huikai Li and Jie Wu. "LSOD-YOLOv8s: A lightweight small object detection model based on YOLOv8 for UAV aerial images", Engineering Letters, vol. 32, no. 11, pp2073-2082, 2024.

Muhammad Noorazlan Shah Zainudin is a senior lecturer at Universiti Teknikal Malaysia Melaka. He received his bachelor's degree, master's degree in Computer Science from Universiti Teknologi Malaysia and doctorate in Intelligent Computing from Universiti Putra Malaysia. His current research interest includes Artificial Intelligence, Data Mining, Pattern Recognition and Machine Learning.

Muhammad Afiq Farhan Ibrahim is an undergraduate student from Universiti Teknikal Malaysia Melaka. He received his bachelor's degree in Electronic Engineering from Universiti Teknikal Malaysia Melaka.

Nurul Zarirah Nizam is a senior lecturer at Universiti Teknikal Malaysia Melaka. She received her bachelor's in Marketing and master degree in Management from Universiti Malaysia Terengganu. She received her doctorate (PhD) in Business Administration from Aichi University, Japan. Her current research interest includes Green marketing, Consumer behavior, Supply chain and logistics.

Wira Hidayat Mohd Saad is an associate professor at Universiti Teknikal Malaysia Melaka. He received his bachelor's degree in Electrical and Electronic Engineering and doctorate (PhD) in Multimedia System Engineering from Universiti Putra Malaysia. His current research interest includes Medical signal and Image processing, Embedded artificial intelligence and Deep learning in computer vision.

Muhammad Raihaan Kamarudin is a lecturer at Universiti Teknikal Malaysia Melaka. He received her bachelor's and master degree in Electronic Engineering from Takushoku University, Japan. Currently his pursuing his doctorate (PhD) in neuroscience and brain modelling. His current research interest includes Brain Modelling, Neuromorphic Devices and Neural Network Applications.

Muhammad Idzdihar Idris is a senior lecturer at Universiti Teknikal Malaysia Melaka. He received her bachelor's degree in Electronic System Engineering from Hiroshima University, Japan, master's degree in microelectronics from Universiti Kebangsaan Malaysia and doctorate (PhD) in Semiconductor Devices from Newcastle University, UK. His current research interest includes Fabrication of Semiconductor Devises (solar cell, transistor, MOS capacitor, diode) and IC Design.

Mohd Safirin Karis is a senior lecturer at Universiti Teknikal Malaysia Melaka. He received her bachelor's in Electronics and master degree in Control from Universiti Teknologi Malaysia. His current research interest includes Control system and Neural network.

Sufri Muhammad is a senior lecturer at Universiti Putra Malaysia. He received her bachelor's degree, master's degree and doctorate (PhD) in Computer Science from Universiti Putra Malaysia. His current research interest includes Service-Oriented Architecture, Service Engineering, Semantic-Based Approach and Context-Aware Mobile Cloud Learning.

Raihani Mohamed is a senior lecturer at Universiti Putra Malaysia. She received bachelor's degree from International Islamic University Malaysia, a master's degree from Universiti Teknologi MARA, and obtained her doctorate (PhD) in Computer Science from Universiti Putra Malaysia. Her research interests include smart home systems, ambient assisted living environment, machine learning, data science, and big data analytics.