International Research Journal of Advanced Engineering and Science
ISSN (Online): 2455-9024

Building a Fruit Identification System Model on a Conveyor Belt Using Image Processing Techniques

Vu Huy Trung*, Nguyen Hong Quan, Nguyen Xuan Thien
Academy of Military Science and Technology, Hanoi, Vietnam

Abstract—*This article presents the results of building the fruit identification and classification system (ICS) model on a conveyor belt based on image processing techniques. First, the mechanical and control system model of the ICS is designed. The system is capable of counting and evaluating the quality of fruit moving on the conveyor belt through proximity sensors and image processing techniques. The YOLOv3 image processing algorithm is used to identify defects on fruits and their color. The Raspberry Kit central controller, Arduino Nano circuit and PiV2 Camera are used to integrate motor control, receive image processing signals and signals from sensors such as weight sensors and proximity sensors. Results of fruit identification, counting and quality are displayed on the LCD and HDMI screens. The control algorithm of the entire system is divided into specific working cycles. The classification system model is implemented in practice and operates effectively according to the original design. The results of this research have important implications for the practical use of highly efficient image processing techniques and integrated control techniques of multiple devices in an industrial system.*

Keywords— *Classification system, fruit, identification, image processing*

I. Introduction

Today, society is increasingly developing with increasing levels of industrialization and modernization to improve life. Therefore, the application of science and technology is becoming more and more widespread, popular and highly effective in most economic and technical fields as well as in social life. In the era of the 4.0 industrial revolution, more and more modern equipment is used to automatically control production and product processing processes. This leads to the formation of flexible production systems that allow high levels of automation based on the use of CNC machines and industrial robots. Automatic sorting systems are also an important element in those industrial production processes [1]. This is a system used to classify items in a warehouse or distribution center based on their destination. Random streams are sorted into orders for shipping. Product placement and delivery parameters depend on the type of company or application in the supply chain. The automatic classification system helps completely replace humans in product classification with the highest accuracy [2], [3]. In contrast, traditional product classification methods require a large workspace and require a large number of workers to participate in classification. This often leads to increased classification time costs and frequent errors. Initially, sorting systems using barcodes were quite popular [4]. They are applied to classify mail, parcels, and finished products packed into cartons, bags, and affix barcodes and QR codes. This system has a capacity of up to 10,000 products/hour [5]. Production efficiency is 3-5 times higher than traditional sorting methods. Next, a product classification system by volume was also developed. This is a product sizing system applied to many different types of products based on the principle of direct weight checking, then divided into weight groups according to requirements [6], [7]. Along with the development of science, based on product characteristics, a classification system based on product size and color is also researched and developed [8], [9]. This system automatically sorts not only industrial goods but also foods such as fruits and vegetables on the basis of their size and color because this criterion is often considered mainly before processing. customers choose and buy products [10]. A flexible tactile sensor was developed and presented in [11] combined with artificial intelligence algorithms allowing fruit identification and classification. A swarm algorithm combined with deep learning techniques (AFC-ETSAFDL) is described in [12] to recognize different types of fruits from captured images. With the increasingly strong development of computational tools and algorithms, image recognition techniques are also of interest in research and application in automatic classification systems. Foreign fibers in cotton seeds were detected and identified in [13] based on the improved YOLOv7 model during cotton processing. The ripeness of tomatoes was identified in [14] through the MTD-YOLOv7 algorithm with the combination of deep learning convolutional neural network and YOLOv7 model. Again, the YOLO algorithm is also used to identify flowers and fruits on strawberry plants and is presented in [15]. Accordingly, the algorithm is deployed to identify very small objects in real time thanks to improvements in image resolution and quality. The automatic fruit loading system presented in [16] used the YOLOv5 model to recognize stems and sepals in real time for fruit packaging and status adjustment. The optimization algorithm is also applied to reduce the computational load for the inputs being the posture of the fruit. The model works effectively and meets practical requirements. The improved YOLOv3 model described in [17] is used to detect apple growth in orchards as a basis for improving productivity. Size, color, cluster density and a number of other apple characteristics are considered in evaluating growth stages. It is clear that most research focuses mainly on recognition and classification algorithms. Meanwhile, the overall system including mechanical and control components is hardly mentioned. Most of these classification systems have a high degree of automation and high accuracy. They operate stably and are suitable for many different types of products. However, designing and fully integrating the system is always a big challenge with the combination of many multidisciplinary problems.

This article presents the building of a model of an automatic tomato sorting system. This system has the function of dividing tomatoes with the same or different attribute groups together for packaging or eliminating objects that do not meet production requirements. In particular, the problem of identification and classification based on image processing algorithms is specifically described to ensure the functionality of the system. The YOLOv3 algorithm is used for image recognition. A small concept model is designed and fabricated to evaluate the system's ability to complete functions. This process includes main tasks such as designing a mechanical structure that meets the working capacity and stability of the system. After that, elements such as cameras and Raspberry kits are used in image processing, fruit classification and signal transmission to the Arduino control circuit.

II. Material and Method

A. Classification system designing

The fruit classification system is designed to ensure main functions including: transporting products on conveyor belts; check the quality and classify fruits (pass or fail); count products; displays the quantity and weight of the product on the LCD screen. Thus, the system includes the following specific components:

Mechanical components and structures include:

(1) Camera support frame: ensures the camera works best

(2) Tomato supply system: this is the place to store and move tomatoes into the grading system

(3) Conveyor system: performs the task of transporting tomatoes to the product tray.

(4). Tomato dividing system: evenly divide the number of tomatoes into the finished box

(5) and (6) Box supply system: delivers empty boxes to the location where tomatoes are sorted.

(7) Weighing table: weighs the mass of tomatoes on the tray.

The system model is described in Fig. 1.

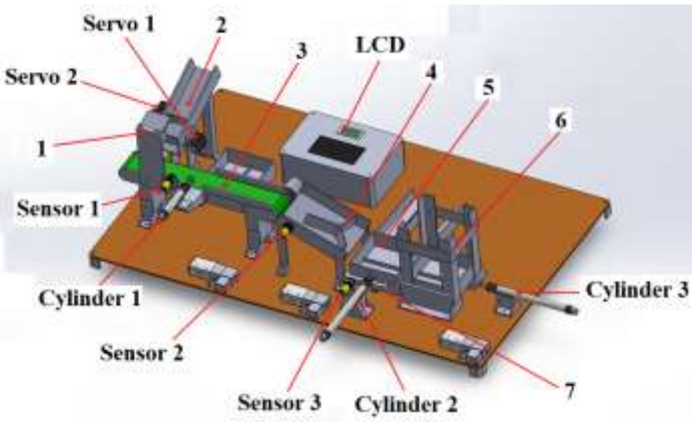


Fig. 1. Classification system model

Drive system and sensors include:

Servo motors: supply tomatoes to the conveyor belt

Pneumatic cylinder 1: pushes green tomatoes (fail) from the conveyor belt to the container.

Pneumatic cylinder 2: push the red tomato container after there are enough tomatoes on it

Pneumatic cylinder 3: supplies the tomato box to the scale position

Sensor 1: detects green tomatoes (disqualified)

Sensor 2: detects red tomatoes (satisfactory)

Sensor 3: detects that the container is already on the scale

Components of the control system include:

Microcontroller: Raspberry and Arduino control the system actuators.

Sensors: ensure object identification and detection

Camera: performs the task of image processing and tomato type identification

Loadcell sensor: determines tomato weight

B. Implement image recognition algorithm

1) Prepare input data

To prepare the input data set to extract tomato features, the criteria for classifying and checking tomato quality are determined as follows:

Criteria for classifying tomatoes: based on the color of tomatoes, including: red corresponds to ripe fruit (qualified) and green corresponds to young fruit (disqualified).

Criteria for checking tomato quality: this is a difficult and complex criterion. To control fruit quality, it is possible to check based on many different signs such as: looking for scratches and insect damage on the fruit skin; The degree of crushing or severe distortion on the fruit body; Check for blue spots and black streaks in the skin... In this article, large, prominent black spots on fruit bodies are used as quality control criteria.

From the above two criteria, about 1,000 images are built corresponding to the four groups of fruits that need to be classified (red; red-failed; green; green-failed). The results of collecting input images are described in Fig. 2.

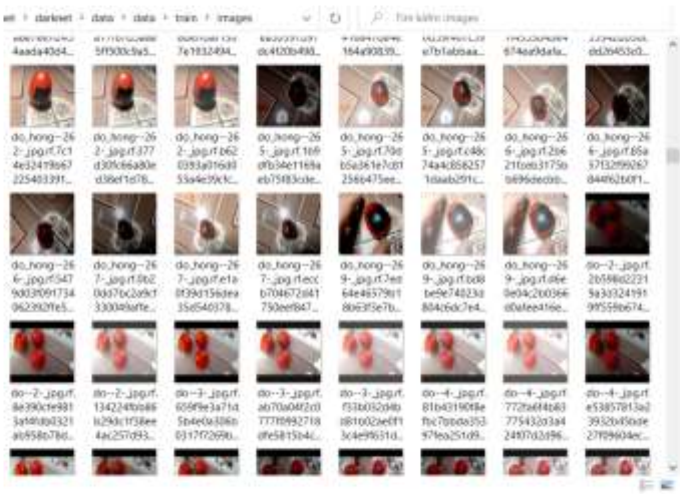


Fig. 2. Original image data

Vu Huy Trung, Nguyen Hong Quan, Nguyen Xuan Thien, “Building a fruit identification system model on a conveyor belt using image processing techniques,” *International Research Journal of Advanced Engineering and Science*, Volume 9, Issue 1, pp. 158-163, 2024.

After preparing the input data, the roboflow framework module is used to support labeling each fruit that needs identification (Fig. 3).

Fig. 3. Labeled input data

To test the recognition ability of the training model, the initial data file is divided into three parts with the following ratios: Training (70% with 691 images), Test (10% with 97 images), Validation (20% with 194 photos). With the purpose of enhancing the quality of input images for the recognition system, a number of integrated image processing techniques are used including image saturation, exposuring and image brighness. Adjust brightness to increase object details, and blur images in necessary cases (Fig. 4). Through the above methods, the input data file is fine-tuned to keep the level of accuracy and similarity to images obtained through the camera in reality.

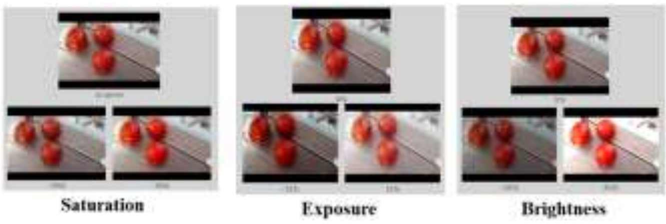


Fig. 4. Some input data processing techniques

2) Model training

With large input data, training a neural network model for the purpose of recognizing any object on a low-configuration device is very difficult. The training process is time consuming and training cannot be sustained over a long period of time. Therefore, the GoogleColab support tool is used in this case. The steps to train the model on GoogleColab are as follows:

Step 1: Import the Pytorch library that supports training.

Step 2: Reconstruct the YOLOv3 network structure

Step 3: Import the dataset after labeling and preprocessing the images via Roboflow.

Step 4: Prepare the image data file and labeled labels

Step 5: Convert labeled layers to named layers (red, red-failed, green, green -failed).

Step 6: Conduct model training

Fig. 5. Pytorch library

After training for 300 Epochs, the P (Precision) and R (Recall) indexes both reached a relatively high level of ~0.9 and the mAP (Average Precision) index ~0.97 (Fig. 6).

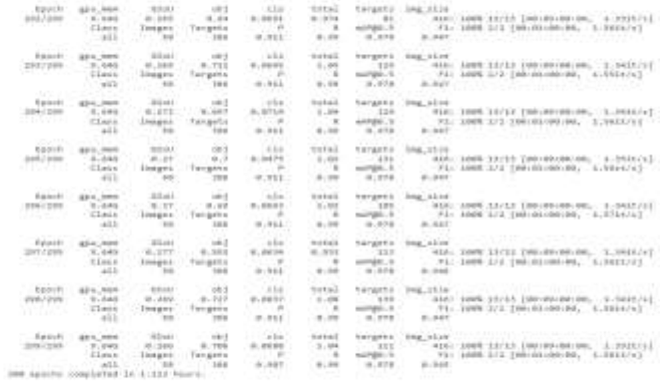


Fig. 6. Results after training

Fig. 7 shows the loss value after training and validation gradually decreasing over epochs. The above indexes are almost equal to 0 at epoch 300. The P and R indexes both reach a relatively high level of ~ 1.0 and the mAP index ~ 1.0. This proves that the training process has quite high accuracy.

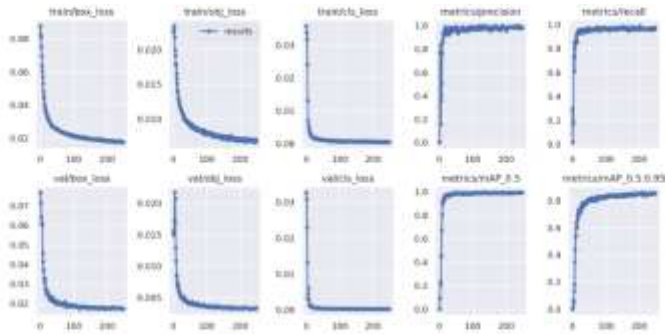


Fig. 7. Graph representing parameters after training

With an input image of 4 classes of objects to be identified, the results of evaluating the accuracy of each class are displayed as shown in Fig. 8.

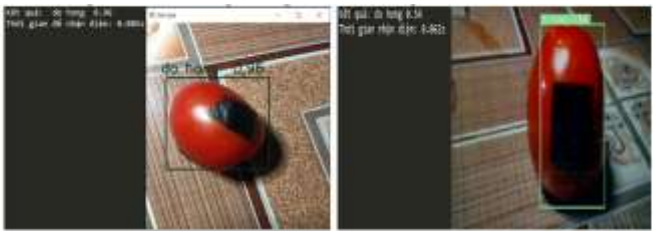
Fig. 8. Accuracy of each class

However, to accurately test the training time and recognition ability of the YOLOv3 network, the training process of the YOLOv3 network model is compared with the Darknet network model on the same input data file. The

International Research Journal of Advanced Engineering and Science
ISSN (Online): 2455-9024

results obtained are as shown in Fig. 9, Fig. 10, Fig. 11 and Fig. 12.

YOLOv3 (P=0.99; Time = 0.089s); Darknet (P=0.59; Time = 0.06s)
Fig. 9. Figure 9. Identification of red tomatoes



YOLOv3 (P=0.96; Time = 0.085s); Darknet (P=0.54; Time = 0.062s)
Fig. 10. Identification of red-failed tomatoes

YOLOv3 (P=0.95; Time = 0.077s); Darknet (P=0.74; Time = 0.058s)
Fig. 11. Identification of green tomatoes

YOLOv3 (P=0.98; Time = 0.074s); Darknet (P=0.7; Time = 0.06s)
Fig. 12. Identification of green-failed tomatoes

Accordingly, the comparison results show the difference between the two identification models YOLOv3 and Darknet. With the same still image input, the recognition time is not too different (~ 0.02 seconds), but YOLOv3 offers higher accuracy than Darknet (YOLOv3 from 95-99%, Darknet from 50-70%). Thus, using the YOLOv3 model to identify and process images is appropriate within the scope of this research.

C. Integrate image processing techniques in the classification system

The classification system is implemented as depicted in Fig. 13.

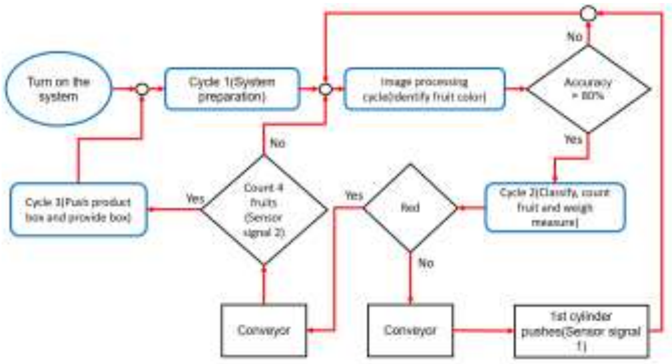


Fig. 13. General operating diagram of the system

Accordingly, the system is initially powered to operate. Cycle 1 is performed first with activities including the provision of tomato boxes. At this time, the conveyor belt, motors and sensors all work.

Next, the system enters the image processing cycle (Fig.14) after the tomato is placed in the recognition position. The tomato recognition accuracy was determined. If the accuracy value is less than 80%, the algorithm will ask to return to the identification step. On the contrary, if the recognition accuracy is higher than 80%, the system moves to Cycle 2. Fig. 15 shows the product classification algorithm based on image processing.

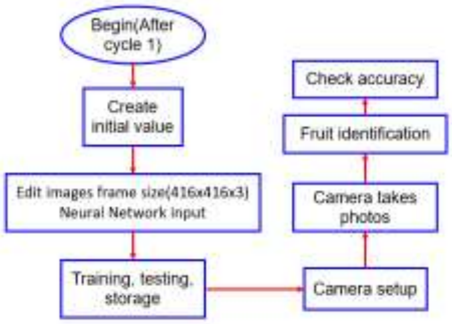


Fig. 14. Image processing flow chart (after Cycle 1)

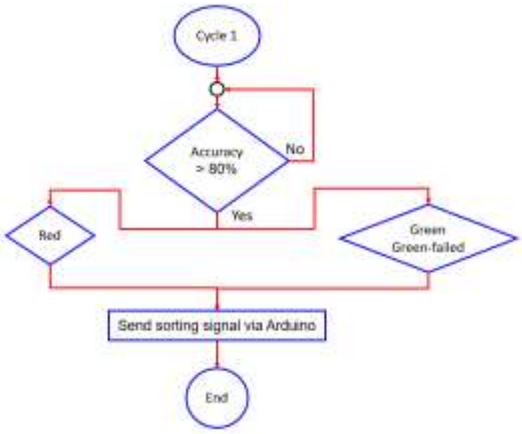


Fig. 15. Diagram of product classification implementation

In Cycle 2, the tomatoes, after being identified, will begin to pass through the conveyor belt to the sorting position. If it is not a red fruit (i.e. it is green or green-failed) then it will be pushed by Cylinder 1 via the signal of sensor 1. If it is a red fruit then it will be passed through sensor 2 to the box contains red fruit. At this time, the counting and weighing sensor works. When there are not enough 4 fruits in the container, Cycle 1 and Image Processing Cycle are performed again. Conversely, when the box contains 4 fruits, Cycle 3 is performed.

In cycle 3, the box containing the product with enough quantity of 4 fruits will be pushed out, and a new box will be supplied to the scale position. Cycle 1 is continued again.

D. Test results and evaluation

The actual model is manufactured based on the corresponding materials of the system. For the mechanical system, the conveyor belt width is designed to be 60mm, suitable for tomato size (about 30~50mm). The model is designed with two additional partitions on both sides of the conveyor belt to prevent products from falling out during transportation. The conveyor belt design material chosen is rough PVC to increase friction with the fruit, thereby not making the fruit slippery when moving at high speed. The speed of the conveyor belt is maintained stable at 0.1m/s. However, conveyor speed can be customized at many different levels to increase performance in necessary cases. The control system is designed and manufactured using the Raspberry Pi 3B+ circuit and Arduino NANO Atmega328P. During system operation, Raspberry Pi3B+ and Arduino NANO continuously transmit and receive data. The UART transmission type was chosen because both control kits have this transmission standard built-in. Therefore, the setup process is quite easy. On the other hand, the UART communication standard has fast and stable transmission and reception speeds, meeting the system's requirements well.

Fig. 16. Raspberry Pi 3B+ and Arduino NANO Atmega328P circuit

The Pi V2 camera is used and has the function of collecting actual image signals and sending data to the Raspberry Pi 3B+ block. Loadcell YZC-133 is used to convert the load or force applied to it into an electrical signal. This electrical signal can be a change in voltage, current or frequency depending on the load cell type and the corresponding measuring circuit.

Fig. 17. Hardware elements of the control system

Module HX711 is used to amplify signals from Loadcell to Arduino. The system is integrated with the E18-D80NK proximity sensor. This sensor uses infrared light to determine the distance to obstacles with fast response and very little noise due to the use of eyes that receive and emit infrared rays at separate frequencies. It can adjust the desired alarm distance via a potentiometer. The sensor output is in open collector form, so it is necessary to add 1 or more to the source at the signal pin when in use. The 5-inch LCD screen is used with a hardware resolution of 800×480, configurable by software up to 1920×1080. This type of LCD supports popular Mini PCs such as Raspberry Pi B3+, Jetson Nano, BB Black, Banana Pi, as well as general computers. Furthermore, a multi-language OSD menu is integrated for power management, brightness/contrast adjustment. On the other hand, the LCD has a 3.5mm audio jack, speaker connector, HDMI audio output support, and VGA input support. MG996R Servo motor, L298 DC motor control circuit, pneumatic cylinder and 5/2 AIRTAC solenoid valve are selected.

The actual model works and is depicted in Fig. 18.

Fig. 18. The actual tomato classification system model

The classification results based on the image processing model are shown in Fig. 19.

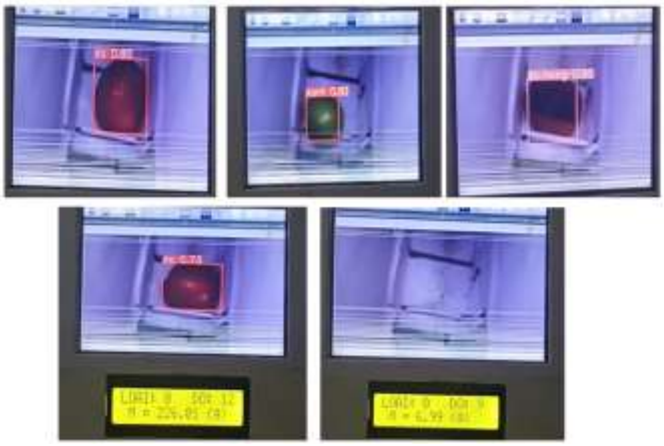


Fig. 19. Results of recognition and display on the real model

For the classification process, based on the trained model, the identified object whose features are extracted and compared with the input data. The classification results are displayed on the system screen. Specifically, if the tomato is red, the screen displays "Do 0.80", in which the value 0.80 is the level of confidence with which the camera recognizes the red fruit. If the detected tomato is green, the screen displays “Xanh 0.82”, where the value 0.82 is the level of confidence with which the camera recognizes the green fruit.

For the fruit quality inspection process, the tomato quality inspection system is based on the presence or absence of black streaks on the fruit body. If the red fruit has black streaks, the screen displays “Do hong 0.80”. In which the value 0.80 is the level of confidence with which the camera recognizes the damaged red fruit. On the contrary, if the green fruit has a defect, the screen displays “xanh hong 0.73”, with the value 0.73 being the level of confidence with which the camera recognizes the damaged green fruit.

After sorting and checking the quality of the fruit, those that meet the standards will be brought to the end of the conveyor belt and stored in boxes. For each box of 4 fruits, the Loadcell system will weigh the weight and display it on the LCD screen. So in reality, the system functions properly, the identification, classification and quality assessment modules all work well.

III. CONCLUSIONS

In general, the concept model of the tomato identification and classification system based on the YOLOv3 image recognition algorithm has been deployed and works effectively in practice. The main components of the system include mechanical devices, controls and sensors that are integrated to complete the overall system that has been implemented. Based on a large enough input data set and carefully processed with the YOLOv3 recognition model, the problem of identifying and classifying tomatoes as pass or fail has been solved effectively. Proximity and weight sensors allow counting and provide full information displayed on HDMI and LCD screens. The results of system building and implementation of identification and classification algorithms in this study are the basis for being able to deploy the system in practice at an industrial level.

REFERENCES

[1] C. Soliman, B. C. Thomas, G. Giannarini, Evolution and Implications of the Novel CAMUS Reporting and Classification System: From Rationale to End Product, Europan Urology Open Scienc, 123-126, 2023.
[2] W. L. Dai, D. C. Li, Applying systematic diagnosis and product classification approaches to solve multiple products operational issues in shop-floor integration systems, Expert Systems with Applications 37, 6373–6380, 2010.
[3] T. Kremer, N.J. Rowan, G. McDonnell, A proposed cleaning classification system for reusable medical devices to complement the Spaulding classification, Journal of Hospital Infection 145, 88-98, 2024.
[4] K. M. Gallaghera, M. Jensen, M. Paynea, R. Towne, An imperceptible barcode can reduce the muscle activity required to scan common consumer packaged goods, International Journal of Industrial Ergonomics 72, 80–85, 2019.
[5] T. Rodriguez, D. Haaga, D, S. Calhoon, Automation and workflow considerations for embedding Digimarc Barcodes at scale. IS&T/SPIE Electronic Imaging, 9409, 940905–940914, 2015.
[6] A. J. P. Cassimiroa, E. S. Ramosa, V. E. Cabrerab, E. N. A. Freitas, Milk Weighing Scale based on Machine Learning, Preprint submitted to Smart Agricultural Technology, 1-18, 2024.
[7] S. Kaunkid, A. Aurasopon, A. Chantiratiku, Automatic milk quantity recording system for small-scale dairy 900 farms based on internet of things, Agriculture 12, 1877, 2022.
[8] J. C. Miranda a, J. G. Mola, Fruit sizing using AI: A review of methods and challenges, Postharvest Biology and Technology 206, 112587, 2023.
[9] H. X. Huynh, B. H. Lam, H. V. C. Le, T. T. T. Le, N. D. Trung, Design of an IoT ultrasonic-vision based system for automatic fruit sorting utilizing size and color, Internet of Things 25, 101017, 2024.
[10] C. Hampson, K. Sanford, J. Cline, Preferences of Canadian consumers for apple fruit size. Can. J. Plant Sci, 82 (1), 165–167, 2022.
[11] Y. Wei, L. Cai, H. Fang , H. Chen, Fruit recognition and classification based on tactile information of flexible hand, Sensors & Actuators: A. Physical 370, 115224, 2024.
[12] A. H. Alharbi, S. Alkhalaf, Y. Asiri, S. Abdel-KhalekR. F. Mansour, Automated Fruit Classification using Enhanced Tunicate Swarm Algorithm with Fusion based Deep Learning, Computers and Electrical Engineering 108, 108657, 2023.
[13] Q. Li, W. Ma, H. Li, X. Zhang, R. Zhang, W. Zhou, Cotton-YOLO: Improved YOLOV7 for rapid detection of foreign fibers in seed cotton, Computers and Electronics in Agriculture, 219, 108752, 2024.
[14] W. Chen, M. Liu, C. J. Zhao, X. Li, Y. Wang, MTD-YOLO: Multi-task deep convolutional neural network for cherry tomato fruit bunch maturity detection, Computers and Electronics in Agriculture 216, 108533, 2024.
[15] Y. Bai, J. Yu, S. Yang, J. Ning, An improved YOLO algorithm for detecting flowers and fruits on strawberry seedlings, Biosystems Engineering 237, 1–12, 2024.
[16] Z. Wang, L. Jin, S. Wang, H. Xu, Apple stem/calyx real-time recognition using YOLO-v5 algorithm for fruit automatic loading system, Postharvest Biology and Technology 185, 111808, 2022.
[17] Y. Tian, G. Yang, Z. Wang, H. Wang, E. Li, Z. Liang, Apple detection during different growth stages in orchards using the improved YOLO-V3 model, Computers and Electronics in Agriculture 157, 417–426, 2019.