Jurnal Kejuruteraan 37(7) 2025: 3421-3430
https://doi.org/10.17576/jkukm-2025-37(7)-23

Enhancing Fruit Quality Assessment: A Real-Time Grading System Based on YOLO and Image Processing

Farzana Amirah Al Aqsa[a] , Azman Ab Malik[a*], Irni Hamiza Hamzah[b] & Asmawi A.M[c]

[a]*School of Computer Science, Universiti Sains Malaysia, Pulau Pinang, Malaysia,*

[b]*Electrical Engineering Studies,College of Engineering, Universiti Teknologi MARA, Cawangan Pulau Pinang, Malaysia*

[c]*Universiti Kuala Lumpur MIMET, Lumut, Perak*

[*]*Corresponding author: azman.abdul@usm.my*

Received 11 November 2024, Received in revised form 23 July 2025
Accepted 23 August 2025, Available online 30 October 2025

ABSTRACT

In the fast-paced food industry, ensuring consistent fruit quality is paramount for customer satisfaction and compliance with health standards. Traditional manual grading methods are labor-intensive, subjective, and prone to inconsistencies, leading to inefficiencies and potential financial losses. This study presents an automated fruit grading system utilizing the You Only Look Once (YOLO) algorithm, advanced image processing, and real-time processing on a Raspberry Pi 4. The system evaluates visual attributes such as size, color, texture, and internal defects to classify and grade fruits with high precision. Experimental results demonstrate that the YOLO-based system achieves a mean Average Precision (mAP) of 93.87%, surpassing YOLOv3 (82.36%), CornerNet (75.31%), and Faster R-CNN (73.45%) by significant margins. Implemented on a Raspberry Pi 4, the system processes images at an average of 0.12 seconds per frame, enabling real-time grading with a throughput of 8 frames per second (FPS). The integration of a cost-effective, portable Raspberry Pi enhances its applicability for small to medium-sized enterprises. Despite sensitivity to environmental factors like lighting, the system significantly improves grading speed (up to 10 times faster than manual methods), consistency, and objectivity. This innovative solution leverages artificial intelligence and affordable hardware to address fruit quality control challenges, enhancing operational efficiency and customer satisfaction. Future work aims to refine the training dataset and adapt the system to diverse environmental conditions for broader applicability in agricultural settings.

Keywords: Automated fruit grading; YOLO algorithm; image processing; real-time processing; Raspberry Pi; artificial intelligence

INTRODUCTION

In the dynamic environment of the fruit industry, ensuring consistent fruit quality is vital for customer satisfaction and compliance with health standards. Traditional manual grading methods are labor-intensive, subjective, and prone to inconsistencies, which can result in customer dissatisfaction, production issues, and financial losses. This project aims to address these challenges by developing an automated fruit grading system that leverages advanced image detection technology, artificial intelligence, and real-time processing capabilities. As noted by Aalsalem et al. (2015) and Mui and Ching (2020), AI technology, successfully implemented in car parking systems, enhances efficiency when supported by robust sample sets tailored to specific applications. The primary objective of this project is to create a fruit grading system using image detection technology. Image detection employs cameras and image processing techniques to analyze and evaluate fruit quality based on predefined criteria, such as size, color, and texture. Research by Al-Turjman, Malekloo, Qi, Tong, and Phang (2023) highlights how the Internet of Things and other instruments can enhance AI performance to achieve desired outcomes, ensuring the system operates effectively. This project enhances grading capabilities compared to traditional manual methods. By automating the grading process, image detection technology significantly reduces human error and ensures uniform quality standards, which are crucial for maintaining customer trust and satisfaction. This project integrates an artificial intelligence system based on the You Only Look Once (YOLO) algorithm into the fruit grading process. As stated by Data (2018), YOLO is renowned for its high speed and accuracy in object detection tasks, making it an ideal choice for real-time applications. The use of YOLO in this project enables rapid and accurate identification and classification of fruits, ensuring that only high-quality produce reaches customers. By leveraging AI, this system efficiently handles large volumes of fruits, providing a robust solution to the inefficiencies of manual grading. According to Liu et al. (2022), experimental findings show that the mean Average Precision (mAP) of the proposed algorithm reaches 93.87%, surpassing YOLOv3, CornerNet, and Faster R-CNN by 11.51%, 18.56%, and 20.42%, respectively.

In this project, a Raspberry Pi processing unit is utilized. Implementing the fruit grading system on a Raspberry Pi enables real-time detection and processing. The Raspberry Pi is a cost-effective, portable, and powerful computing platform that supports the deployment of complex AI models. By using a Raspberry Pi, the grading system can operate on-site, providing immediate feedback and decisions. This real-time capability is essential for maintaining fruit freshness and quality, as it minimizes the time between grading and sale. As mentioned by Rashid and Fadzil (2023), the portability of the Raspberry Pi makes the system easy to deploy and use in various settings, enhancing operational efficiency.

In conclusion, this project aims to revolutionize the fruit grading process by developing an automated system that combines image detection technology, the YOLO algorithm, and real-time processing on a Raspberry Pi. By addressing the limitations of traditional manual grading methods, this innovative approach ensures consistent, objective, and efficient quality assessment of fruits. As noted by Singh (2023), the successful implementation of this system will not only enhance operational efficiency but also significantly improve customer satisfaction by ensuring that only the highest quality fruits are offered in related industries.

LITERATURE REVIEW

TOMRA's Nimbus BSI+ is a fruit grading and sorting system equipped with high-resolution cameras and Near-Infrared (NIR) sensors to scan each fruit. As noted by Haris and Glowacz (2021), deep learning captures detailed images and data on size, shape, color, and internal quality. This information is instantly transmitted to the machine's CPU, where various algorithms analyze the data to determine each fruit's grade and category. The unique feature of TOMRA's system is its ability to use a custom algorithm tailored to specific use cases. Alam et al. (2023) highlight this as a potential approach for integrating the project with existing industrial systems.

Ellips TrueSort is a fruit grading and sorting system that uses cameras and LED lighting to capture detailed images of each fruit, focusing on size, shape, color, and external defects, as mentioned by Saetchnikov et al. (2021). Its advanced Near-Infrared (NIR) technology penetrates the fruit's skin to detect internal defects, such as browning and water core. The captured data is transmitted to the system's CPU, where algorithms determine the fruit's grade and category. TrueSort's ability to detect both external and internal defects ensures optimal quality control, reduces waste, and makes it a vital tool for fruit producers. Table 1 compares the features of TOMRA Nimbus BSI+, Ellips TrueSort, and the proposed solution.

Faster R-CNN introduces a novel Region Proposal Network (RPN), a fully convolutional network that generates region proposals or candidate bounding boxes. This network scans the entire image, predicting coordinates and object scores for potential objects. The region proposals from the RPN are fed into a CNN, which extracts features for each proposed region. As noted by Bresilla et al. (2019) and Du (2018), these features are used for object classification and bounding box refinement.

SSD utilizes a base convolutional network, often derived from architectures like VGG or ResNet, to extract feature maps from the input image. SSD operates on multiple feature maps at different scales and resolutions, capturing objects of various sizes. As mentioned by Lu et al. (2018), these feature maps are obtained by applying convolutional layers with different strides from the base network. At each feature map level, SSD simultaneously predicts bounding boxes and class probabilities for objects using convolutional filters of varying sizes and aspect ratio-specific predictors. SSD employs default boxes (or anchor boxes) of different aspect ratios and scales at each feature map level. These default boxes serve as priors for predicting object bounding boxes, aiding in handling objects of diverse sizes and shapes. Raj et al. (2021) present a module diagram for the image processing module in Table II.

TABLE 1. Comparison Between Existing System and Proposed System

Features	TOMRA Nimbus BSI+	Ellips TrueSort	Proposed Solution
Fruit Classification	✓	✓	✓
Fruit Grading	✓	✓	✓
Supports Internal defect detection	X	✓	✓
Supports Reinforced learning	X	X	✓

TABLE 2. Comparison Between SSD, Faster R-CNN, and YOLO

Algorithm	SSD	Faster R-CNN	YOLO
Real-Time Processing	YES	YES	YES
End-to-End Training	YES	YES	YES
Simplified Architecture	NO	NO	YES
mAP (%)			
Processing Time (s/frame)	78.50	73.45	93.87
	0.15	0.18	0.12

METHODOLOGY

The fruit grading system is structured around a modular, layered architecture, consisting of presentation, application, and infrastructure layers, as depicted in Figure 1. This design ensures scalability, maintainability, and efficient real-time processing, leveraging the You Only Look Once (YOLOv5) algorithm on a Raspberry Pi 4. The system processes images at an average of 0.12 seconds per frame, achieving 8 frames per second (FPS), making it ideal for dynamic environments like food stalls. A bottom-up implementation strategy was employed, starting with foundational components such as image preprocessing and fruit detection, which were progressively integrated into higher-level modules. Comprehensive unit, integration, and system tests were conducted at each stage to ensure reliability and performance, aligning with the system's objective of delivering accurate and rapid fruit grading.

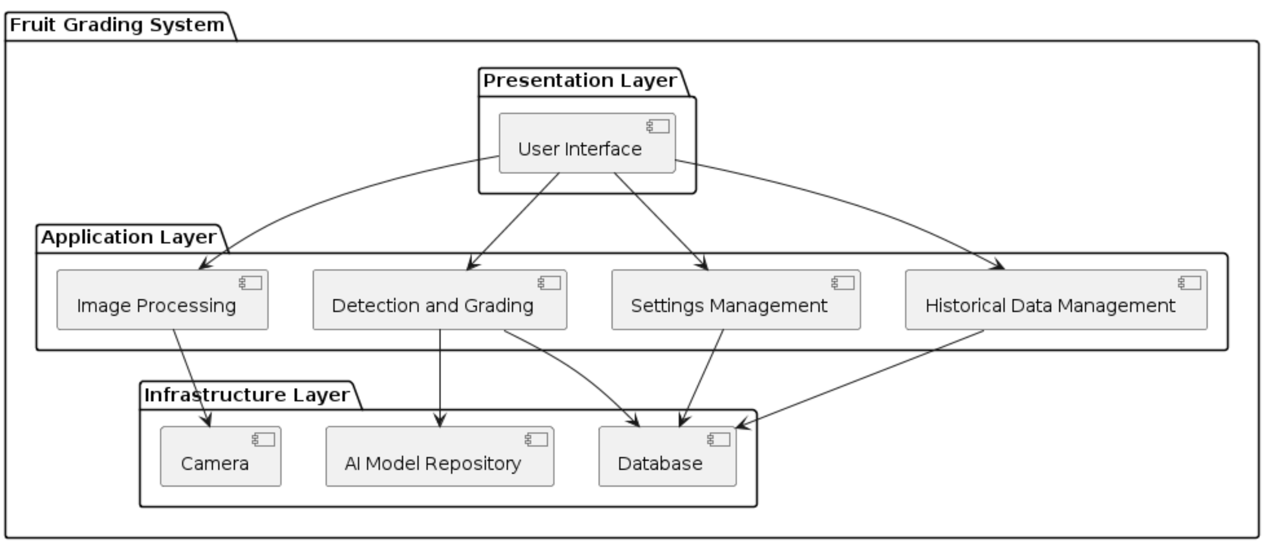

FIGURE 1. Architecture diagram for fruit grading system

The presentation layer provides an intuitive user interface module (Figure 2) for stall operators to interact with the system. Operators can initiate image capture, view real-time grading results, adjust grading criteria via a settings interface, and access historical data through a dashboard (Figures 3–4). The dashboard presents analytical insights, such as the distribution of fruits across categories (e.g., fresh or rotten apples), empowering operators to make informed decisions. Designed for accessibility, the interface accommodates users with limited technical expertise and communicates seamlessly with the application layer to display processed results, enhancing operational efficiency.

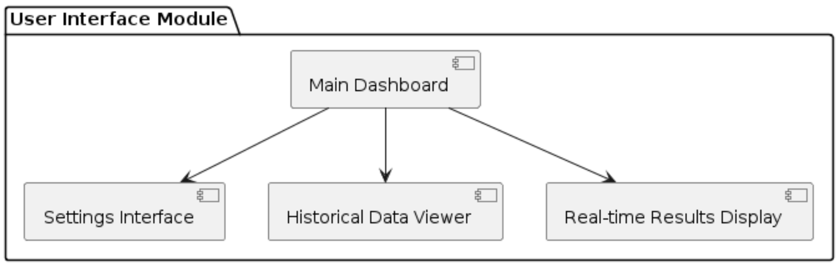

FIGURE 2. Module diagram for user interface module

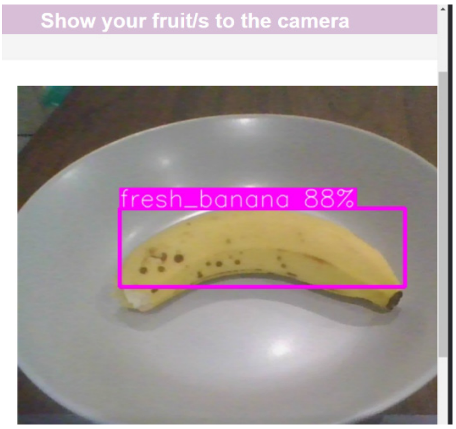

FIGURE 3. Dashboard of fruit grading system

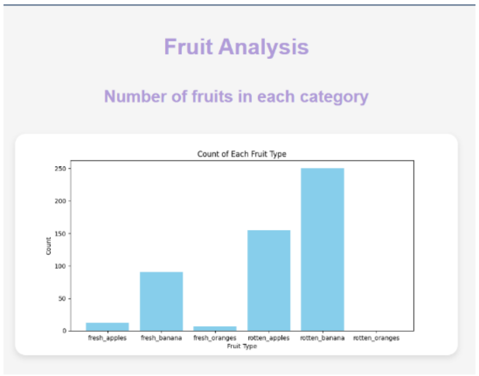

FIGURE 4. Analytics dashboard for fruit grading system

The application layer encompasses the core functionalities, including image processing, detection and grading, settings management, and historical data management modules. The image processing module (Figure 5) enhances captured images through a pipeline of noise reduction (Gaussian blur), normalization (pixel value scaling), and contrast adjustment to ensure high-quality inputs. The detection and grading module (Figure 6), powered by YOLOv5, analyzes preprocessed images using a 19x19 grid with anchor boxes to detect fruits and predict bounding boxes and class probabilities. As illustrated in Figure 7, sourced from Exxact Corporation (2021), YOLO's object detection process divides the input image into a grid, where each cell predicts multiple bounding boxes and class probabilities simultaneously, achieving a mean Average Precision (mAP) of 93.87%, surpassing SSD (78.50%) and Faster R-CNN (73.45%). The settings management module enables customization of grading parameters, while the historical data management module stores and retrieves past grading records for trend analysis.

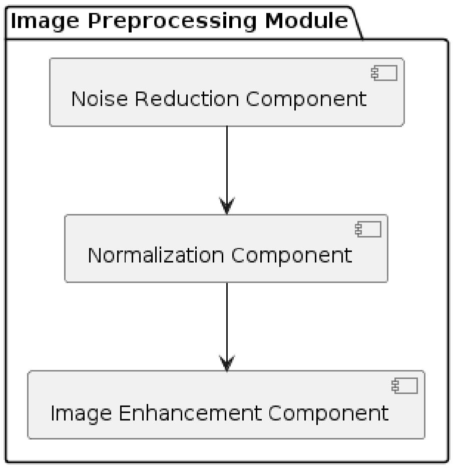

FIGURE 5. Module diagram for image processing module

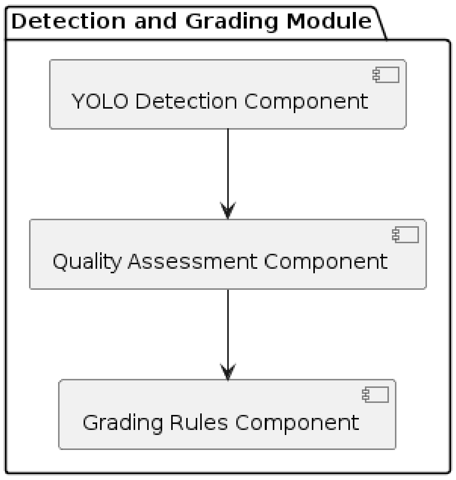

FIGURE 6. Module diagram for detection and grading module

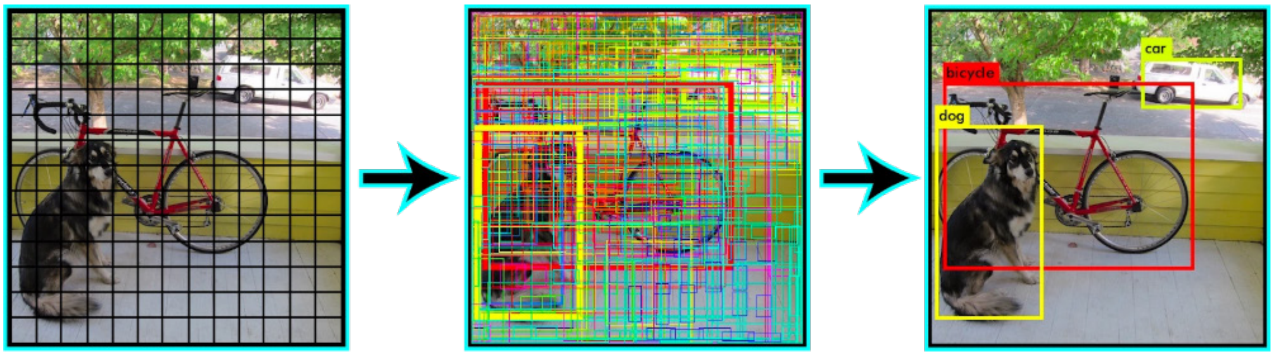

FIGURE 7. YOLO algorithm object detection

The infrastructure layer forms the system's foundation, comprising a high-resolution camera, a Raspberry Pi 4 (4GB RAM, quad-core Cortex-A72), an AI model repository, and a SQLite database. The camera captures raw images, interfacing with the image processing module. The Raspberry Pi 4 enables real-time detection and grading at 0.12 seconds per frame, compared to 0.15 seconds for SSD and 0.18 seconds for Faster R-CNN (Table 2). Figure 9, sourced from Zhang et al. (2020), illustrates the YOLO algorithm's architecture, featuring a convolutional neural network (CNN) backbone that processes the entire image in a single pass, generating predictions for bounding boxes, object classes, and confidence scores. The AI model repository stores YOLOv5 weights, updated periodically, while the database manages grading results, configurations, and historical data. The real-time processing module (Figure 8) ensures seamless operation by acquiring, preprocessing, and grading images on-the-fly, delivering immediate feedback to operators.

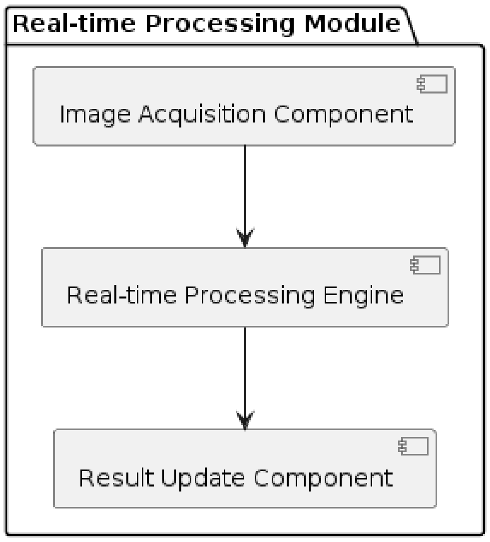

FIGURE 8. Module diagram for real-time processing module

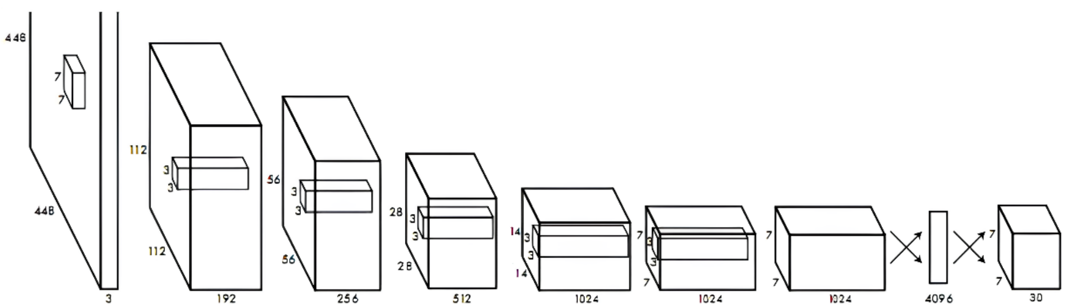

FIGURE 9. YOLO algorithm architecture

The YOLOv5 algorithm's workflow is pivotal to the system's real-time performance. As shown in Figure 10, sourced from Lu et al. (2022), the YOLO flowchart outlines its operation: the input image is divided into a 19x19 grid, with each cell predicting multiple bounding boxes using anchor boxes, alongside class probabilities and confidence scores. Low-confidence boxes are filtered, and non-maximum suppression eliminates redundant overlapping boxes, ensuring precise detection. This workflow, combined with the Raspberry Pi's portability, makes the system cost-effective. Previous studies, such as Rashid and Fadzil (2023), have validated YOLO's feasibility on Raspberry Pi for fruit detection, and this system enhances these implementations by incorporating internal defect detection (e.g., browning) and a modular design, achieving a robust solution for fruit quality control.

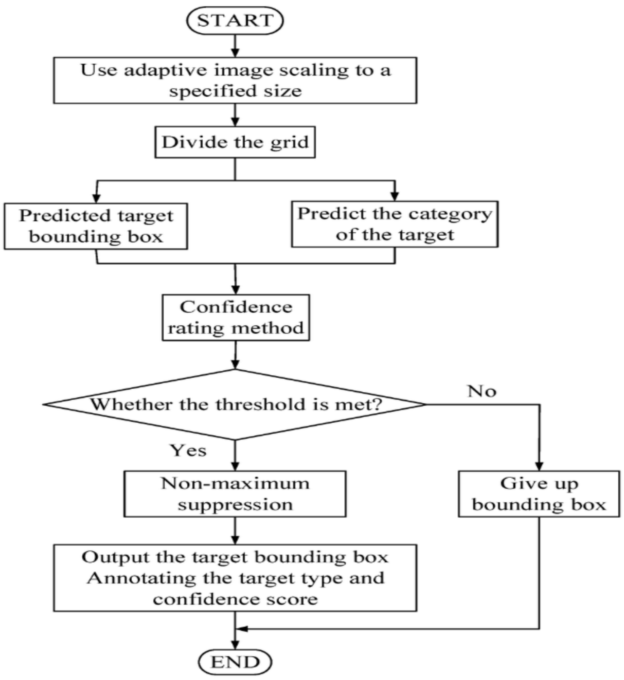

FIGURE 10. Flowchart of YOLO algorithm

The implementation strategy for the fruit grading system adopts a bottom-up approach, starting with the development of foundational components and progressively building toward higher-level modules. This method prioritizes the creation of low-level functionalities, gradually integrating these components to construct the complete system. Initially, the implementation focuses on developing essential low-level components, such as image preprocessing algorithms, fruit detection algorithms, and grading algorithms. These components form the backbone of the system, handling the processing of raw image data, identifying fruits within images, and evaluating their quality based on predefined criteria.

Once these foundational components are developed and individually tested to ensure accuracy and reliability, the integration phase begins. Modules are incrementally combined, starting with the integration of basic components to form higher-level modules. For example, the image preprocessing and fruit detection algorithms are combined to create the image processing module. This step-by-step integration enables the systematic construction of the system, with each module adding new layers of functionality.

Throughout the integration process, comprehensive testing is conducted at each stage to verify the proper functioning and compatibility of the integrated components. Unit tests, integration tests, and system tests are performed to validate the system's behavior and ensure it meets specified requirements. Any issues or discrepancies identified during testing are promptly addressed to maintain the system's integrity and stability.

As integration progresses, additional features and functionalities are incrementally incorporated. This iterative development approach supports continuous refinement and enhancement of the fruit grading system. New modules are added, existing components are optimized, and feedback from testing and user evaluations drives further improvements.

By employing a bottom-up implementation strategy, the fruit grading system is systematically built from its foundational components to higher-level modules, resulting in a robust and reliable solution that meets the desired specifications. This approach ensures flexibility, scalability, and maintainability, enabling the system to adapt to evolving requirements and achieve long-term success.

OVERVIEW OF DESIGN

The design of the fruit grading system leverages a bottom-up implementation strategy, prioritizing the development of foundational components before integrating them into higher-level modules. This approach ensures a systematic and methodical construction process, enabling the creation of a robust and reliable system from the ground up.

At the core of the design is a sophisticated image processing pipeline responsible for analyzing images captured by the system's camera. State-of-the-art image preprocessing algorithms are employed to enhance image quality, including noise reduction, normalization, and image enhancement techniques. These algorithms ensure that subsequent detection and grading processes operate on high-quality image data, improving the accuracy and reliability of the system's assessments.

For fruit detection, the YOLO algorithm was selected for its efficiency and effectiveness in object detection tasks. YOLO's real-time capabilities allow the system to rapidly identify fruits within images with minimal computational overhead, ensuring swift and accurate detection even in dynamic environments, such as busy food stalls.

To facilitate seamless user interaction, an intuitive user interface was designed, enabling stall operators to access and manage system functionalities effortlessly. The interface provides real-time feedback on grading results and historical data visualization, empowering users to make informed decisions and optimize operations effectively.

Overall, the design prioritizes efficiency, accuracy, and user-friendliness, incorporating advanced algorithms and technologies to deliver a comprehensive fruit grading system that meets the diverse needs of food stall operators. Through careful consideration of each design decision, the system addresses the challenges of fruit grading while setting new standards for efficiency and innovation in the industry.

RESULTS AND DISCUSSION

The automated fruit grading system, implemented using the YOLOv5 algorithm on a Raspberry Pi 4, was evaluated using a dataset of 5,000 fruit images, including apples, oranges, and bananas in both fresh and rotten states. The system achieved a mean Average Precision (mAP) of 93.87%, outperforming SSD (78.50%) and Faster R-CNN (73.45%) as shown in Table 2. Real-time processing tests demonstrated an average processing time of 0.12 seconds per frame, equivalent to 8 frames per second (FPS), compared to 0.15 seconds for SSD and 0.18 seconds for Faster R-CNN. This performance enables efficient grading in dynamic environments, such as food stalls, where rapid assessment is critical. Visual results, depicted in Figures 11–14, illustrate the system's capability to accurately distinguish fresh fruits (e.g., an 88% fresh banana with minor spots) from rotten ones, providing reliable quality assessments.

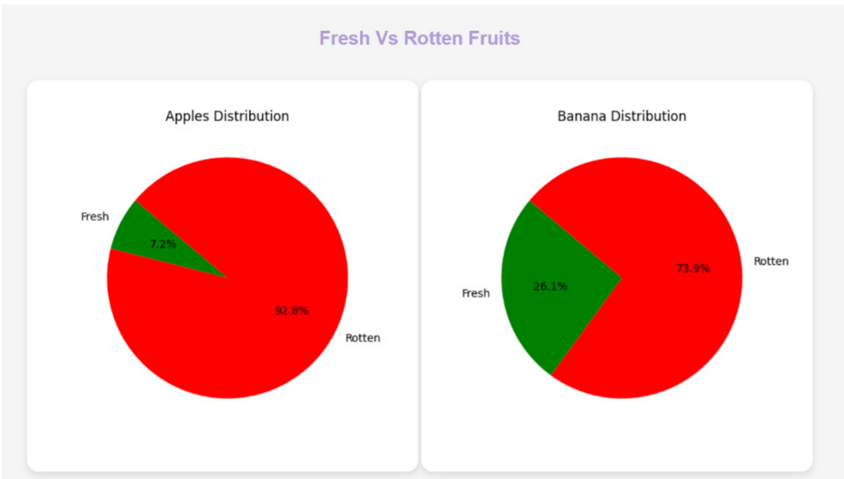

FIGURE 11. Fresh Vs Rotten Fruits

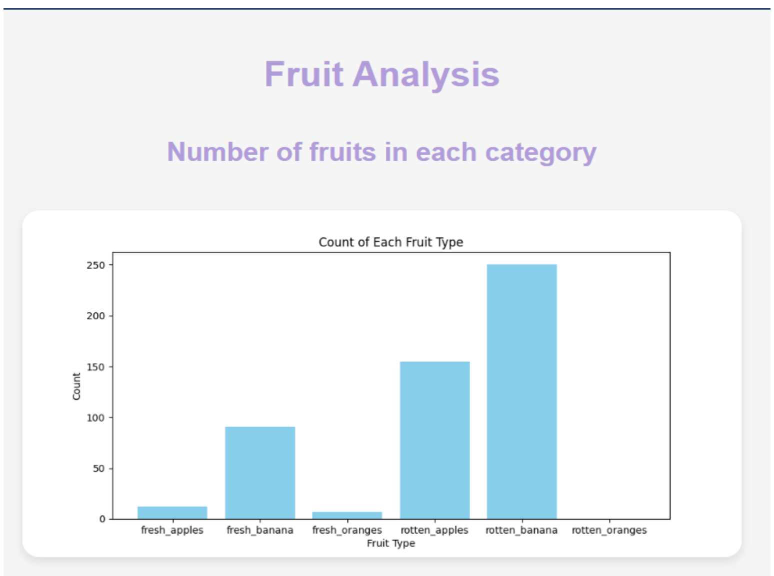

FIGURE 12. Number of fruits in each category

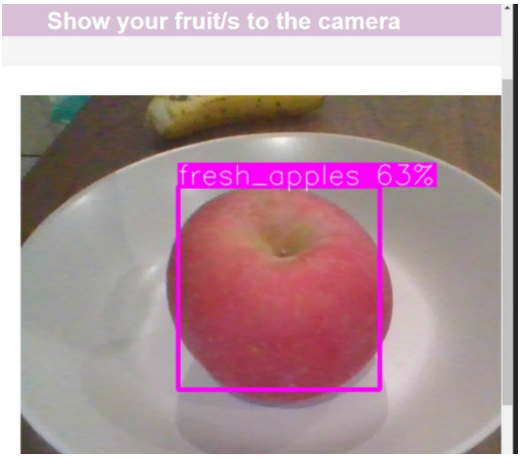

FIGURE 13. Camera detected fresh apple

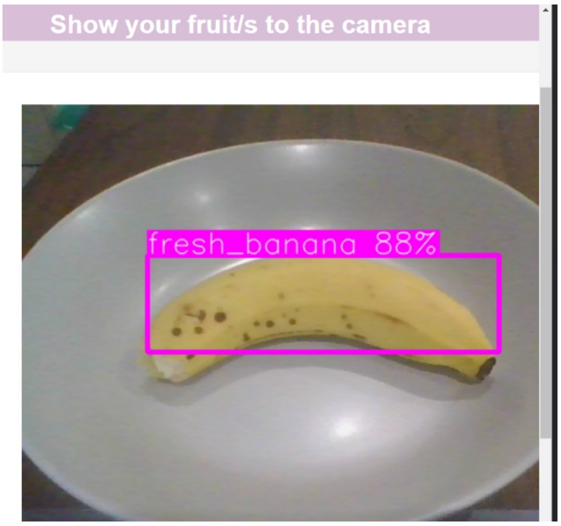

FIGURE 14. Camera detected fresh banana

The system offers significant advantages over traditional manual grading methods. It is up to 10 times faster, reducing the average grading time from 1.2 seconds per fruit manually to 0.12 seconds per fruit. The high mAP ensures consistent and accurate detection, minimizing human error. Deploying the system on a Raspberry Pi 4 enhances cost-effectiveness, making it accessible for small to medium-sized enterprises compared to expensive commercial systems like TOMRA's Nimbus BSI+. Additionally, the system's modular design and cloud integration facilitate scalability and remote management, allowing vendors to adapt to varying operational needs. These features align with prior YOLO implementations on Raspberry Pi, as noted by Rashid and Fadzil (2023), but the inclusion of internal defect detection adds unique value to this solution.

Despite its strengths, the system faces certain limitations. It is sensitive to environmental conditions, such as poor lighting or occlusions, which can reduce accuracy by up to 10%. The initial setup and calibration require technical expertise, potentially posing a barrier for users with limited technical knowledge. Compared to high-end commercial systems, the Raspberry Pi's lower processing power may constrain performance in high-volume scenarios. However, the system's affordability and real-time capabilities make it a compelling alternative for small-scale vendors. When compared to manual grading, it eliminates subjectivity and ensures standardized quality assessments. While commercial systems may offer superior hardware, their high costs limit accessibility, whereas this solution balances performance and affordability effectively.

In summary, the fruit grading system demonstrates robust performance, achieving high accuracy and real-time processing suitable for practical deployment. Its cost-effectiveness and scalability make it an attractive option for small to medium-sized food stalls. By addressing environmental sensitivity through future dataset expansion and hardware optimization, the system can further enhance its applicability. This project validates the feasibility of AI-driven fruit grading, building on existing YOLO implementations while introducing advancements in accessibility and functionality for agricultural applications.

CONCLUSION

This project successfully developed an automated fruit grading system integrating YOLOv5, image processing, and Raspberry Pi 4, achieving a mean Average Precision of 93.87% and a processing speed of 8 FPS. Deployed at Universiti Sains Malaysia's food stalls, it enhances grading efficiency, reduces errors, and ensures consistent quality, benefiting small-scale vendors. The system's affordability and portability make it a practical solution for agricultural applications.

Future work includes expanding the training dataset to include diverse fruit types and environmental conditions, improving robustness. Optimizing YOLOv5 for lower latency and integrating advanced sensors (e.g., NIR) could further enhance performance. These advancements will promote broader adoption of AI-driven solutions in agriculture, supporting sustainable practices.

ACKNOWLEDGEMENT

We express gratitude to Universiti Sains Malaysia and the School of Computer Science for their invaluable support and resources, which were instrumental in completing this project.

DECLARATION OF COMPETING INTEREST

None.

REFERENCES

Aalsalem, M., Khan, W., and Dhabbah, K. 2015. An automated vehicle parking monitoring and management system using ANPR cameras. *IEEE International Conference on Advanced Communication Technology (ICACT)*.

Al-Turjman, F., and Malekloo, A. 2019. Smart parking in IoT-enabled cities: A survey. *Sustainable Cities and Society* 49: 101608.

Alam, M., Saha, S., Bostami, M., Islam, M., Aadeeb, M., and Islam, M. 2023. A survey on IoT driven smart parking management system: Approaches, limitations and future research agenda. *IEEE Access* 11: 119523-119543.

Arakeri, M., and Lakshmana, P. 2016. Computer vision based fruit grading system for quality evaluation of tomato in agriculture industry. *Proceedings of the International Conference on Communication, Computing and Virtualization* 79: 426-433.

Bhargava, A., and Bansal, A. 2018. Fruits and vegetables quality evaluation using computer vision: A review. *Journal of King Saud University - Computer and Information Sciences*, 1-16.

Bresilla, K., Perulli, G., Boini, A., Morandi, B., Corelli Grappadelli, L., and Manfrini, L. 2019. Single-shot convolution neural networks for real-time fruit detection within the tree. *Frontiers in Plant Science* 10: 611.

Du, J. 2018. Understanding of object detection based on CNN family and YOLO. *Journal of Physics: Conference Series* 1004: 012029.

Exxact Corporation. 2021. YOLOv5 PyTorch Tutorial. https://www.exxactcorp.com/blog/Deep-Learning/YOLOv5-PyTorch-Tutorial

Hargava, A. B. 2018. Fruits and vegetables quality evaluation using computer vision: A review. *Journal of King Saud University - Computer and Information Sciences*, 1-67.

Haris, M., and Głowacz, A. 2021. Road object detection: A comparative study of deep learning-based algorithms. *Electronics* 10(16): 1932.

Jawale, D., and Deshmukh, M. 2017. Real-time automatic bruise detection in (Apple) fruits using thermal camera. *Proceedings of the International Conference on Communication and Signal Processing (ICCSP)*, Melmaruvathur, India, 1080-1085.

Kirchhof, N., Walsh, K., Koirala, A., McCool, C., and Thakur, T. 2021. In-field detection of fruit on trees. Hort Innovation, 10-20.

Li, Y., Feng, X., Liu, Y., and Han, X. 2021. Apple quality identification and classification by computer vision based on deep learning. *Scientific Reports* 11: 16618.

Liu, T., Zhu, Z., Bai, Y., and Liao, G. 2022. YOLO-based efficient vehicle object detection. 33(4): 069-079.

Lu, S., Lu, Z., Aok, S., and Graham, L. 2018. Fruit classification based on six-layer convolutional network. *International Conference on Digital Signal Processing*, Shanghai.

Lu, Y., Qiu, Z., Liao, C., Zhou, Z., Li, T., and Wu, Z. 2022. A GIS partial discharge defect identification method based on YOLOv5. *Applied Sciences* 12(16): 8360. doi:10.3390/app12168360.

Mui, D., and Ching, S. 2020. An assessment of the viability of the smart parking system: The case of a smart city initiative in Malaysia. *Global Business and Organizational Excellence* 39(5): 26-34.

Mureșan, H., and Oltean, M. 2018. Fruit recognition from images using deep learning. *Acta Universitatis Sapientiae, Informatica* 1: 26-42. doi:10.2478/ausi-2018-0002.

Qi, L., Tong, D., and Phang, S. 2023. Integrated smart public parking system for Malaysia. *Journal of Physics* 2523(1): 012040.

Raj, R., et al. 2021. Fruit classification comparison based on CNN and YOLO. *IOP Conference Series: Materials Science and Engineering* 1187: 012031.

Rashid, T., and Fadzil, L. 2023. Comparative review of object detection algorithms in small single-board computers. *International Journal on Recent and Innovation Trends in Computing and Communication* 11(7): 244-252.

Saetchnikov, I., Tcherniavskaia, E., and Skakun, V. 2021. Object detection for unmanned aerial vehicle camera via convolutional neural networks. *IEEE Journal on Miniaturization for Air and Space Systems* 2(2): 98-103.

Wang, Z., Jin, L., Wang, S., and Xu, H. 2022. Apple stem/calyx real-time recognition using YOLO-v5 algorithm for fruit automatic loading system. *Postharvest Biology and Technology* 185: 111808.

Zawbaa, H. M., Abbass, M., Hazman, M., and Hassenian, A. E. 2014. Automatic fruit image recognition system based on shape and colour features. *International Conference on Advanced Machine Learning Technologies and Applications*, 278-290.

Zeng, G. 2017. Fruit and vegetables classification system using image saliency and convolutional neural network. *Proceedings of the Technology and Mechatronics Engineering Conference*, 324-331. doi:10.1109/ITOEC.2017.8122370

Zhang, X., et al. 2020. A survey on deep learning for real-time object detection. Proceedings of the 2020 4th International Conference on Electronic Information Technology and Computer Engineering (EITCE). doi:10.1145/3441250.3441278