Automatic Grading and Sorting Machine for Eggplant Using Machine Learning

Erika R. Manalo, Monica M. Magnaye, Angela V. Labuguen & Merwin A. Carandang

One of the leading crops in the Philippines is the eggplant, locally known as "talong." On average, eggplant is cultivated on 21,225 hectares annually, producing approximately 9.95 tons per hectare, which is about half of the Asian average (ISAAA, 2023). Eggplants typically have a long, slender, cylindrical shape that can be straight or slightly curved. Botanically classified under the Solanaceae family, which also includes peppers, tomatoes, and potatoes, eggplants vary in color from white to dark purple (Taher et al., 2017). Although commonly regarded as a vegetable, eggplants are technically a fruit because they develop from flowering plants and contain seeds.

Fronda (2022) studied vegetable production in Nueva Ecija, Philippines, and found that cultural practices and input requirements significantly affect vegetable output. Farmers indicated that higher production volumes result in increased profits, making grading and sorting a critical step in maximizing returns. Similarly, In Janagdong II, Sariaya, Quezon, Philippines, eggplants and other vegetables are primarily graded and sorted manually, which introduces challenges such as subjective assessment, fatigue, inconsistency, and decreased productivity. Because eggplants are relatively easy to cultivate, many farmers prioritize planting them, making an effective grading and sorting system particularly valuable.

To address these challenges, this study developed an automatic grading and sorting machine for eggplant using machine learning. The system focuses exclusively on eggplants and utilizes image processing to classify vegetables into three categories: good, semi-good, and defective, based on size, shape, and appearance. Sensors are employed to guide each eggplant to its appropriate bin, ensuring accurate segregation. The system is designed for convenience and efficiency: eggplants are placed on a conveyor belt, pass through the camera's field of view, and undergo image processing to extract data for quality assessment. The ultimate goal is to sort eggplants continuously and reliably into their respective grade-level containers.

The benefits of automated sorting are significant when compared to manual methods. Automated systems can handle large volumes of vegetables in shorter periods, increasing processing speed and reducing delays caused by fatigue and human error. They also provide consistent and objective grading based on predetermined criteria, whereas manual sorting may vary due to individual subjectivity. Moreover, automated systems can incorporate multiple sorting criteria simultaneously, whereas manual sorting is limited to basic assessments. From a labor and cost perspective, automation reduces the need for a large workforce, resulting in operational savings and improved reliability. Additionally, automated sorting systems can collect and analyze data to monitor performance and continuously improve efficiency, an advantage not readily available with manual methods.

This study focuses on the design and development of an eggplant automatic grading and sorting machine using machine learning. By implementing this system, labor costs can be reduced, productivity and accuracy can be increased, and the overall efficiency of vegetable grading and sorting can be enhanced, particularly in agricultural operations.

Theoretical Framework

Automation and Artificial Intelligence in Vegetable Sorting

Mishra et al. (2024) highlighted that traditional manual sorting of vegetables is a labor-intensive, time-consuming, and subjective process. Inconsistencies arise because different workers apply varied criteria, which affects the quality grading and market value of the produce. Automation, supported by artificial intelligence (AI), addresses these challenges by standardizing grading procedures. By integrating Raspberry Pi and Arduino microcontrollers with sensors and computer vision, the system can detect, classify, and segregate eggplants efficiently. This reduces human fatigue and improves productivity, enabling continuous operation without breaks or decline in performance (Valtonen et al., 2025).

AI-assisted automation provides significant benefits for small and medium-scale farmers (Kumari et al., 2025; Hoque & Padhiary, 2024; Gupta & Kumar Pal, 2025; Bazargani & Deemyad, 2024; Assimakopoulos et al., 2024; Song et al., 2025; Rathor et al., 2024). It ensures that grading is uniform and objective, which can increase customer satisfaction and potentially improve profits. By using cost-effective components like Raspberry Pi and Arduino, the system becomes accessible to farmers with limited resources. The successful integration of AI and automation demonstrates that modern technology can significantly enhance traditional agricultural practices, reducing labor costs while maintaining high standards of accuracy and reliability (Mishra et al., 2024).

Image Processing and Object Recognition Techniques

Loresco and Dadios (2018) focused on improving the accuracy of vegetable detection using image processing and object recognition. Convolutional Neural Networks (CNNs) were applied to extract essential features from eggplants, while fuzzy logic handled uncertainties such as variations in shape, orientation, and lighting. This combination allowed the system to classify vegetables accurately into grades such as Good, Semi-Good, or Defective. Keras and MATLAB provided frameworks for model training and optimization, ensuring the detection algorithms were reliable and precise. The study demonstrated that the integration of AI and image processing could overcome the limitations of manual inspection by providing objective, repeatable results.

Object recognition models enable the system to handle real-world complexities, such as overlapping vegetables or inconsistent appearances (Khan et al., 2025; Xiao et al., 2023; Jamali et al., 2025; Edozie et al., 2025). By incorporating these advanced algorithms, the sorting machine can assess multiple quality parameters simultaneously, ensuring accurate grading under diverse conditions. This capability is crucial for commercial applications where consistency and reliability are necessary for maintaining product quality and market competitiveness. The use of image processing and object recognition in agricultural automation illustrates the potential of AI to enhance operational efficiency and reduce human errors (Loresco & Dadios, 2018; Jiang et al., 2025; Song et al., 2025; Sow et al., 2024; Bhat et al., 2025).

Deep Learning and Crop Detection

Zheng et al. (2018) emphasized the role of deep learning in improving crop detection and quality assessment. Using a large dataset of 1,600 images, the system could identify multiple vegetables, including eggplants, cucumbers, tomatoes, and peppers, based on size, color, and surface defects. Deep learning techniques such as image segmentation and keypoint detection allowed precise localization and analysis of each vegetable, enabling accurate classification into quality grades. This approach ensures that even subtle differences in appearance are captured, which manual inspection might overlook.

Furthermore, the deep learning framework enables continuous improvement through model training with new data (Prapas et al., 2021). As the system processes more images, it refines its detection capabilities, increasing accuracy and reducing misclassification. This scalability makes the system suitable for commercial agricultural operations with large volumes of produce (Totin et al., 2020). By leveraging deep learning, the system ensures high standards of quality assessment, improves sorting efficiency, and contributes to more consistent product grading.

Integration of Software and Hardware Components

The successful operation of the automated eggplant grading system relies on the integration of software and hardware components. Python and TensorFlow handle image processing and deep learning tasks, while OpenCV manages real-time camera feeds. Thonny IDE provides a platform for coding and debugging on the Raspberry Pi, ensuring smooth execution of algorithms. Arduino, programmed using C++ through Arduino IDE, manages physical tasks such as motor rotation, sensor detection, and bin sorting. Together, these components enable the system to detect, classify, and physically sort vegetables accurately and efficiently.

This integration ensures that the hardware and software communicate seamlessly, allowing the system to respond promptly to detected inputs. The Arduino executes commands for actuators, such as servo motors, based on image recognition results from the Raspberry Pi, enabling automated sorting. The combination of reliable software algorithms with responsive hardware ensures consistent operation, minimizes downtime, and enhances overall system reliability. This comprehensive integration highlights the importance of synchronizing computational and mechanical components in agricultural automation for optimal performance (Mishra et al., 2024; Loresco & Dadios, 2018).

Implications for Agricultural Efficiency and Productivity

Automated eggplant sorting offers significant improvements in agricultural productivity compared to manual methods. Manual sorting is slow, inconsistent, and requires a substantial workforce, which increases labor costs. Automated systems can process larger volumes of produce in a shorter time, maintaining consistent quality and minimizing human error. By providing objective and standardized grading, automated systems can improve market competitiveness and profitability for farmers.

Additionally, automated sorting aligns with the principles of precision agriculture, which seeks to optimize efficiency and resource utilization. Data collected during sorting, such as size, color, and defect rates, can be analyzed to inform future cultivation practices, storage decisions, and supply chain management. This enables smarter, data-driven decision-making, enhancing operational efficiency and reducing waste. Ultimately, AI-based vegetable sorting systems demonstrate the potential to transform agricultural practices, providing reliable, efficient, and cost-effective solutions for modern farming (Mishra et al., 2024; Loresco et al., 2019).

Research Framework

The data collected in this study served as the foundation for developing and evaluating the automatic eggplant grading and sorting system.

Data

Vegetable information was gathered, which includes the names and types of vegetables to be sorted. This ensures that the system can identify and classify each vegetable accurately. The categorized characteristics of the vegetables, size, color, and appearance, were documented. These characteristics are essential for the CNN algorithm, enabling the machine to accurately detect and classify vegetables based on predefined quality parameters. Finally, grading information was recorded, representing the classification outcomes after the sorting process: good, semi-good, and reject. This information provides feedback on the system's effectiveness in maintaining quality standards during sorting.

Design

The study employed a developmental research design to explore and develop innovative features for the automated sorting system. The primary goal of the study was to enhance the accuracy and efficiency of vegetable grading while ensuring that quality standards are maintained. This approach allowed the developers to observe, analyze, and refine the system iteratively, providing practical solutions tailored to the needs of local farmers.

Respondents

The respondents consisted of farmers from Brgy. Janagdong 2 in Sariaya, Quezon, who are actively engaged in vegetable cultivation, particularly eggplants. To ensure a representative evaluation, the study used stratified sampling, categorizing respondents based on relevant characteristics to maintain diversity in feedback. Since the barangay has a total of 35 farmers, all of them were included in the study to provide comprehensive insights into the practical performance and usability of the proposed system.

Instrument

Data collection involved observations and interviews with randomly selected farmers to gather qualitative insights on current manual sorting practices, challenges, and expectations. Additionally, the system evaluation was conducted using survey questionnaires, designed based on the ISO 25010 software quality standard, with five questions per criterion. The questionnaire addressed key aspects of the system, including functionality, efficiency, reliability, and usability, ensuring that the evaluation captured both technical performance and user satisfaction.

Statistical Analysis

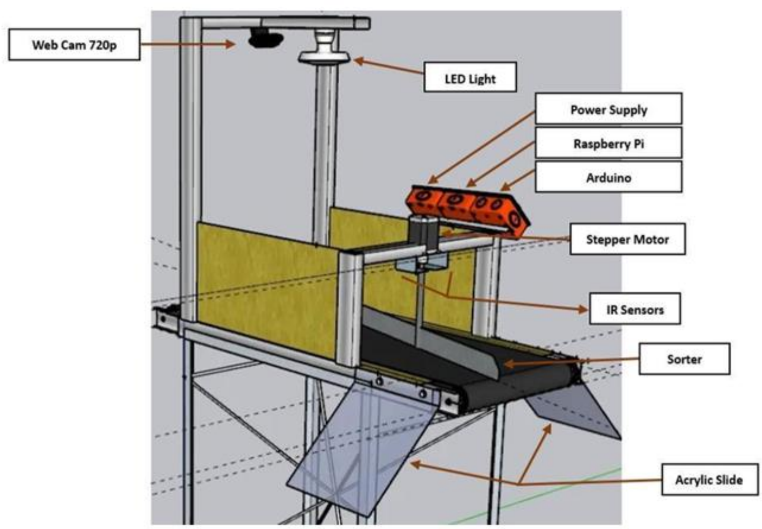
The researchers analyzed the survey responses using the 5-point Likert scale and average weighted mean (AWM). The ISO 25010 criteria guided the analysis, covering functional suitability, performance efficiency, compatibility, usability, reliability, security, maintainability, and portability. The Likert scale allowed respondents to indicate their level of agreement with each statement regarding the system's performance, from "strongly disagree" to "strongly agree." This method provided quantitative insights into user satisfaction and highlighted areas for improvement in system design and operation. By applying AWM, the study was able to determine the overall effectiveness and acceptance of the proposed automatic sorting machine among the farmers.

Experimental Design

Figure 1 illustrates the initial design of the sorting machine as visualized by the proponents.

Figure 1

Sorting machine sample design



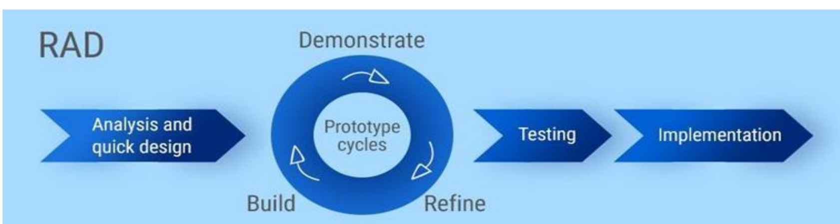
The design concept was developed after reviewing various sources, which served as the foundation for the developers. However, the concept remains flexible and open to modifications based on feedback from farmers or recommendations from industry professionals. The prototype is designed at waist height to ensure that farmers can easily access and operate the machine without bending, thereby reducing the risk of back strain. Additionally, it incorporates a sideways slide mechanism to control the momentum of falling vegetables, preventing damage during the sorting process and ensuring that the produce maintains its quality.

Procedures of the Different Phases

According to Puzhevich (2020), the Rapid Application Development (RAD) model is a variant of Agile methodology that emphasizes rapid prototype development and minimizes rigid planning. This approach allows developers to implement updates based on user feedback rather than adhering to a strict schedule. The RAD model focuses on active development activities with the goal of quickly creating a functional system.

Figure 2

Rapid Application Development



Analysis and quick design. In the initial phase, research and preliminary drafts for the system design were conducted. The developers carefully assessed how to apply the CNN algorithm to recognize vegetables' color, size, and quality. The entire design and process of the sorting machine were systematically outlined, ensuring that each step aligned with the desired functionality before finalizing the design.

Build, Demonstrate & Refine. During this phase, the prototype was developed and constructed according to the finalized design. Materials were secured, and the CNN algorithm was implemented to enable the machine to accurately identify the color, size, and quality of vegetables. Each module of the system was tested individually to verify proper functioning, and necessary modifications were made to address any deficiencies.

Testing. The testing phase involved rigorous debugging and evaluation to confirm that the machine met its intended purpose. The developers examined all system components and processes to ensure accuracy and reliability, preparing the prototype for deployment with minimal errors.

Implementation. In the final phase, the proposed system became fully operational and ready for routine use. A maintenance plan was established to guide users in resolving potential issues over time, ensuring sustained functionality and efficiency of the sorting machine.

Technical Framework

This section presents the overall design of the project and evaluates its effectiveness.

Materials

The following materials were utilized in the development of the prototype.

Software

Table 1 presents the software programs utilized in developing the machine. These programs are essential for the successful functioning of the system, particularly for image processing, model training, and hardware communication.

Table 1

Software specifications

Software	Description
Python	Python is a popular, high-level and a general-purpose programming language. It is the language used to program the Image Processing feature.
C++	An object-oriented programming language that provides clear structure to programs and allows code to be reused. This language is used to program the Arduino in order to interact with the machine components.
Arduino IDE	Arduino IDE contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them.
Thonny	Thonny is an integrated development environment for Python that is designed for beginners. It is where the Image Processing is coded and debugged.
OpenCV	OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. It is used for camera logic.
Tensorflow	TensorFlow is a rich system for managing all aspects of a machine learning system. It is used for deep learning of image processing.

Python was primarily used for programming the image processing features via the Thonny IDE. OpenCV handled the camera feed and processed images of the vegetables. TensorFlow was employed to train the system's CNN model for accurate vegetable recognition. C++ was used to program the Arduino microcontrollers, and Arduino IDE facilitated the uploading of programs to the Arduino hardware, enabling communication with the prototype's components.

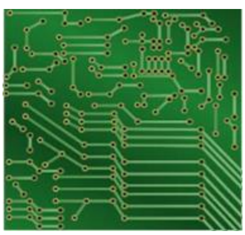
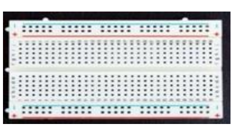
Hardware

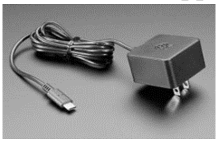
Table 2 presents the hardware components required for the development of the machine. Each component plays a specific role in ensuring the proper functioning of the system and facilitating the grading and sorting process.

Table 2

Hardware specifications

Hardware	Description	Specification
Raspberry Pi 4 	Raspberry Pi is a credit-card-sized minicomputer that turns a monitor, TV, mouse, or keyboard into a full-fledged PC	SDRAM: 1GB, 2GB, 4GB, or 8GB LPDDR4-3200 (depending on model) IEEE 802.11ac wireless at 2.4 and GHz, Bluetooth 5.0, and BLE Ethernet Gigabit Two USB 3.0 ports and two USB 2.0 ports. 40-pin GPIO header for Raspberry Pi (fully backwards compatible with previous boards) There are two micro-HDMI ports
Arduino Mega 2560 	This software is used for writing, compiling, and uploading the code into the Arduino board. This unit comes with a USB interface so a USB cable can be used to connect the device with the computer through which you can transfer sketch (Arduino program is called a sketch) to the board.	The ATmega2560 is a Microcontroller The operating voltage of this microcontroller is 5volts The recommended Input Voltage will range from 7volts to 12volts The input voltage will range from 6volts to 20volts The digital input/output pins are 54 where 15 of these pins will supply PWM o/p. Analog Input Pins are 16 DC Current for each input/output pin is 40 mA DC Current used for 3.3V Pin is 50 mA

Hardware	Description	Specification
		Flash Memory like 256 KB where 8 KB of flash memory is (SRAM) is 8 KB (EEPROM) is 4 KB
LED Lights 	A light-emitting diode is a semiconductor device that emits light when current flows through it. Electrons in the semiconductor recombine with electron holes, releasing energy in the form of photons.	Long Life: LEDs can last over 100,000 hours (10+ years) if used at rated specifications No annoying flicker like from fluorescent lamps LEDs are impervious to heat, cold, shock and vibration LEDs do not contain breakable glass Solid-State, shock and vibration resistant Extremely fast turn On/Off times Low power consumption puts less load on the electrical systems increasing battery life
Circuit Board 	A printed circuit board (PCB; also printed wiring board or PWB) is a medium used to connect electronic components to one another in a controlled manner.	8 x 12cm (3.2 x 4.7″) outside dimensions 1.6mm (0.64″) thickness FR-4 double-sided construction with green solder mask Holes on 2.54mm (0.1″) centers in grid pattern with grid numbered/lettered in copper on two edges of the board. Holes are plated through with same pad pattern on both sides of the board. Edge solder pads are on 2.54mm (0.1″) centers HASL (SnPb) plating over copper for best solderability
Bread Board 	A breadboard, solderless breadboard, or protoboard is a construction base used to build semipermanent prototypes of electronic circuits. Unlike a perfboard or stripboard, breadboards do not require soldering or destruction of tracks and are hence reusable.	Distribution Strips are two Wire Size is 21 to 26 AWG wire Tie Points are two hundred Withstanding Voltage is 1,000V AC Tie points within IC are 630 Insulation Resistance is DC500V or 500MΩ Dimension is 6.5*4.4*0.3 inch Rating is 5Amps ABS plastic through color legend ABS heat Distortion Temperature

Hardware	Description	Specification
		is 183° F (84° C) Hole or Pitch Style is 2.54mm
RPi 4 Power Supply 	The Raspberry Pi 15W USB-C Power Supply is designed to power Raspberry Pi 4 and Raspberry Pi 400 computers	USB-C power supply specially designed and tested for the Raspberry Pi 4 Includes noise filter for added stability 18 AWG Cable UL Listed 5ft Cable Length
Servo Motor 	A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity, and acceleration. It consists of a suitable motor coupled to a sensor for position feedback.	Weight: 55g Dimension: 39.5mm x 20.5mm x 40.7mm Stall Torque: 9.4kg/cm (4.8v); 11kg/cm (6v) Op. speed: 0.20sec/60degree (4.8v); 0.16sec/60degree (6.0v) Operating voltage: 4.8~ 6.6v Gear Type: Metal gear Temperature range: 0- 55deg Servo wire length: 30cm Rotation angle: 180 degree
Stepper Motor 	A stepper motor, also known as step motor or stepping motor, is a brushless DC electric motor that divides a full rotation into a number of equal steps.	Size: 42.3 mm square × 48 mm, not including the shaft (NEMA 17) Weight: 350g Shaft diameter: 5 mm “D” Steps per revolution: 200 Current rating: 1.2 A per coil Voltage rating: 4 V Resistance: 3.3 Ω per coil Holding torque: 3.2 kg-cm (44 ozin) Inductance: 2.8 mH per coil Lead length: 30 cm (12″) Output shaft supported by two ball bearings
IR Sensor 	An infrared detector is a detector that reacts to infrared radiation.	The operating voltage is 5VDC I/O pins – 3.3V & 5V Mounting hole The range is up to 20 centimeters The supply current is 20mA The range of sensing is adjustable Fixed ambient light sensor

Hardware	Description	Specification
Ultrasonic Sensor 	Ultrasonic Sensors measure the distance to the target by measuring the time between the emission and reception.	Power Supply: 3.3V – 5V. Operating Current: 8mA. Working Frequency: 40Hz. Ranging Distance: 3cm – 350cm/3.5m. Resolution: 1 cm. Measuring Angle: 15 degree. Trigger Input Pulse width: 10uS TTL. Dimension: 50mm x 25mm x 16mm
Buzzer 	A buzzer or beeper is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric.	12V Active Buzzer Alarm Magnetic Long Continuous Beep Tone
LCD 	LCD (Liquid Crystal Display) is a type of flat panel display which uses liquid crystals in its primary form of operation.	20x4 2004 LCD Display with I2C Adapter Module for Arduino Text Color: White Back light: Blue LED Backlight Dimensions: 98mm x 60mm x 20mm
Jumper Wires 	A jump wire is an electrical wire, or group of them in a cable, with a connector or pin at each end, which is normally used to interconnect the components of a breadboard or other prototype without soldering.	Length: 20cm. Pitch: 2.54mm. Female to female. 3pin (2.54mm) to 3pin (2.54mm) header. Withstand voltage: 300V DC Min. Insulation resistance: ≥ 5MΩ Min. Contact resistance: ≤ 5Ω Max.s
Container 	Any receptacle or enclosure for holding a product	Material: Plastic Size: 32cm x 25cm

The Raspberry Pi serves as the central processing unit of the system, where the image processing module is programmed and executed. The Arduino microcontroller manages the interaction and coordination between other hardware components. LED lights act as indicators to display the machine's operational status. The prototype frame and breadboard are used for connecting and testing the electrical components of the machine.

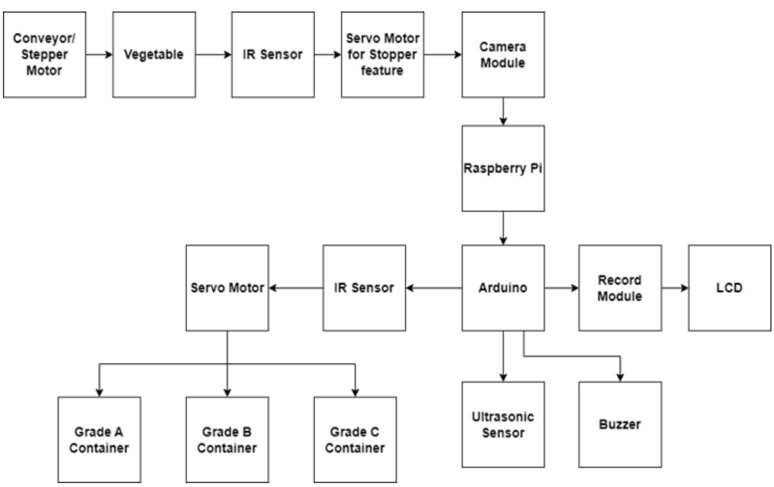
The servo motor controls the stopper mechanism of the prototype, while the stepper motor drives the conveyor system and handles the actual sorting of vegetables. The IR sensor detects the presence of vegetables on the conveyor, and the ultrasonic sensor, along with the buzzer, signals when the container is full. The LCD display provides notifications and records of the sorted vegetables, allowing the user to monitor system performance. Jumper wires connect all electrical components, ensuring proper transmission of signals and power. Finally, the container serves as the receptacle for the sorted vegetables, completing the sorting process.

Modeling

Figure 3 illustrates the relationships between the devices and materials used in the proposed system in a simplified manner.

Figure 3

Block diagram



When the machine is powered on, the conveyor belt, driven by the stepper motor, begins operating. As a vegetable is inserted, it reaches the first IR sensor, which detects the vegetable and sends a signal to the first servo motor to activate the stopper feature. The camera module then scans the vegetable, and the CNN algorithm, programmed on the Raspberry Pi, processes the image to determine the vegetable’s category. The Raspberry Pi sends this information to the Arduino, which controls the sorting mechanism. The vegetable then moves to the second IR sensor, which signals the stepper motor to guide it into the appropriate container. Simultaneously, the Arduino updates the count of sorted vegetables, stores the information in the records module, and displays the output on the LCD. This process is repeated for each subsequent batch of vegetables inserted into the machine.

Figure 4

Circuit diagram of the prototype

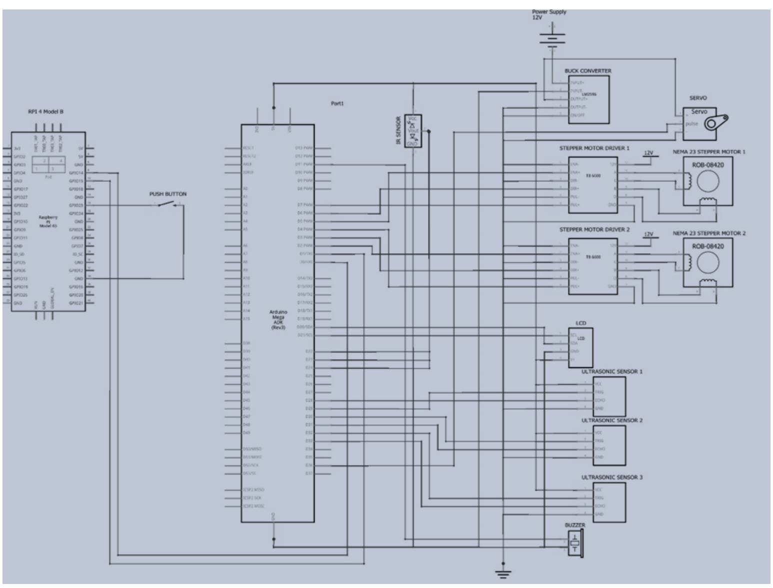


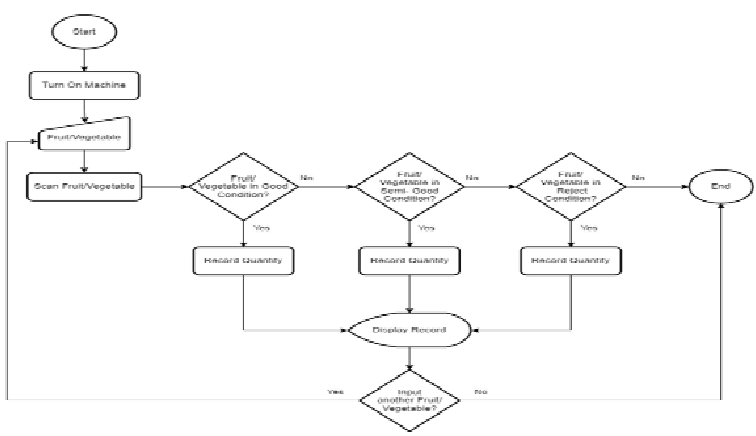
Figure 4 illustrates the overall architecture and circuit layout of the system, designed for easy installation and use. The diagram displays three infrared (IR) sensors, one servo motor, three ultrasonic sensors, two stepper motors, and an LCD connected to the board. The Arduino functions as the processor of the prototype, while the Raspberry Pi, which connects to the camera module, handles image processing. Serving as the central unit, the Raspberry Pi sends commands to the Arduino, which executes the sorting operations. The IR sensors detect vegetables and control the stopper mechanism, coordinating the sorter's movement, while the servo motor lifts and lowers the stopper. Obstacle detection is achieved through the ultrasonic sensors, which identify the vegetables and indicate when the container is full. The stepper motors control the conveyor belt and the movement of the sorter, ensuring precise handling of the vegetables. Meanwhile, the LCD module provides notifications, displaying error messages and the total count of sorted vegetables over a given period. By integrating these components effectively, the grading and sorting machine fulfills all functional requirements.

System Design

Figure 5 illustrates the system flow of the proposed sorting machine.

Figure 5

Flow chart



The process begins when the user turns on the machine and manually inputs a vegetable. The machine then scans the vegetable to determine its category. If the vegetable does not match any predefined categories, it will bypass the sorting process and continue to the next item. However, if the vegetable belongs to a recognized category, the system records the quantity of each sorted vegetable. After recording, the machine displays the updated record on the interface. The sorting process continues in a loop as long as the user continues to input vegetables; otherwise, the process concludes.

Figure 6

System architecture of the prototype

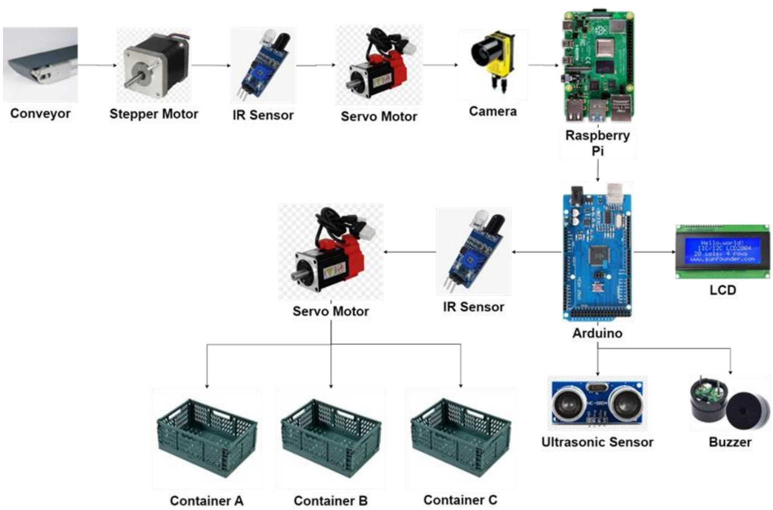


Figure 6 depicts the system architecture, illustrating the connections and interactions of the key components in the sorting machine. The process begins with the conveyor, powered by the stepper motor, which rotates to move the vegetable along the sorting path. When a vegetable is inserted, it passes the first IR sensor, which detects the item and sends a signal to the first servo motor to activate the stopper feature, ensuring the vegetable is properly positioned for scanning. The camera then captures an image of the vegetable, and the Raspberry Pi, programmed with the CNN algorithm, determines the vegetable's category and accuracy based on its characteristics.

Meanwhile, the Arduino manages the remaining hardware components, ensuring proper coordination between motors, sensors, and indicators. The ultrasonic sensor measures the distance between the vegetable and the container, triggering the buzzer if the container is near capacity to alert the user. Finally, the second IR sensor confirms the vegetable is in the sorting area, signaling the stepper motor to move the vegetable into its designated category. This architecture ensures precise, automated sorting while maintaining real-time monitoring and user notifications.

Figure 7

System structure of the prototype

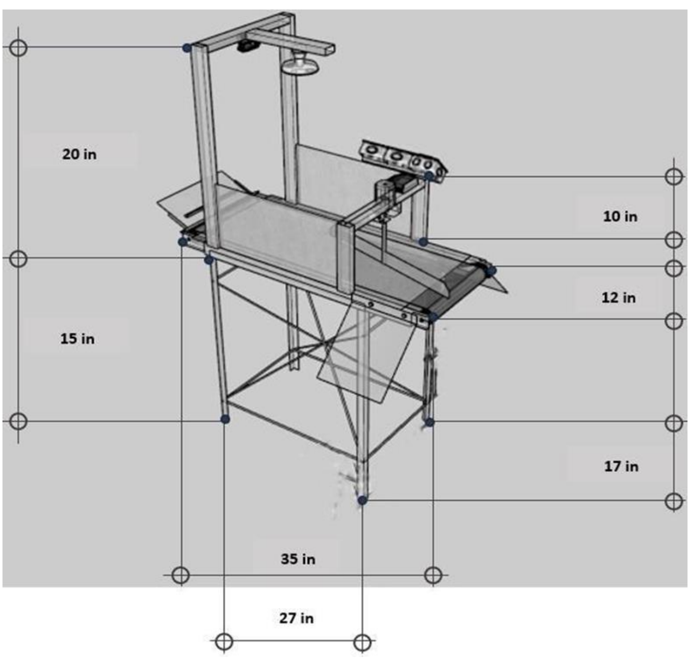


Figure 7 illustrates the system structure of the machine. Each component of the machine is clearly labeled to provide a better understanding of its functions and interactions. The prototype is designed at waist height to allow users to access the machine comfortably, eliminating the need to bend over while sorting and thereby reducing the risk of back strain. Additionally, the width of the sorting machine is adjustable based on the size of the vegetables being processed, ensuring efficient handling and smooth operation during the grading and sorting process.

Figure 8

Prototype's storyboard

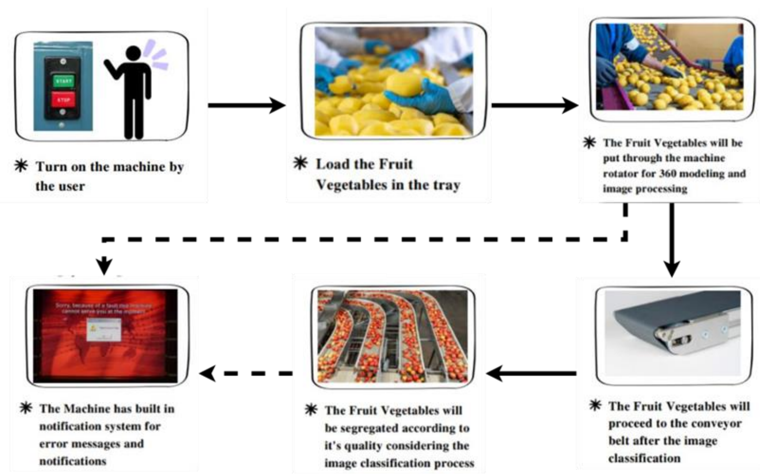


Figure 8 illustrates the step-by-step process of the proposed sorting machine, detailing the workflow from initial operation to the built-in notification system. The solid lines represent the standard flow of operations, showing how each stage progresses sequentially during the grading and sorting process. In contrast, the broken lines indicate alternative paths that occur when an error is detected, triggering the notification system to alert the user. This design ensures that the machine operates efficiently while providing immediate feedback to address issues, maintaining accuracy and minimizing disruptions in the sorting process.

Figure 9

Data flow diagram level 0

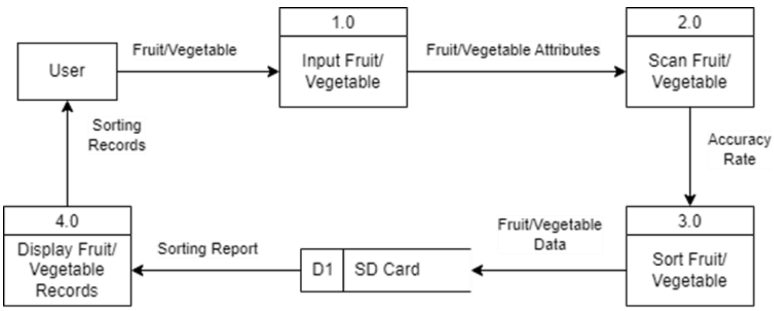


Figure 9 presents the processes and flow of data within the proposed sorting machine. Initially, the user inputs a specific vegetable into the machine, which then scans the vegetable's attributes, such as size, color, and quality. In process 2.0, the system evaluates the scanned data and determines the vegetable's accuracy rating, assigning it to the appropriate category for sorting. Following the sorting decision, the system records the data onto the SD Card for storage and future reference. Finally, the sorting results are displayed to the user, providing a clear summary of the processed vegetables and ensuring transparency and traceability in the sorting operation.

Figure 10

Data flow diagram level 1

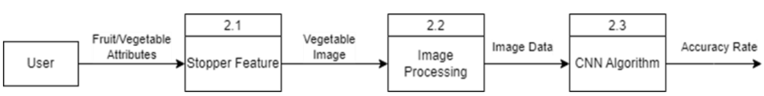


Figure 10 provides a detailed explanation of the sub process in process 2.0, specifically focusing on the scanning procedure. When the user inputs a particular vegetable onto the conveyor, the stopper feature acts as a temporary queue to position the vegetable accurately for image capturing. The camera then captures the vegetable's image, which is subsequently processed through the CNN algorithm. This algorithm analyzes the vegetable's attributes, such as size, color, and quality, to determine its category. The captured data are used to train the CNN model, improving the system's accuracy in grading and sorting the vegetables for future operations.

Figure 11

Entity relationship diagram

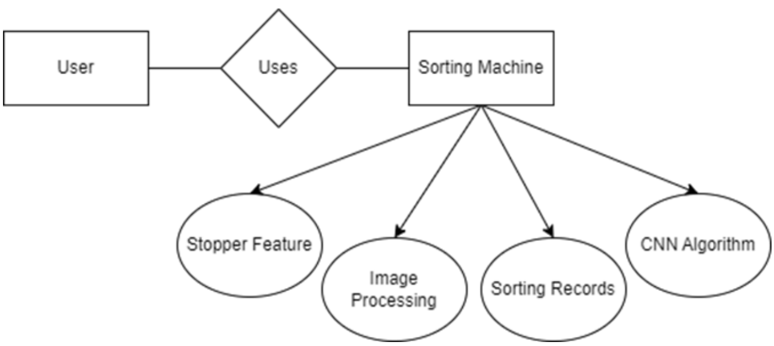


Figure 11 illustrates the functional relationship between the user and the sorting machine, highlighting how they interact during the sorting process. When the User inputs a particular vegetable into the machine, the vegetable is first detected by the system, triggering the stopper feature. This feature temporarily holds the vegetable in place and signals the start of the image processing procedure. The captured image data are then processed through the CNN algorithm, which analyzes the vegetable's attributes to ensure accurate grading and sorting. Once the vegetable is categorized, the system automatically records the sorted item, and the Raspberry Pi communicates the necessary data to the Arduino, which executes the physical sorting actions. This coordinated workflow ensures efficiency, accuracy, and seamless interaction between the User and the machine.

Figure 12

System output

Figure 12 illustrates the output of the developed system, showing that the prototype includes a notification system that displays processes, errors, records, and the AI’s confidence percentage in grading the vegetables. The system’s data are stored on an SD card, making them easily accessible and viewable through the LCD monitor during the sorting process. The design was developed with the user in mind, ensuring that the system’s functions and processes are easy to understand and operate.

After installing the essential software applications, the training phase of the system followed a step-by-step procedure. The first step involved gathering images to be used for training and compiling them into a single directory. Following this, the developers proceeded with the coding phase to implement the CNN algorithm and prepare the system for accurate detection and classification.

Figure 13

Validation and accuracy graph

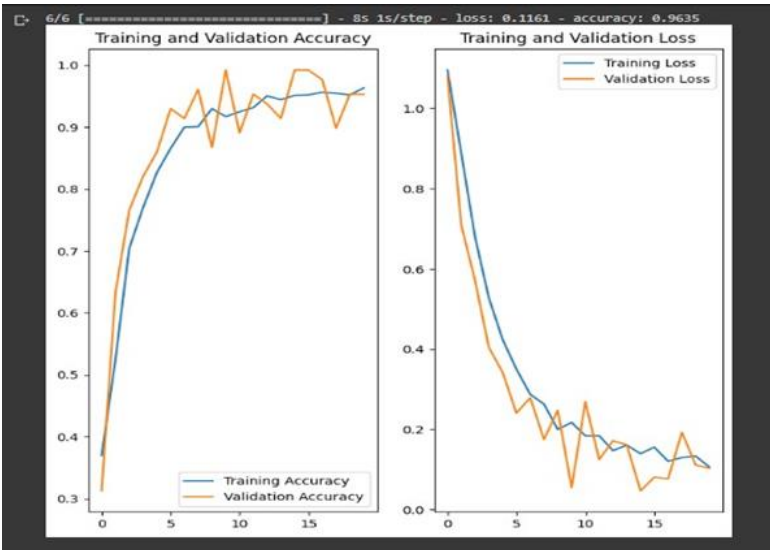


Figure 13 presents the graphical representation of validation and accuracy during training. To determine whether the model is effectively learning and generalizing the data, the validation loss should ideally be higher than the training loss. Validation accuracy helps monitor the model's generalization ability and detect overfitting or underfitting. As the model learns, validation metrics typically improve; however, if the model begins to overfit the training data, these metrics may stop improving or even start to decline.

Accuracy is a common evaluation metric used in classification tasks and indicates the percentage of correctly classified samples. It is essential to monitor accuracy on both the training and validation sets. If the model performs well only on the training data but fails to generalize to the validation data, the validation accuracy may plateau or decrease. This suggests that the model is overfitting, memorizing the training data rather than learning the underlying patterns. A significant gap between training and validation accuracy is a strong indicator of overfitting.

The validation and accuracy graphs serve as valuable visual tools for understanding and analyzing the model's performance throughout the training process. They assist in making informed decisions about model tuning, detecting overfitting, and evaluating the model's generalization capability.

Figure 14

Accuracy test

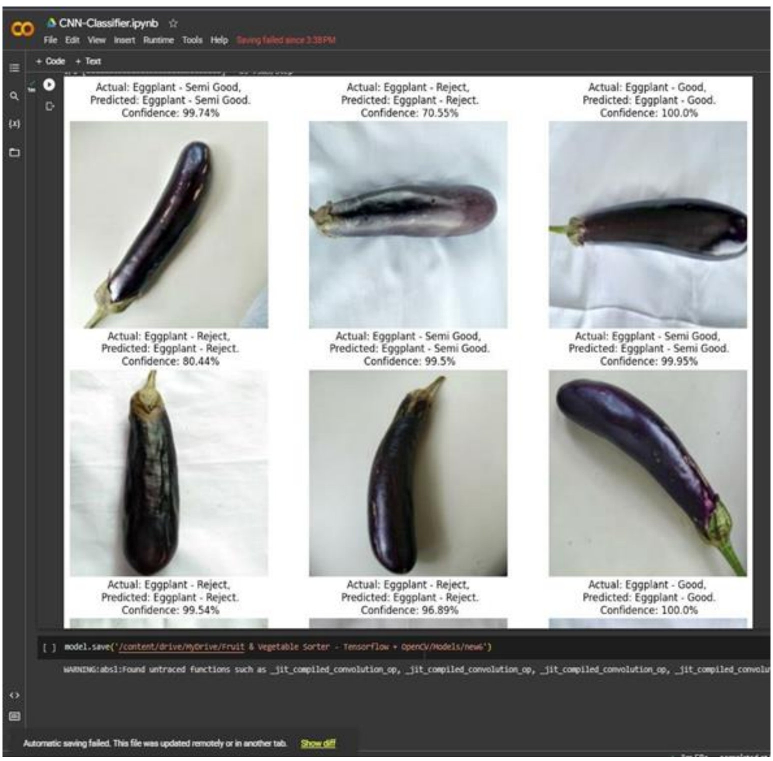


Figure 14 shows the model's confidence percentage. This phase demonstrates how the model is tested using unseen data to determine the effectiveness of the training process. The actual image displays the real appearance of the vegetable, the Predicted Image shows the model's prediction regarding the vegetable's grade, and the confidence percentage indicates how certain the model is about its prediction.

Evaluation of the System

This section shows the results of the survey evaluation. Following ISO25010, the criteria for evaluation are functional suitability, performance efficiency, compatibility, usability, reliability, security, maintainability, and portability. Each criterion has five questions that the respondents rated.

Table 3 presents the materials, tools, and equipment used in the development of the prototype. It includes the quantity and respective prices of each item, along with the total cost of producing the system. The researchers conducted a cost-benefit analysis to evaluate the feasibility of implementing the machine.

Table 3

Cost benefit analysis of the prototype

Materials, Tools & Equipment	Quantity	Unit Price	Total Cost
Raspberry Pi 4B Kit	1	12,000	12,000
-Power Supply	1		
-SD Card	1		
-SD Card Reader	1		
Arduino Mega 2560	1	1,117	1,117
LM 2596 Buck Converter	1	48	48
Buzzer	1	49.75	49.75
DIY Wires	2	224.75	449.50
Servo	1	630	630
Stepper Motor	2	2,200	4,400
Stepper Driver TB 6600	2	2,600	5,200
Web Cam 720P	1	900	900
IR Sensor	2	29	58
Bearing	4	90	360
1 x 2 Tubular	1	219	219
Angle Bar 1Meter	2	80	160
Round Bar 8MMx1/2M	7	35	259

Materials, Tools & Equipment	Quantity	Unit Price	Total Cost
Stepper Shaft Coupler	2	49.75	99.5
PVC Reducer	4	25	100
PVC Coupling	8	25	200
Chain Adjuster	2	35	70
Acrylic Sheet 2mm	1	1,900	1,900
Acrylic Sheet 3mm	1	960	960
Conveyor Belt	1	378	378
10mm Bolt & Nuts	8	10	80
Cable Tie 1 pack	1	150	150
Junction Box	3	75	225
LED	3	98	294
Push Button 1 pack	1	149.75	149.75
12V DC Power Supply	1	184	184
SPDT Switch	1	49.75	49.75
Prototyping Board	1	25	25
220 ohm resistor 1 pack	3	74.67	224.01
Ultrasonic Sensor	3	39.75	119.25
PVC Pipe	1	195	195
Basket	3	10	30
Sprocket	2	120	240
Chain	1	247	247
Total Amount			**31,536.76**

Table 4

Overall weighted mean of farmers' evaluation

Criteria	Weighted Average Mean
Functional Suitability	3.86
Performance Efficiency	3.62
Compatibility	3.77
Usability	3.77
Reliability	3.86
Security	3.96
Maintainability	3.78
Portability	3.88
Overall Weighted Mean	**3.81**

Table 4 presents the overall evaluation results from the farmers. Among the criteria assessed, Security obtained the highest mean score of

3.96, followed by Portability (3.88), Functional Suitability and Reliability (3.86), Maintainability (3.78), Compatibility and Usability (3.77), and Performance Efficiency (3.62), resulting in an overall weighted mean of 3.81, interpreted as Agree.

The evaluation of Functional Suitability yielded a mean of 3.86 (Agree), indicating that the system effectively utilizes image processing and the stopper mechanism for accurate vegetable grading.

Performance Efficiency received a mean of 3.62 (Agree), suggesting that the stopper feature enhances data processing and classification accuracy while reducing manual labor and saving time during sorting.

Compatibility achieved a mean of 3.77 (Agree), showing that system notifications are beneficial, hardware and software components are well-integrated, and the machine's dimensions and stand-alone setup are convenient and free from physical defects.

For Usability, the mean score of 3.77 (Agree) indicates that the prototype is user-friendly, the interface is easy to understand, and users can conveniently monitor sorting results on the LCD screen. Sorting records are also automatically stored and accessible through the SD card.

Reliability obtained a mean of 3.86 (Agree), demonstrating that the system functions without an internet connection, accurately performs sorting, detects errors, and maintains accessible local records.

The Security criterion, with the highest mean of 3.96 (Agree), confirms that the system operates without requiring internet access or personal data, ensuring secure and accurate vegetable sorting through reliable image processing.

Maintainability recorded a mean of 3.78 (Agree), verifying that the system's notification feature effectively supports users and that the machine operates efficiently with minimal maintenance requirements after deployment.

Lastly, Portability received a mean of 3.88 (Agree), indicating that the system is easy to install, stable in its setup, and can be improved further without the need for frequent repositioning.

Conclusion

The successful design and development of the hardware to sort eggplants into Grades A, B, and C proved beneficial and effective for the sorting process. The system's physical infrastructure and functionality were fully achieved. Additionally, the study successfully collected a dataset of eggplant samples, which served as training data for grading and sorting.

As reflected in the accuracy graph and confirmed through the accuracy test, the TensorFlow and CNN algorithms were successfully trained to recognize and classify vegetables based on texture, color, and defects. Furthermore, with all test cases passed, the proposed system met all the ISO 25010 software quality criteria.

Bibliography

Assimakopoulos, F., Vassilakis, C., Margaris, D., Kotis, K., & Spiliotopoulos, D. (2024). Artificial intelligence tools for the agriculture value chain: Status and prospects. *Electronics*, *13*(22), 4362. https://doi.org/10.3390/electronics13224362

Azmirul Hoque, & Padhiary, M. (2024). Automation and AI in precision agriculture: Innovations for enhanced crop management and sustainability. *Asian Journal of Research in Computer Science, 17*(10), 95–109. https://doi.org/10.9734/ajrcos/2024/v17i10512

Bazargani, K., & Deemyad, T. (2024). Automation's impact on agriculture: Opportunities, challenges, and economic effects. *Robotics, 13*(2), 33. https://doi.org/10.3390/robotics13020033

Bhat, I. A., Ansarullah, S. I., Ahmad, F., et al. (2025). Leveraging artificial intelligence in agribusiness: A structured review of strategic management practices and future prospects. *Discov Sustain, 6*, 565. https://doi.org/10.1007/s43621-025-01260-3

Edozie, E., Shuaibu, A. N., John, U. K., et al. (2025). Comprehensive review of recent developments in visual object detection based on deep learning. *Artificial Intelligence Review, 58*, 277. https://doi.org/10.1007/s10462-025-11284-w

Fronda, J. G. (2022). The supply chain of vegetable production in the Philippines: The case of Nueva Ecija farmers. *Open Journal of Social Sciences, 10*, 16–27. https://doi.org/10.4236/jss.2022.1013003

Gupta, G., & Kumar Pal, S. (2025). Applications of AI in precision agriculture. *Discov Agric, 3*, 61. https://doi.org/10.1007/s44279-025-00220-9

Hoque, A., & Padhiary, M. (2024). Automation and AI in precision agriculture: Innovations for enhanced crop management and sustainability. *Asian Journal of Research in Computer Science, 17*(10), 95–109. https://doi.org/10.9734/ajrcos/2024/v17i10512

ISAAA. (2023). *Pocket K No. 48: Global status of commercialized biotech/GM crops in 2022*. International Service for the Acquisition of Agri-biotech Applications. https://www.isaaa.org/resources/publications/pocketk/48/default.asp

Jamali, M., Davidsson, P., Khoshkangini, R., et al. (2025). Context in object detection: A systematic literature review. *Artificial Intelligence Review, 58*, 175. https://doi.org/10.1007/s10462-025-11186-x

Khan, Z., Shen, Y., & Liu, H. (2025). Object detection in agriculture: A comprehensive review of methods, applications, challenges, and future directions. *Agriculture, 15*(13), 1351. https://doi.org/10.3390/agriculture15131351

Kumari, K., Mirzakhani Nafchi, A., Mirzaee, S., & Abdalla, A. (2025). AI-driven future farming: Achieving climate-smart and sustainable agriculture. *AgriEngineering, 7*(3), 89. https://doi.org/10.3390/agriengineering7030089

Loresco, P. J., & Dadios, E. P. (2018, July). Fuzzy logic-based identification method for eggplant class. In *Proceedings of the 5th International Conference on Communication and Computer Engineering (ICOCOE'2018), Swiss-Garden Hotel Melaka, Melaka, Malaysia*.

Mishra, A., Bhatia, S. K., & Mishra, R. (2024). Sorting machine for fruits and vegetables for agricultural advancements using IoT. *International Journal of Research and Review in Applied Science, Humanities, and Technology, 1*(2), 40–48. https://doi.org/10.71143/6m3z1250

Prapas, I., Derakhshan, B., Mahdiraji, A. R., et al. (2021). Continuous training and deployment of deep learning models. *Datenbank Spektrum, 21*, 203–212. https://doi.org/10.1007/s13222-021-00386-8

Rathor, A. S., Choudhury, S., Sharma, A., Nautiyal, P., & Shah, G. (2024). Empowering vertical farming through IoT and AI-driven technologies: A comprehensive review. *Heliyon, 10*(15), e34998. https://doi.org/10.1016/j.heliyon.2024.e34998

Sow, S., Ranjan, S., Seleiman, M. F., Alkharabsheh, H. M., Kumar, M., Kumar, N., Padhan, S. R., Roy, D. K., Nath, D., Gitari, H., & Wasonga, D. O. (2024). Artificial intelligence for maximizing agricultural input use efficiency: Exploring nutrient, water and weed management strategies. *Phyton-International Journal of Experimental Botany, 93*(7), 1569–1598. https://doi.org/10.32604/phyton.2024.052241

Taher, D., Solberg, S. Ø., Prohens, J., Chou, Y., Rakha, M., & Wu, T. (2017). World Vegetable Center eggplant collection: Origin, composition, seed dissemination and utilization in breeding. *Frontiers in Plant Science, 8*, 1484. https://doi.org/10.3389/fpls.2017.01484

Totin, E., van Mierlo, B., & Klerkx, L. (2020). Scaling practices within agricultural innovation platforms: Between pushing and pulling. *Agricultural Systems, 179*, 102764. https://doi.org/10.1016/j.agsy.2019.102764

Valtonen, A., Saunila, M., Ukko, J., Treves, L., & Ritala, P. (2025). AI and employee wellbeing in the workplace: An empirical study. *Journal of Business Research*, *199*, 115584. https://doi.org/10.1016/j.jbusres.2025.115584

Xiao, F., Wang, H., Li, Y., Cao, Y., Lv, X., & Xu, G. (2023). Object detection and recognition techniques based on digital image processing and traditional machine learning for fruit and vegetable harvesting robots: An overview and review. *Agronomy*, *13*(3), 639. https://doi.org/10.3390/agronomy13030639

Zheng, Y.-Y., Kong, J.-L., Jin, X.-B., Su, T.-L., Nie, M.-J., & Bai, Y.-T. (2018). Real-time vegetables recognition system based on deep learning network for agricultural robots. In *Proceedings of the 2018 Chinese Automation Congress (CAC)* (pp. 2223–2228). IEEE. https://doi.org/10.1109/CAC.2018.8623610