Journal of Engineering and Sustainable Development
Vol. 28, No. 04, July 2024
https://jeasd.uomustansiriyah.edu.iq/index.php/jeasd
ISSN 2520-0917
Research Article
https://doi.org/10.31272/jeasd.28.4.3

Design and Development of Eggplant Fruit and Shoot Borer (Leucinodes Orbonalis) Detector Using Near-Infrared Spectroscopy

Maria Patrice L. Lajom[1], Joseph Paul R. Remigio[2], Edwin R. Arboleda[3], Rhen John Rey R. Sacala[4*]

[1,2,3,4] Department of Computer, Electronics and Electrical Engineering, Cavite State University, Don Severino Delas Alas Campus, Indang, Cavite, Philippines
[1]https://orcid.org/0009-0008-9632-8938
[2]https://orcid.org/0009-0009-9988-8727
[3]https://orcid.org/0000-0001-9371-8895
[4]https://orcid.org/0009-0007-4989-1974
[*]Email: rhenjohnrey.sacala@cvsu.edu.ph

Article Info

Received 15/11/2023
Revised 25/03/2024
Accepted 28/03/2024

Abstract

An Eggplant Fruit and Shoot Borer (EFSB) is a disease that affects the entirety of the eggplant fruit if not detected. Hence, a detector was proposed in the form of a handheld gun. It was designed and developed to non-invasively classify eggplant fruits that are non-infested and infested with EFSB. Using an Arduino Nano as its microcontroller and a near-infrared spectroscopy (NIRS) module, insect infestation is determined and displayed through its OLED display. Measured reflectance data through the NIRS module of the detector is then stored inside a MicroSD module for further use. Since the prototype was developed for online monitoring, portability was given of utmost importance, pattering the design in the form of a handheld gun, inside of which was powered by a 9V rechargeable battery. The 3D-printed chassis of the detector houses the aforementioned components and modules, alongside with switches for power and near-infrared detection. Through Support Vector Machine (SVM), the classifier model was trained and developed using Jupyter and was extracted as a C++ code for the Arduino Nano module. Compared with a farmer's traditional performance in terms of accuracy, precision, and speed, the prototype performed better with an accuracy of 84%, precision of 72.83%, and an average speed of 9.736 seconds.

Keywords: Insect Infestation; Spectroscopy; Support Vector Machine; Machine Learning

1. Introduction

Eggplant (Solanum melongena L.) is an annual vegetable crop popular in both Asian and other Mediterranean countries [1]. Known as “talong" in the Philippines and "brinjal" to India and Bangladesh, eggplant is often characterized as one of the most popular, most important, and inexpensive crops in Asia [2], [3] and is also known to be one of the very few vegetables that are sold cheaply in both rural and urban poor areas [4]. Eggplant is considered the 5th most important crop in the world with production accounting for 50 million yearly, producing a value of more than 10 billion US Dollars annually. Currently, the Philippines ranks as the 10th eggplant-producing country, with over 0.24 million tons of eggplant produced. In the Philippines, eggplants produced annually account for 2.6 billion Philippine pesos, making it the number one vegetable crop in terms of volume and production within the country [5]. For this reason, eggplant continually proves to be an important source of income especially for small-owned and resource-poor farmers.

One of the biggest problems faced by farmers is the heavy damage caused by various insect-pest diseases. This problem reflects heavily on their expected earnings [6]. Widespread insect infestation of Leucinodes orbonalis Guenée, better known as the Eggplant Fruit and Shoot Borer (EFSB), has been the biggest problem for eggplant production in Asia. EFSB causes major damage to the eggplant at its larvae stage by producing holes and tunneling inside the fruit [7], feeding on internal tissues [8]. Parvin [9] stated that a newly hatched larvae may directly bore into an eggplant's fruit resulting in an increased number of larvae per fruit. After boring into the nearest fruit, the larvae clog the entrance hole with black-colored excreta and remain concealed inside that particular part [10], [11]. Owen [10] and Mollah [11] added that due to the nature of the infestation entrance holes are not outright visible, rendering it damaged internally. Due to this, the fruit will be deemed unmarketable and unfit for human consumption. The spectroscopic technique is generally considered a well-known technique used for monitoring and detecting insect infestation in the internal quality of fruit [12]. One of the well-known spectroscopic techniques feasible for non-destructive spotting of an insect's presence and damage in fruits is near-infrared spectroscopy (NIRS) [15]-[17]. NIRS is an analytical technique that measures the absorption bands produced by the overtones and combined excitations to characterize molecular structures. Due to recent developments, NIRS has advanced in recent years and is widely used in various fields such as bioscience, medical diagnosis, food science, and agriculture [18]-[13]. Detecting insect infestation through the use of NIRS can be achieved by indirectly identifying the changes in the spectral properties of infested tissues due to internal darkening, dehydration, or contamination [14]. For it to be fully utilized in the detection and spotting of pest infestation, Jamshidi [15] noted that the proper wavelength range and proper identification of the optical quantification mode are to be considered necessary. Aside from NIRS, another spectroscopic technique used in agricultural settings is the Fourier transform infrared spectroscopy (FTIR). While NIRS is concerned with the absorption of electromagnetic radiation within the 780-2500 nm wavelength range, FTIR uses light in the mid-infrared region (400-4000 cm-1) [16]. Both NIRS and FTIR are deemed as a non-destructive analysis method that focuses on the absorption, reflection, emission, and diffuse-reflectance of light. However, the analysis made by NIRS is considered simple and fast enough that it would only take between 15 and 90 seconds to fully analyze an acquired sample [17]. In addition, NIRS is also used for noninvasive and on-line monitoring. One of the major advantages of NIR-based technologies is their ability to provide fast measurement speeds allowing concurrent measurement for several constituents [18]. Another advantage of NIRS is its versatility in use for both qualitative and quantitative analysis due to its high signal-to-noise ratio and low cost of implementation, yielding excellent performance [19]. Because of this, NIRS is often used as a non-destructive technique for quality control practices in the food and agricultural industry [20]. Such methods of detection require the need for a machine to make models that can train themselves to improve, perceive complex patterns, and look for solutions to present-day problems by utilizing previous data [21]. Machine learning utilizes four distinct approaches namely supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning [22]. Supervised learning is a machine learning model for gathering input-output relationship information for a system and is established from a given set of paired input-output training data [23]. Supervised machine learning are algorithms that require external assistance. According to Gupta, Mishra, Singhal, and Kumar [24], as a subset of machine learning, supervised learning automatically adapts to experience to further improve itself without any explicit programming. The training dataset contains an output variable that should be predicted or classified. Furthermore, all models deduce pattern-like information from the training dataset and use them for the classification or prediction of the test dataset. Moreover, by decreasing the number of resources that are required in describing large data sets, feature extraction comes into play [25]. Supervised learning can be categorized into two: classification and regression [26]. Classification is the allocation of data into the given groups defined on the dataset depending on their features [26]. The work of Gurung [22] states that classification can use classification and decision trees, support vector machines (SVM), logic regression, random forest, artificial neural networks (ANNs), or other models. In addition, some of the most known supervised machine learning algorithms are neural networks (feed and forward recurrent), support vector machines, random forests, self-organizing maps, and Bayesian networks [27]. In binary classification, SVM creates a linear hyperplane (a decision boundary) to separate instances of one class from the other [28]. Furthermore, SVMs can efficiently execute non-linear classification by utilizing kernel tricks, implicitly plotting their inputs into high-dimensional feature spaces [29]. SVM creates margins in the middle of classes in a way that distances between the margin and classes are at maximum and thus, keeps the classification error at minimum. Consumers are considered as the driving force of the market. The food industry has been continuously receiving demands from consumers regarding quality product assurance. A consumer's trust is vital, and it has been proven to continually diminish due to issues in food quality [30]. Therefore, improvement of safety monitoring of food quality is a must. Quality inspection is badly needed to ensure the integrity of the product on hand and it is a crucial task that involves evaluation and determining the acceptability of a product [31], [32]. In Bangladesh, 80% of the farmers identify insect infestation as a criterion for judging marketable eggplants while 20% by shape and size [33]. This is a form of visual inspection. Traditional methods such as visual methods of detection are regarded as subjective, laborious, and time-consuming while manual sorting is now viewed as inadequate for the detection of insect pests and hidden damages in the internal quality of a crop [34]. Magwaza et al. [35] stated that the latest trend in the agribusiness industry is to decline the usage of subjective assessment and increase the adoption of an objective, quantitative, and non-destructive technique for quality assessment instead. Other techniques such as machine vision systems involve digital image processing which can be used for the detection of infestation within fruits and vegetables through identification of small exit holes on surfaces. However, several studies have stated that the use of such a technique faces challenges involved in penetrating visible light inside of fruits and is considered unreliable because insect damage is difficult to distinguish and is often misinterpreted as surface damage when marks are similar [36]-[40]. Therefore, this study would seek to develop a device using the NIRS technique capable of detecting EFSB infestation in eggplant fruits non-invasively. The study has the potential to help the following individuals:

Farmers - The potential brought about by the study has the potential to greatly help farmers identify and quantify losses during the postharvest stages caused by EFSB infestation. Successful removal of infested marketed eggplants could gain the trust of customers, leading to an increase in sales within the eggplant market. In addition, NIR spectroscopy increases the accuracy of classification as the device will be able to determine infestations unnoticeable by the naked eye. When compared to objective assessments done by a device, subjective assessments are usually more prone to error. Visual inspection, when done multiple times, might lead to inaccurate classification (e.g., infested eggplant fruits might be labeled as non-infested and vice versa). Furthermore, rotting may occur when an eggplant is infested. Failure to check for infested eggplants when repacking for market delivery might affect the quality of healthy eggplants contained in the same pack. Lastly, opening the eggplant for double-checking infestations is not necessary as this might lead to the larvae escaping and possibly evolving into a moth. When the larvae enter their moth stage, they will cause greater harm to the farm as it lays new eggs and thus, infestation in growing crops is possible. For this reason, this study will pave the way for farmers to start using a noninvasive method suitable for proper assessment.

Vendors – As a part of their products sold, vendors can also benefit from the study, particularly the small-scale fruit and vegetable enterprises. Through this, assurance can be made to their daily patrons and customers, ensuring the products sold are safe for human consumption. Such action is a big step in gaining their trust, thus preventing future profit losses.

Consumers – As they are the ones consuming the products on hand, quality assurance is a must when purchasing food products in the market. Consumers are very particular and meticulous when it comes to buying fruits and vegetable crops in the market. Consumers are willing to pay higher prices as long as they are assured that the eggplant fruit is free from infestation. For consumers, buying infested eggplants will mean that a part of the fruit containing the larvae is regarded as a waste and therefore, needs to be cut and removed before consumption.

1.1. Paper Subdivisions

Throughout the paper, sections are structured as follows: first, the aforementioned Introduction sets the stage by providing relevant background information and discussing the significance and contribution of the paper. Next, the Methodology section details the framework of the study, alongside the methods and materials employed. The Results and Discussion section then presents the findings, analyzes their significance, and interprets them accordingly. Finally, the Conclusions section summarizes key takeaways and offers potential future endeavors for further research.

2. Materials and Methods

2.1. Conceptual Framework

The conceptual framework of the study is shown in Fig. 1. An NIRS sensor with six different wavelengths was used to quantitatively measure the color and intensity of the reflected light from the samples. These reflectance measurements from the samples served as the input for the training of the SVM classifier model. The trained classifier model determines if a sample is infested by EFSB based on the reflectance measurement acquired from the spectral sensor. The algorithm for the classifier model was programmed using Jupyter Notebook and was administered in Arduino Nano using C++ as its programming language. A series of classification tests and data storage tests for the hardware and software was done to ensure proper functionality.

2.2. Origin of Samples

As shown in Fig. 2 and Fig 3., the samples used in this study were obtained from Michael's Eggplant Farm in Quezon Province and John's Vegetable Farm in Imus City respectively. A total of 300 eggplant fruits were acquired, whereas, 250 samples were allocated for the training dataset and testing dataset. The training dataset comprises 70 percent (175 samples) of the 250 samples while the remaining 30 percent (75 samples) were used for the testing dataset. The remaining 50 eggplant fruit samples were allocated for the final evaluation of the device.

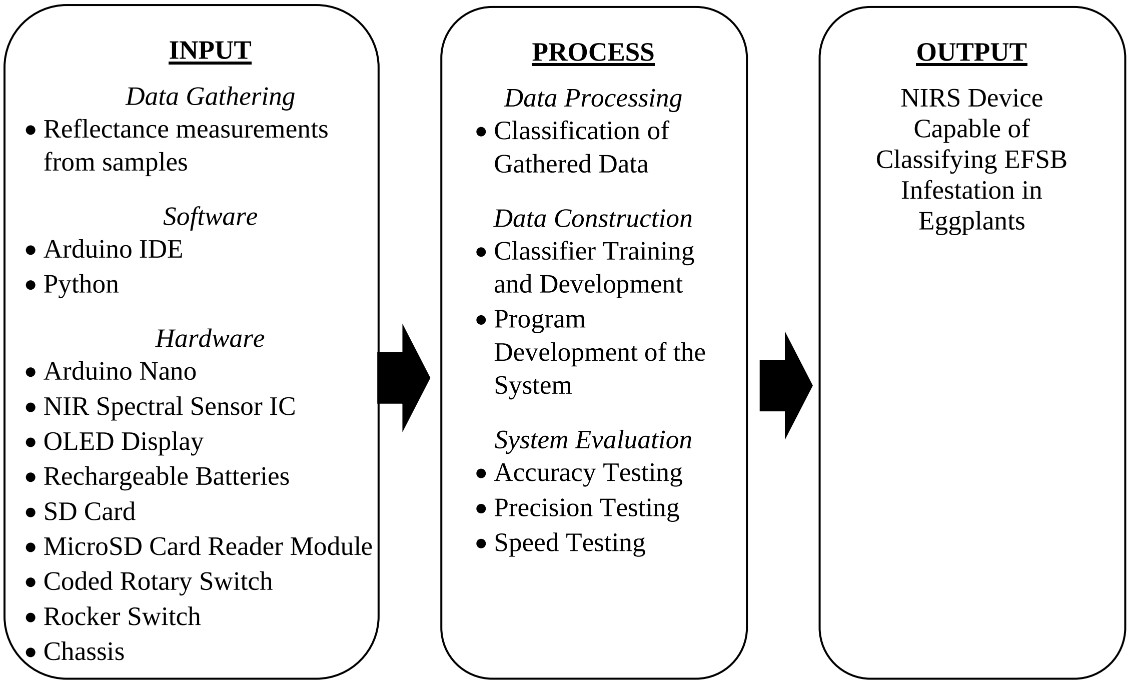

Figure 1. Conceptual Framework of the Study

Figure 2. Michael's Eggplant Farm in Quezon Province

Figure 3. John's Vegetable Farm in Imus City

2.3. Developing the Circuitry of the EFSB classifier

The circuitry and enclosure of the EFSB classifier were developed using the following hardware and software:

2.3.1. Arduino Nano

The Arduino Nano, as shown in Fig. 4 served as the main controller or the brain of the device. It is a small and breadboard-friendly board based on the ATmega328p. The Nano features a Mini-B USB connector and comes with pin headers for easy attachments. The ATMega328 processing unit runs with a clock speed of 16 MHz, features 32 KB of flash memory (of which 2KB is used by the bootloader), and has an operating voltage of 5V. The Nano board also features 8 analog input pins and 22 digital I/O pins (6 of which are PWM pins).

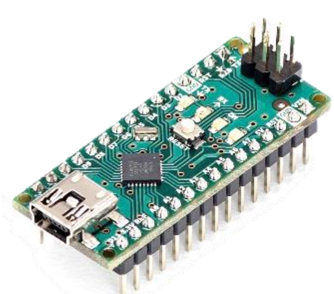

Figure 4. Arduino Nano

2.3.2. NIR Spectral Sensor IC

To acquire the reflectance measurements of the samples, SparkFun's AS7263 Near Infrared (NIR) Spectral Sensor IC model was used as shown in Fig. 5 to detect the wavelength range at 610-860nm of light. Specifically, the NIR Spectral Sensor IC contains 6 near-IR wavelength channels: 610, 680, 730, 760, 810, and 860 nm of light.

Figure 5. AS7263 NIR Spectral Sensor IC

2.3.3. OLED Display

An OLED display Module (as shown in Fig. 6) with a white display was used for displaying the interface and output of the device. The 128x64 OLED Display Module, 1.3-inch size of the screen is suitable for creating a portable and compact prototype. The screen also contains 4 pins for VCC, GND, SCL, and SDA.

Figure 6. 128x64 OLED Display Module

2.3.4. MicroSD Card Reader Module

A MicroSD card reader module as illustrated in Fig. 7 was used to store reflectance measurement data that is used for training and testing the classifier model. Six pins make up the control interface: GND, VCC, Master In Slave Out (MISO), Master Out Slave In (MOSI), Serial Clock (SCK), and Chip Select (CS). The MOSI and MISO are used for the SPI interface. The GND pin connects to the ground. The VCC pin connects to the power supply. SCK is the connection for the SPI.

Figure 7. MicroSD Card Reader Module for Arduino

2.2.4. Rechargeable LiPoly Battery

A 400mAh (9V) rechargeable Lithium-Polymer battery with a charging voltage of 4.25V was used as the power supply of the device as shown in Fig. 8. It has a distinct USB charging port (can be charged with a USB cable; no additional charger is needed) and only takes 1.5 hours to be fully charged.

Figure 8. Znter 9V 400mAh USB Rechargeable LiPoly Battery

2.2.5. Coded Rotary and Rocker Switch

Switches as shown in Fig. 9 such as a rotary switch were used to toggle between the settings and to trigger the scanning and classifying function of the prototype; while a single rocker switch was used to turn the prototype on and off.

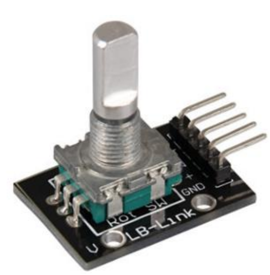

Figure 9. Coded Rotary Switch (Top) and Rocker Switch (Bottom)

2.2.6. Chassis

A 3D printer was used for creating the chassis of the device. The PLA filament used in 3D printing is suitable for the construction of handheld portable devices due to its affordable, safe, and outstanding material properties.

2.2.6. Arduino IDE

The Arduino Integrated Development Environment (IDE) acts as the software for all the released Arduino boards. Specifically, the Arduino IDE contains a text editor for writing programs, a message area, a text console, a toolbar with buttons for frequently used operations, and several menus. In this study, the program for the prototype was constructed in the Arduino IDE and was uploaded to the Arduino Nano microcontroller board.

2.2.7. Jupyter Notebook

The Jupyter Notebook is a free and open-source software application that caters to three primary programming languages: Julia, Python, and R, hence the name Jupyter. The software is used to create and share documents with live code, equations, visualizations, and texts. Jupyter also comes with the IPython kernel, enabling the possibility of programming in Python. In this study, the Jupyter was used to develop the program for the deployed model.

2.3. Block Diagram of the Prototype

The block diagram of the prototype is shown in Fig. 10. The prototype is powered up once the switch is connected to the power source. The power source was composed of a single 9V lithium-polymer rechargeable battery cell. The rocker switch, rotary switch, and spectral sensor served as the input of the Arduino Nano hardware. The rotary switch served as the start button. If pressed, the program for acquiring the reflectance measurements and classifying of samples commences. The OLED Display's function was to display the classification result.

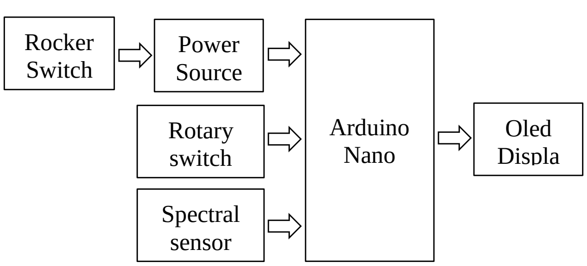

Figure 10. Block diagram of the EFSB detector

2.4. Schematic Diagram of the Prototype

The schematic diagram of the prototype is shown in Fig. 11, while the labeled pinouts of the Arduino Nano are shown in Fig. 12. The rocker switch is connected in series with the 9V power supply. The 9V power supply is connected to the VIN and GND pin of Arduino Nano. To establish a common ground connection, all the GND pins of the components (NIR sensor, 9V power supply, MicroSD card module, and OLED Display) are connected to the GND pins of the Nano. The Arduino Nano supplies 3.3V to the NIR sensor and at the same time, supplies 5V to the OLED display. Both SCL ports of the OLED display and NIR sensor were connected to the designated SCL pin (A5) of the Nano. Similarly, both the SDA ports of the said components are also connected to the designated SDA pin (A4) of the Nano. This allows the Arduino Nano to establish communication and control both components. To synchronize data transmission, the SCK pin of the MicroSD card module was connected to the SCK digital pin (D13) of the Nano. The MISO and MOSI pins of the said component were also connected to the designated MISO (D12) and MOSI (D11) pins of the Nano. The MOSI pin was used by the Nano to send information to the MicroSD card module while the MISO pin was used to receive information coming from the MicroSD card module. The CS pin of the MicroSD card module connected to the CS pin (D10) of the Nano allowed the MicroSD card to be selected and establish communication with the Nano. The CLK pin of the rotary switch connected to the digital pin 9 (D9) of the Nano was used to determine the amount of the switch's rotation. Similarly, its DT pin connected to the digital pin 8 (D9) of the Nano was used to determine the direction of the switch's rotation. Lastly, the SW pin of the rotary switch connected to the digital pin 7 (D7) of the Nano was used to determine if the rotary switch is pushed and thus, sets the voltage to LOW.

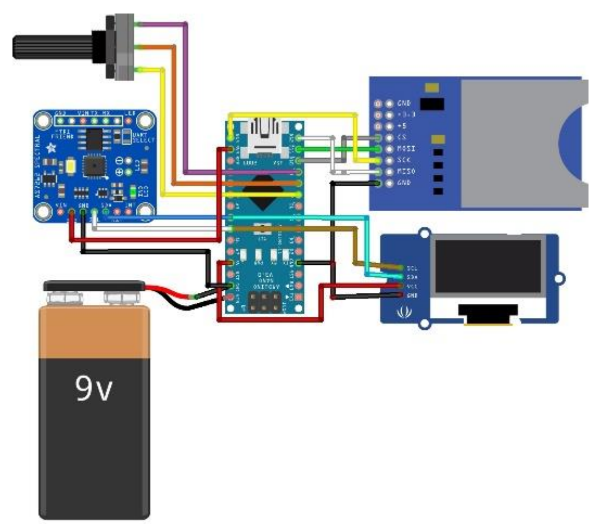

Figure 11. Schematic diagram of the EFSB detector

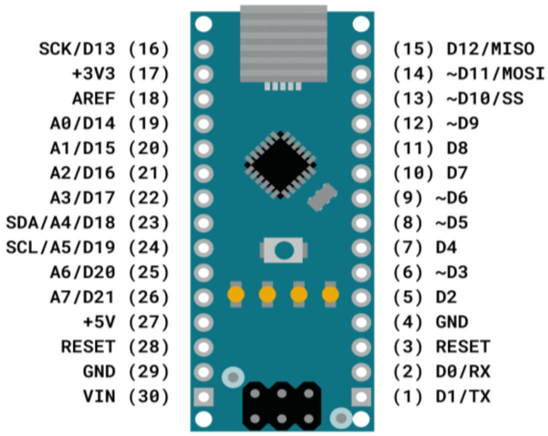

Figure 12. Arduino Nano Pinout

2.5. The Prototype

Both the external view and dimensions of the output prototype are shown in Fig. 13. The NIR sensor was mounted on the frontmost side of the prototype. This placement allows easy and successful reading of reflectance measurements from the sample. The OLED display which shows the output of the device was mounted on the top of the prototype. The rocker switch which turns the device on and off together with the SD card holder was mounted on the right side of the prototype. The coded rotatory switch which acts as a trigger for the classifier to scan and classify the samples was mounted on the frontmost side of the handle. Finally, the charging port was placed on the back side of the handle. The placement of the components was done in such a manner as to grant easy access to the user. The placement of the NIR sensor, OLED display, Arduino Nano, and the power supply inside the prototype is shown in Fig. 14.

Figure 13. External Views of the Prototype with Dimensions

Figure 14. Internal View of the Prototype

2.6. SVM Classifier Model Training and Deployment

The workflow for the construction of the SVM classifier model is shown in Fig. 15. A dataset was obtained using the prototype pre-programmed to acquire reflectance measurements in the reflectance mode. To train the SVM classifier model, both the training and testing dataset contained predetermined (infested and noninfested) samples of eggplant fruits. An expert farmer served as the guide in identifying infested and noninfested eggplants present on the farm. Each sample was divided and scanned into five different regions. After acquiring the reflectance measurements, each sample was cut in half to determine if the predetermined classifications were correct. A single validated region's reflectance measurement is retained afterward.

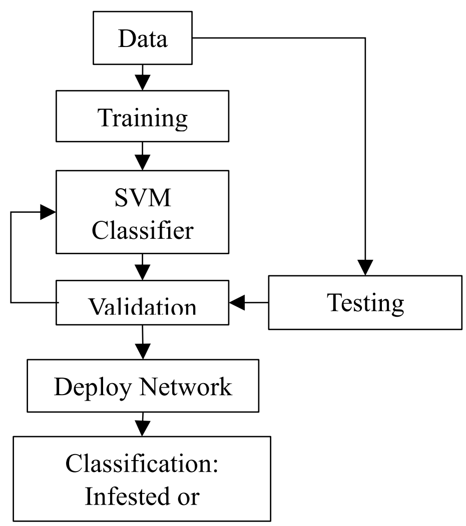

Figure 15. Workflow for the SVM Classifier Model Construction

For the feature selection, the acquired reflectance measurement data of the six (6) near-infrared channels (610 nm, 680 nm, 730 nm, 760 nm, 810 nm, and 860 nm) of the spectral sensor was plotted and shown in Fig. 16. Based from the plot, it was deduced that all channels were relevant for training the preferred machine learning algorithm. Hence, all of the data were retained and used to train the SVM classifier model.

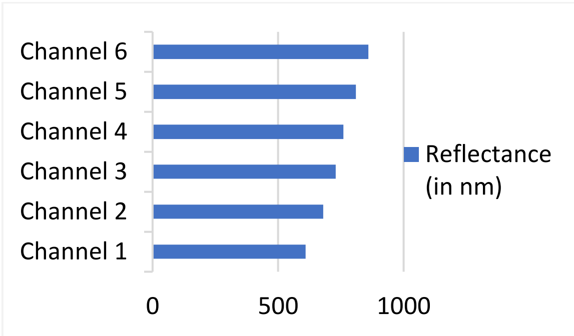

Figure 16. Plotted Reflectance AMeasurement Data for Feature Selection

The workflow of the SVM classifier model training is shown in Fig. 17. Python offers a large and potent collection of packages like NumPy, SciPy, Pandas, Scikit-learn, and others that are necessary for data science and machine learning. The NumPy (Numerical Python) module was imported into the program to perform mathematical and logical operations on arrays. Micromlgen, a package built on NumPy, supports porting of commonly used machine learning classifiers such as SVM to plain C. Scikit-learn, a library for machine learning that was also built on NumPy, offers effective iterations of many popular machine learning algorithms such as SVM and provides a selection of efficient tools for machine learning. The function train_test_split() imported from sklearn was used to split the given dataset into the training and testing data. Once the best C parameter and kernel type were identified, the SVM classifier model was created using the function SVC(). The function svm. fit() was used to train the classifier using the dataset. The function svm. predict() was used to predict the classifier's accuracy. The function print(port()) was used to extract and convert the SVM classifier program to C to deploy it in Arduino Nano. The printed accuracy of the classifier is shown in Fig. 18. The network was deployed after validating its accuracy.

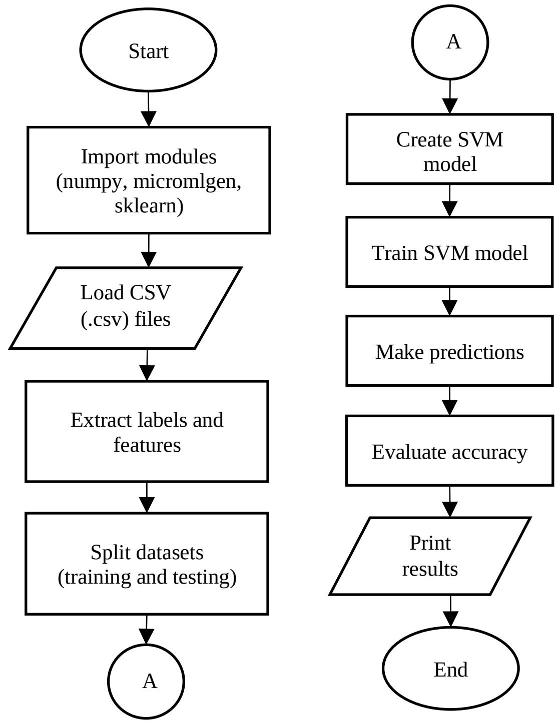

Figure 17. Workflow of SVM Classifier Model Training

Figure 18. SVM Classifier Model Training Accuracy

2.7. Development of the System Software

The development of the Python program is shown in Fig. 19. The rocker switch functioned as the ON/OFF switch of the prototype. The prototype is powered up once the rocker switch is turned on. Necessary modules were imported into the program. The model. h module contained the program for the SVM classifier. Similarly, the <U8g2lib.h> module was used for the OLED display control, AS726X.h for the NIR sensor control, and <SPI.h> and <SD.h> for the SD card control. Several global variables were defined and used throughout the program. The functions for start-up screen 1, start-up screen 2, main menu, screen predict, classify, and scan sample was initialized. The Arduino Nano's GPIO pins were assigned based on the required pins of the components of the prototype.

When the prototype is turned on, the start-up screen 1 shown in Fig. 20 was displayed followed by the start-up screen 2 and the main menu shown in Figs 21-22. A time interval of two seconds was utilized before the screen changed its display. The rotary switch acted as the component that introduces an interrupt to the whole program flow. The user was directed to the main menu every time the rotary switch was rotated either clockwise or counterclockwise. On the other hand, the scanning and classification of samples commenced every time the rotary switch was pushed. The result of the classification displayed on the OLED screen is shown in Fig. 23-24. The device is turned off simply by using the rocker switch that was used to turn it on.

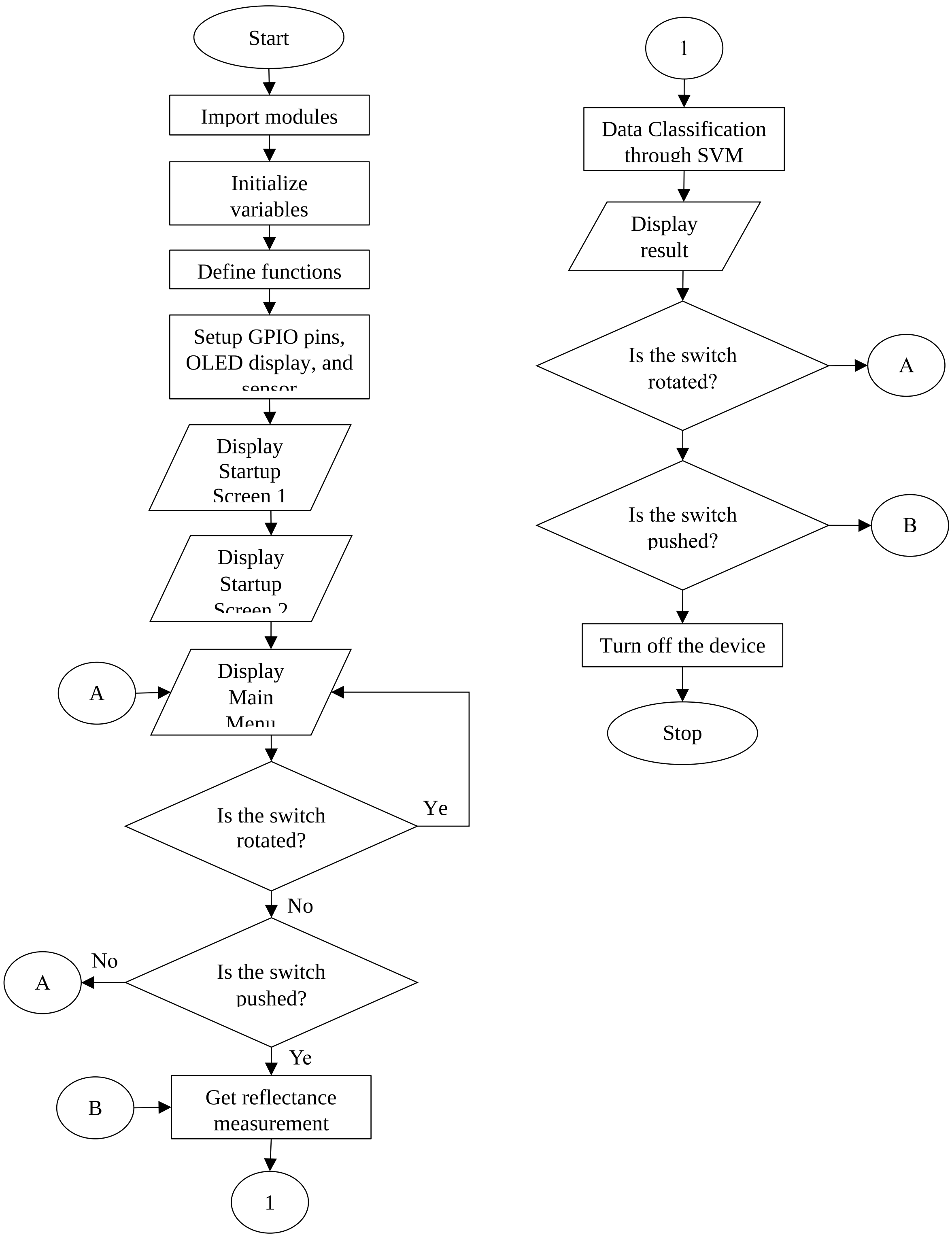

Figure 19. Development of the system's program flowchart

Figure 20. Start-Up screen 1 of the prototype

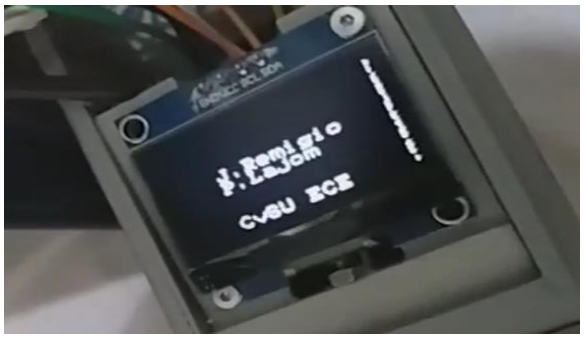

Figure 21. Start-Up screen 2 of the prototype

Figure 22. Main Menu screen of the prototype

Figure 23. Uninfested classification result of the prototype

Figure 24. Infested classification result of the prototype

2.8. Evaluation of the System

Samples of eggplant fruits that underwent evaluation of the system and traditional inspection are shown in Fig. 25. Such results of the evaluation were subjected to a Confusion Matrix, also known as an error matrix, which is a table often used to evaluate the performance of an algorithm in machine learning classification. A confusion matrix table presents the summary of the actual versus the predicted results of the classifier. Specifically, a confusion matrix table includes the total number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) obtained by the classifier model. In this study, the data from the confusion matrix was used to compute the precision of a binary classifier. Precision refers to the number of instances that are relevant among the total instances the model retrieved. A sample confusion matrix table used for a binary classifier is shown in Table 1.

Table 1. Sample confusion matrix table for a binary classifier.

		Predicted	
		Uninfested	Infested
Actual	Uninfested	TN	FP
	Infested	FN	TP

Where:

True Positives (TP) = Correctly classified infested eggplants

True Negatives (TN) = Correctly classified uninfested eggplants

False Positives (FP) = Incorrectly classified infested eggplants

False Negatives (FN) = Incorrectly classified uninfested eggplants

Figure 25. Samples of eggplants subject to evaluation

Using the confusion matrix, the precision of the model is represented by Equation (1).

$$Precision = \left(\frac{TP}{TP+FP}\right) x\ 100 \quad (1)$$

A McNemar's Test also referred to as the "within-subjects chi-squared test", is a statistical test used for paired nominal data. Specifically, a McNemar's Test is used to compare the prediction power of two machine learning (or statistical) models. In this study, McNemar's test was used to determine if there is a significant difference between the performance of the two classifiers in terms of accuracy. Accuracy of the model refers to the amount of data points that have been correctly predicted out of all the data points. The sample confusion matrix table used in McNemar's test is shown in Table 2.

Table 2. Sample confusion matrix table used in McNemar's Test

		Classifier Model 2	
		Correct	Incorrect
Classifier Model 1	Correct	A	B
	Incorrect	C	D

Where:

A = Correctly classified samples for both classifier models.

B = Correctly classified samples for classifier model 1 and incorrectly classified samples for classifier model 2.

C = Correctly classified samples for classifier model 2 and incorrectly classified samples for classifier model 1.

D = Incorrectly classified samples for both classifier models.

Using the confusion matrix above, McNemar's Test equations are calculated through Equations (2), (3), and (4).

$$x^2 = \frac{(B-C)^2}{(B+C)} \quad (2)$$

$$DF = (r-1)x(c-1) \quad (3)$$

$$p - value = 1 - cdf(x^2) \quad (4)$$

Where:

B = Correct classification for classifier model 1 only

C = Correct classification for classifier model 2 only

r = Number of rows

c = Number of columns

DF = Degrees of freedom

Using the same confusion matrix, the accuracy of both classifier models is represented through Equations (5) and (6).

$$Model\ 1\ Accura = \left(\frac{A+B}{A+B+C+D}\right) x\ 100 \quad (5)$$

$$Model\ 2\ Accuracy = \left(\frac{A+C}{A+B+C+D}\right) x\ 100 \quad (6)$$

A Paired t-test is used to investigate the mean difference between two variables of the same subject. In a paired t-test, groups are associated by being in the same item, group of people, or under the same conditions. In this study, a paired t-test was used to determine which method of classification performs better than the other in terms of speed. Speed refers to the amount of time it takes for both classifying methods to successfully classify a sample. The paired t-test formula is described by Equation (7).

$$t = \frac{\sum d}{\sqrt{\frac{n(\sum d^2)-(\sum d^2)}{n-1}}} \quad (7)$$

Where:

d = Difference per paired values

n = Number of samples

Σd = Sum of the differences

2.9. Cost Computation

The total cost computation of the components used for the construction of the prototype along with the miscellaneous fee is shown in Table 3. The miscellaneous fee includes the shipping fees, transportation fees, and the total cost of eggplant samples.

Table 3. Total cost computation

QTY.	DESCRIPTION	UNIT COST (in PHP)	TOTAL COST (in PHP)
1	Arduino Nano	1,062.00	1,062.00
1	NIR Spectral Sensor IC	1,330.00	1,330.00
1	OLED Display	200.00	200.00
1	SD Card	160.00	160.00
1	MicroSD Card Module	35.00	35.00
1	9V Rechargeable Battery	499.00	499.00
1	Coded Rotary Switch	141.00	141.00
1	Rocker Switch	36.00	36.00
1	3D Printed Chassis	2,000.00	2,000.00
	Miscellaneous	3,000.00	3,000.00
Total		**8,463.00**	**8,463.00**

3. Results and Discussion

3.1. Accuracy Testing

The summary of the tabulated classification results during the evaluation is shown in Table 4. A total of 50 eggplant samples randomly picked from the farm's produce to be delivered to respective markets were evaluated. The prediction of the farmer and the prototype were compared to the actual classification of the sample. The farmer used the traditional visual method of classification in Fig. 26 while the prototype used the trained SVM classifier algorithm as seen in Fig. 27. Specifically, the prototype scans a single eggplant sample into five different regions. If one of the scanned regions was classified as infested, then that particular eggplant sample will be labeled as infested.

Figure 26. Farmer Classification During Evaluation Conducted at Imus Farm

Figure 27. Prototype Classification During Evaluation Conducted at Imus Farm

Table 4. Summary of classification results during evaluation

	No. Of Classified Infested Eggplants	**No. Of Classified Uninfested Eggplants**	**Total No. Of Classified Eggplants**
Farmer	6	44	50
Prototype	14	36	50
Actual	14	36	50

Using the classification results, a 2x2 contingency table for McNemar's test which contains the record of correct and incorrect reading between the two methods of classification was shown in Table 5. Using McNemar's formula (5) and (6) for classifier model accuracy presented in the methodology section, the calculation resulted in an 84% validated accuracy for the prototype whereas the accuracy of the farmer was found to be sixty-eight percent 68%. Thus, there is a 16% percent recorded difference between the accuracy of the prototype and the farmer.

Table 5. Confusion matrix of correct and incorrect reading of classifier models

		FARMER	
		Correct	Incorrect
PROTOTYPE	Correct	A = 30	B = 12
	Incorrect	C = 4	D = 4

To scientifically determine if there was a significant difference between the accuracy of the two classifier models, McNemar's test was used. The Null Hypothesis states that there was no significant difference between the two classifier models in terms of accuracy while the Alternative Hypothesis states that there was a significant difference between the two classifier models in terms of accuracy. The analysis used a 0.05 level of significance which is equivalent to a 95% confidence interval. The result of the test is shown in Table 6.

Table 6. McNemar's test results for accuracy

Chi-Squared (X^2)	Degrees Of Freedom (Df)	P-Value
4	1	0.0455

Based on the results shown in Table 6, the p-value was found to be 0.0455 – a value less than the set level of significance which is 0.05. Thus, the null hypothesis was rejected and the alternative hypothesis was accepted. This indicates that there is a significant difference between the performance of both classifier models in terms of accuracy, with the prototype outperforming the farmer by 16%.

3.2. Precision Testing

Using the data from Table 4, a confusion matrix that contains the predicted classification of the farmer and the actual classification of the samples is shown in Table 7. On the other hand, a confusion matrix that contains the predicted classification of the prototype and the actual classification of the samples is shown in Table 8.

Table 7. Confusion matrix of the actual and predicted (farmer) classifications

		Predicted (Farmer)	
		Uninfested	Infested
Actual	Uninfested	Tn = 32	FP = 4
	Infested	FN = 12	TP = 2

Table 8. Confusion matrix of the actual and predicted (prototype) classifications

		Predicted (prototype)	
		Uninfested	Infested
ACTUAL	Uninfested	TN = 32	FP = 4
	Infested	FN = 4	TP = 10

By utilizing the data from Tables 7-8 and the formula (1) for precision mentioned in the methodology section, the precision of both classifiers was computed and was shown in Table 9. The calculation resulted in a precision of about 33.33% for the farmer's predictions whereas the precision of the prototype's predictions was found to be approximately 72.83%. Therefore, it was deduced that the prototype's precision returns more relevant results, outperforming the farmer's precision by about 39.5%.

Table 9. Computed classification precision in percentage

Classification method	Precision in percentage (%)
Farmer's Classification (Traditional)	33.23%
Prototype's Classification	72.83%

3.2. Precision Testing

The summary of the tabulated speed results during the evaluation is shown in Table 10. The farmer and prototype's speed was recorded using a stopwatch application.

Table 10. Summary of speed results during evaluation

Number of samples	Average speed of classifications (s)	
	Farmer	Prototype
50	10.7356	9.736

To scientifically determine if there was a significant difference between the speed of the two classifier models, the Paired T-test test was used. The Null Hypothesis states that the farmer performs better than the prototype in terms of speed while the Alternative Hypothesis states that the prototype performs better than the farmer in terms of speed. The analysis used a 0.05 level of significance which is equivalent to a 95%

confidence interval. The calculations were done using Microsoft Excel, a powerful spreadsheet tool used for data visualization and analysis. The results of the analysis are shown in Table 11.

Table 11. Paired T-test table calculated in MS Excel

Parameters	Farmer's Speed in Classifying	Prototype's Speed in Classifying
Mean	10.7356	9.736
Variance	1.689914939	3.011269388
Observations	50	50
Pearson	0.169078299	
Correlation		
Hypothesized		
Mean	0	
Difference	49	
df	3.56166934	
t Stat	0.000415905	
P(T<=t) one-tail		
t Critical one-tail	1.676550893	

The calculation resulted in a mean or average classification time of 9.736 seconds for the prototype, whereas, the average classification time of the farmer is approximately 10.736 secs. A mean difference of approximately 1 second was found between the prototype and the farmer's duration of classification. The p-value for the one-tailed test was found to be 0.000416 – a value that is less than the set significance level of 0.05. Thus, the null hypothesis was rejected and the alternative hypothesis was accepted. This indicates that the prototype performs better than the farmer in terms of speed.

4. Conclusions, Recommendations, and Limitations

4.1. Conclusions

Based on the objectives of the study and the conducted test results, it was concluded that:

- The NIR-based prototype was capable of classifying EFSB infestation within an eggplant fruit on-site through the use of the trained Support Vector Machine classifier algorithm
- The prototype outperformed the traditional method of classification in terms of accuracy, precision, and speed
- The prototype's resulting accuracy was 84%, which is 16% more accurate when compared to the traditional method.
- The resulting precision of the prototype was 72.83%, which is 39.5% more precise compared to the traditional method.
- The average speed of the prototype is 9.736, outperforming the traditional method by approximately by a second in classifying EFSB.

4.2. Recommendations

Based on the conclusions mentioned, this study recommends the following:
Numbers/List should be as follows:

1. Include a battery level indicator for charging;
2. Explore the use of microprocessors or other microcontrollers that can provide enough storage space to cater more data for the classifier model;
3. Explore the use of other Near-Infrared spectral sensors that cover a wider range of wavelengths in the Near-Infrared spectrum;
4. Automate the scanning process to improve the prototype's speed when classifying;
5. Test applicability of the prototype's classifier model in classifying eggplant with chilling injury.

4.3. Limitations

The study mainly focused on developing a portable NIRS device for non-invasive detection of EFSB infestation in an eggplant fruit during its post-harvest stage. The device only used supervised machine learning for classifying between infested and non-infested eggplant fruits. Furthermore, the study's area of interest is dedicated to Eggplant *(Solanum melongena L.)* specifically the Fortuner F1 variety only. Thus, other common vegetable crops in the Philippines such as tomato, potato, and cassava were not included.

In addition, the study's area of interest was concerned with the detection of the larvae inside the fruit of the eggplant. It did not include the detection of EFSB in its egg, cocoon, and/or moth stage. The study did not include the detection of EFSB in other plant parts.

The study also did not partake in the detection of other eggplant pests such as Ants, Aphids *(Aphis gossypii)*, Colorado Potato Beetle *(Leptinotarsa decemlineata)*, Cutworm, Spider Mite, Stink bud, Thrips, Whiteflies, and Caterpillar. In addition, diseases related to the crop such as Bacterial Wilt, Damping-off, Verticillium Wilt, Phytophthora Blight, and Phomopsis are not included as well.

The study only focused on the detection of EFSB infestation. Thus, the eradication of EFSB as well as pest disease and management are beyond the scope of this study.

Acknowledgments

The authors would like to extend their earnest thanks and heartfelt gratitude to the Department of Computer, Electronics, and Electrical Engineering of Cavite State University – Don Severino Delas Alas Campus and to all of the individuals who have contributed and provided them continuous support in the fulfillment of this study.

Conflict of interest

The authors of this manuscript declare that there are no conflicts of interest in the publication of this manuscript.

Author Contribution Statement

Authors Maria Patrice Lajom, John Paul Remigio, and Edwin Arboleda proposed the research problem

Authors Maria Patrice Lajom and John Paul Remigio developed the theory and methodology for the proposed problem.

Author Edwin Arboleda verified the analytical methods and supervised the findings of this work.

Author Rhen John Rey Sacala organized and collated the findings of this work.

All authors discussed the results and contributed to the final manuscript.

References

[1] P. Tsouvaltzis, F. Babellahi, M. L. Amodioand G. Colelli, "Early detection of eggplant fruit stored at chilling temperature using different non-destructive optical techniques and supervised classification algorithms", vol. 159, p. 111001, Jun. 2020, https://doi.org/10.1016/j.postharvbio.2019.111001

[2] M. A. Chapman, *Eggplant Breeding and Improvement for Future Climates*. Springer International Publishing, 2020. https://doi.org/10.1007/978-3-319-97415-6_4

[3] R. Babu, K. Chandraand M. S. Omprakash, *Genetic Variability, Heritability, and Genetic Advance in Eggplant (Solanum melongena L.) Genotypes under Normal and Osmotic Stress in Invitro Condition*, vol. vol. 6. International Journal of Current Microbiology and Applied Sciences, 2017.https://doi.org/10.20546/ijcmas.2017.603.087

[4] N. Gürbüz, S. Uluişik, A. Frary, A. Fraryand S. Doğanlar, *Health benefits and bioactive compounds of eggplant*, vol. 268. 2018. doi: https://doi.org/10.1016/j.foodchem.2018.06.093

[5] J. G. Maggay, *Mobile-Based Eggplant Diseases Recognition System using Image Processing Techniques*, vol. 9. 2020. https://doi.org/10.30534/ijatcse/2020/3391.12020.

[6] Y. A. Akter and M. O. Rahman, "Development of a computer vision based Eggplant grading system", in *2017 4th International Conference on Advances in Electrical Engineering (ICAEE)*, Dhaka, Bangladesh: IEEE, Jun. 2017. https://doi.org/10.1109/ICAEE.2017.8255368.

[7] D. M. Hautea, "Field Performance of Bt Eggplants (Solanum melongena L.) in the Philippines: Cry1Ac Expression and Control of the Eggplant Fruit and Shoot Borer (Leucinodes orbonalis Guenée)", vol. 11, no. 6, p. e0157498, Jun. 2016, https://doi.org/10.1371/journal.pone.0157498.

[8] H. Javed, T. Mukhtar, K. Javedand Ata-Ul-Mohsin, "management of eggplant shoot and fruit borer (leucinodes orbonalis guenee) by integrating different non-chemical approaches", vol. 54, no. 1, pp. 65–70, Jun. 2017, https://doi.org/10.21162/pakjas/17.5282

[9] M. S. Parvin, M. M. Rahmanand M. S. Hasan, "Refinement of different combinations of (Brinjal Shoot and Fruit Borer) in Brinjal", vol. 5, no. 3, pp. 196–208, Jun. 2022, https://doi.org/10.51386/25815946/ijsms-v5i3p122

[10]J. C. Chang, D. W. Ponnathand S. Ramasamy, "Phylogeographical structure in mitochondrial DNA of eggplant fruit and shoot borer, Leucinodes orbonalis Guenée (Lepidoptera: Crambidae) in South and Southeast Asia", vol. 27, no. 1, pp. 198–204, Jun. 2016, https://doi.org/10.3109/19401736.2014.880890 .

[11]M. Gautam, S. Kafle, B. Regm, G. Thapaand S. Paudel, "Management of Brinjal Fruit and Shoot Borer (Leucinodes orbonalis Guenee) in Nepal", vol. 3, no. 9, pp. 188–195, Jun. 2019. https://doi.org/10.31080/ASAG.2019.03.0632

[12]E. Owen, E. A. Akpanand I. A. Udo, "Comparative efficacy of some botanical insecticides in the control of eggplant fruit and shoot borer (leucinodes orbonalis guenee)", vol. 22, no. 1, pp. 191–196, Jun. 2023. https://doi.org/10.4314/gjass.v22i1.12

[13]M. M. I. Mollah, N. Hassan, S. Khatunand M. H. Rahman, "Sequential application of bio-pesticides suppresses eggplant shoot and fruit borer, Leucinodes orbonalis Guenee infestation", vol. 10, no. 5, pp. 140–146, Jun. 2022.https://doi.org/10.22271/j.ento.2022.v10.i5b.9054

[14]M. N. Ahmad, A. R. M. Shariffand R. Moslim, "Monitoring insect pest infestation via different spectroscopic techniques", vol. 53, no. 10, pp. 836–853, Jun. 2018.https://doi.org/10.1080/05704928.2018.1445094

[15]B. Shurygin, "Mutual augmentation of spectral sensing and machine learning for Non-Invasive detection of apple fruit damages", vol. 8, no. 12, p. 1111, Jan. 2022.https://doi.org/10.3390/horticulturae8121111

[16]B. M. Shaker and R. M. Fenjan, "Charactrazation of bio-char produced from sesbania stems (sesbania grandiflora)", vol. 27, no. 2, pp. 204–212, Jun. 2023.https://doi.org/10.31272/jeasd.27.2.5.

[17]J. B. Johnson and M. Naiker, "Seeing Red: A review of the use of near-infrared spectroscopy (NIRS) in entomology", vol. 55, no. 9–10, pp. 810–839, Jun. 2019. https://doi.org/10.1080/05704928.2019.1685532

[18]D. K. Ninh, T. Doan, C. K. Ninh, T. X. Nguyen-Thiand N. L. Thành, "Fruit recognition based on near-infrared spectroscopy using deep neural networks", Jun. 2021.https://doi.org/10.1145/3453800.3453817

[19]D. Cozzolino, "The Ability of Near Infrared (NIR) Spectroscopy to Predict Functional Properties in Foods: Challenges and Opportunities", vol. 26, no. 22, p. 6981, Jun. 2021. https://doi.org/10.3390/molecules26226981

[20]S. Juneja, R. G. Moulick, D. Kushwaha, H. A. Gandhiand J. Bhattacharya, "Near-infrared spectroscopy: An important noninvasive and sensitive tool for point-of-care biosensing application", pp. 161–184, Jun. 2022.https://doi.org/10.1016/B978-0-323-85650-8.00004-8

[21]A. A. Adedeji, "Non-Destructive Technologies for Detecting Insect Infestation in Fruits and Vegetables under Postharvest Conditions: A Critical Review", vol. 9, no. 7, p. 927, Jun. 2020.https://doi.org/10.3390/foods9070927

[22]B. Jamshidi, "Ability of near-infrared spectroscopy for non-destructive detection of internal insect infestation in fruits: Meta-analysis of spectral ranges and optical measurement modes", vol. 225, p. 117479, Jun. 2020.https://doi.org/10.1016/j.saa.2019.117479

[23]E. R. Arboleda, "Discrimination of civet coffee using near infrared spectroscopy and artificial neural network", vol. 8, no. 39, pp. 324–334, Jun. 2018.https://doi.org/10.19101/IJACR.2018.839007

[24]Y. Liu, Y. Liuand Y. Song, *Research on near infrared spectroscopy based on DMD*. Nantong, China: SPIE, 2023. https://doi.org/10.1117/12.2664753

[25]A. H. Rashed and M. H. Hamd, "Robust Detection And Recognition System Based On Facial Extraction And Decision Tree", vol. 25, no. 4, pp. 40–50, Jun. 2022. https://doi.org/10.31272/jeasd.25.4.4

[26]X. Wang, "Near-infrared spectroscopy in biomedical applications", in *International Conference on Biomedical and Intelligent Systems (IC-BIS 2022)*, Chengdu, China: SPIE, Jun. 2022. https://doi.org/10.1117/12.2660335

[27]S.-I. Sohn, “An Overview of Near Infrared Spectroscopy and Its Applications in the Detection of Genetically Modified Organisms”, vol. 22, no. 18, p. 9940, Jun. 2021. https://doi.org/10.3390/ijms22189940

[28]G. Gezici and B. Yanıkoğlu, *Sentiment Analysis in Turkish*. Springer International Publishing, 2018. https://doi.org/10.1007/978-3-319-90165-7_12

[29]S. Gurung, *Brief study on machine learning*. Nepal College of Information Technology, 2020. https://doi.org/10.13140/RG.2.2.10196.12169

[30]D. Chopra and R. Khurana, *Linear Regression and Logistic Regression in Introduction to Machine Learning with Python*. Bentham Science Publishers, 2023.https://doi.org/10.2174/97898151244221230101

[31]Gupta, V., Mishra, V. K., Singhal, P., & Kumar, A. (2022). An Overview of Supervised Machine Learning Algorithm in 2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART), Moradabad, India: IEEE. pp. 87–92. https://doi.org/10.1109/SMART55829.2022.10047618

[32]Çelik, Ö. (2018). A Research on Machine Learning Methods and Its Applications. Journal of Educational Technology and Online Learning, 1(3), 25–40. https://doi.org/10.31681/jetol.457046

[33]Biswas, A., Saran, I., & Wilson, F. P. (2021). Introduction to Supervised Machine Learning. Kidney360, 2(5), 878–880. https://doi.org/10.34067/KID.0000182021

[34]Li, M., Ekramirad, N., Rady, A., & Adedeji, A. (2018). Application of Acoustic Emission and Machine Learning to Detect Codling Moth Infested Apples. Transactions of the ASABE, vol. 61, no. 3, pp. 1157–1164. https://doi.org/10.13031/trans.12548

[35]J. A. Nasuli, J. P. Lumbisand E. R. Arboleda, *Arabica coffee bean Quality identification using support Vector Machine-Based Digital Image Processing*. 2023. https://doi.org/10.13140/RG.2.2.19015.47522

[36]J. T. Alander, V. Bochko, B. Martinkauppi, S. Saranwongand T. Mantere, “A Review of Optical Nondestructive Visual and Near-Infrared Methods for Food Quality and Safety”, vol. 2013, pp. 1–36, Jun. 2013. https://doi.org/10.1155/2013/341402

[37]S. Sundaram and A. Zeid, “Artificial Intelligence-Based Smart Quality Inspection for Manufacturing”, vol. 14, no. 3, p. 570, Jun. 2023. https://doi.org/10.3390/mi14030570

[38]T. Welsh, F. Alrimawi, A. Farahani, D. Hassett, A. Zismanand B. Nuseibeh, “Topology-Aware Adaptive Inspection for Fraud in I4.0 Supply Chains”, vol. 19, no. 4, pp. 5656–5666, Jun. 2023. https://doi.org/10.1109/TII.2022.3205369

[39]M. I. Kaysar, M. S. Mia, M. S. Islamand A. K. M. G. Kausar, “Postharvest loss assessment of brinjal in some selected areas of Bangladesh”, vol. 2, no. 2, pp. 118–124, Jun. 2016. https://doi.org/10.18801/ijbmsr.020216.13

[40]D. Li, Z. Song, C. Quan, X. Xuand C. Liu, “Recent advances in image fusion technology in agriculture”, vol. 191, p. 106491, Jun. 2021. https://doi.org/10.1016/j.compag.2021.106491