Roselle Pest Detection and Classification Using Threshold and Template Matching

Ade Bastian [1,*], Adie Iman Nurzaman [1], Tri Ferga Prasetyo [1], and Sri Fatimah [2]

[1] Department of Informatics, Faculty of Engineering Universitas Majalengka, Indonesia;
Email: adieimannurzaman@gmail.com (A.I.N.), triferga.prasetyo@gmail.com (T.F.P.)
[2] Department of Socio-Economic, Faculty of Agriculture Universits Padjadjaran, Indonesia;
Email: srifatimah.sf@gmail.com (S.F.)
*Correspondence: adebastian@unma.ac.id (A.B.)

***Abstract*—Roselle is a fiber-producing plant that has broad benefits for health food, so many farmers are interested in starting to cultivate it. This study aims to design a rosella plant pest detection system to reduce the risk of crop failure or reduced yields of rosella calyx. The design of a system for detecting and classifying rosella pests uses the threshold method as a digital image processing method connected via the internet with information media applications and template matching to detect and classify pests on rosella plants. Detection of pests on rosella plants has been successfully built using a detection system using thresholding and template matching methods. Datasets of rosella plant pests that are not yet widely available encourage the detection of rosella plant pests with datasets from rosella plant objects and limited data testing. Testing with 75% accuracy, the detection process is affected by light and camera quality.**

***Keywords*—classification, threshold method, image processing, template matching, embedded system**

I. Introduction

Roselle plants as the main agricultural commodity in Asia such as Thailand and China are plants originating from India and West Africa [1]. Roselle is known by the Indonesian people since 1922 as an ornamental plant, hedge plant and fiber-producing plant. Roselle is currently a plant that is in demand by the public because of the various products that can be produced from flowers and their fibers so that they experience a fairly high increase in cultivation. There are two types of plant varieties from the *Malva-ceae family* that are commonly cultivated, namely roselle with yellow petals (*Hibiscus sabdariffa var. Altisima*) and roselle with red petals (*H. sabda-riffa var. Sabdariffa*) [2]. Roselle with yellow petals is usually used as stem fiber as a material for making ropes and burlap sacks. Another variety, namely roselle with red flower petals, also began to be cultivated. Chemical compounds contained in the flower petals include: gossypetin, anthocyanin, and glucoside hibiscin which are useful for treating cancer and inflammation, lowering blood pressure, improving blood circulation, lowering blood viscosity. In addition, it can prevent the formation of kidney stones and smooth bowel movements or as a urine laxative. Along with increasing knowledge about the benefits of roselle and the increasing economic value of roselle, many farmers are increasingly interested in starting to cultivate it. In the cultivation of roselle, many factors must be prepared including land preparation, preparation of plant material, nurseries, planting in polybags, spacing, fertilization, pruning, pest control, harvesting and post-harvest [3].

Roselle cultivation must be done properly in order to produce quality roselle plants and petals when harvesting. Often farmers experience crop failure or the yield of roselle petals decreases, this is mostly caused by pests that attack roselle plants so that not a few farmers suffer losses due to pests that cannot be controlled previously. Some pests that are susceptible to attack Roselle include aphids (*Aphididae family*), whitefly (*Bemisia spp*.), ground lice (*Oxy carenus spp.)* and hibiscus mealybug (*Maconellicoccus hirsutus*) [4]. In the Bajo Balsas region of Mexico, the Roselle plant faces the challenge of annual dry climate hazards [5]. In Taiwan, Roselle plant stems suffer from wilt disease. Withered stems are brown in color, stem volume is reduced and covered with white aerial hyphae [1]. In Egypt, Roselle plants suffer from root rot disease [6]. In handling pest attacks, errors often occur such as errors in detecting the type of pest which of course has an impact on errors in the prevention process or the process of eradicating the pests found. Farmers often experience doubts in carrying out the prevention process because of the lack of reliable information in terms of overcoming and eradicating the pests that are found.

To overcome errors in the process of handling pests found, it is necessary to have a tool that has been given a touch of the latest technology, one of which is by utilizing image processing technology or digital image processing and computer vision technology which will be closely related to artificial intelligence [7]. Image segmentation provides further understanding of an image. Image pixels that are close together and do not overlap [8]. The image is divided into several parts to change the representation of the image so that it is easier to examine [9]. Image

Manuscript received December 2, 2022; revised April 13, 2023; accepted May 22, 2023.

doi: 10.18178/joig.11.4.330-342

segmentation is the process of partitioning a digital image into several segments (sets of pixels as super pixels) with the aim of extracting images [10]. Image segmentation serves to distinguish the front image from the background of an image [11]. A simple tool that effectively separates objects from the background [12]. The use of image segmentation in various fields including Computer vision, Video Coding, Military Research Field, Medical Image Analysis, Traffic Image Analysis and Remote Sensing Meteorological Services. Thresholding method that uses a gray scale between the target to be extracted and the image background is the most widely used method [13]. Identification of plant pests can be applied by applying k-means clustering and correspondence filters that can maintain various shapes, sizes, positions and orientations using computationally efficient algorithms [14]. The plant leaf automatic classification system can use shape-based features and neural network classifiers with two approaches, namely one technique based on the moment invariant model and the radius centroid model, both of which are compared with the classification method [15]. Leaf health classification can use a neural network with progressive resizing which classifies healthy leaves, caterpillar effects, wilted leaves, yellow leaves, leaf spots (the most common disease) [16]. The method used in the digital image processing of roselle plant pests this time is the threshold or thresholding method which has aspects in the process of recognizing the shape of the pest and the shape of the damaged plant because it has been attacked by the pest.

Agricultural activities have become more intensive by using IoT technology [17]. To make it easier for farmers in the process of monitoring or monitoring pest attack activities, in addition to using the threshold method as a digital image processing method, this system will be created by applying Internet of Things (IoT) technology that is able to connect via the internet with information media applications, both website-based and online applications. android based application. Agricultural systems are assisted by IoT devices to run smart farming systems that can save time and increase farmer productivity [18].

The leaf dataset available on the website page above, has not found a dataset for the leaves of the Roselle plant. Likewise, the results of tracing several studies, not many have designed a Roselle plant disease detection tool using IoT technology. The main contributions of this research are:

1. Designing a pest detection and classification tool on Roselle plants by applying the Thresholding method to reduce the risk of errors in the handling or prevention process of detected pests.
2. Applying real time notification technology to a system that is integrated with pest detection and classification tools on Roselle plants, so that information about pest detection will be received by farmers more quickly by utilizing IoT technology.
3. Designing a system that is able to integrate (embedded system) pest detection and classification systems on Roselle plants with other systems that support Roselle plant maintenance, such as Roselle petal maturity detection systems and soil fertility systems that are able to provide effectiveness in the Roselle plant maintenance process.

Problems that arise in the field include pests that attack cannot be directly addressed because inspections of pest types must be carried out, pest classification errors are found, and there is no tool that can detect and classify Roselle pests. Identification of needs is obtained by functional analysis, user analysis, hardware and software requirements and process analysis using the thresholding method. Then the hardware and software were designed as a blueprint for the Roselle pest detection tool. This research presents the process of detecting Roselle plant pests by utilizing independently collected datasets because there are not many complete datasets available. The detection uses the Thresholding method which is then designed to be a Roselle plant pest detection tool using IoT. Part of the Roselle plant maintenance embedded system, namely the Roselle petal maturity detection system and soil fertility system.

II. RESEARCH METHOD

The research stages of the detection and classification of roselle plant pests are presented in Fig. 1.

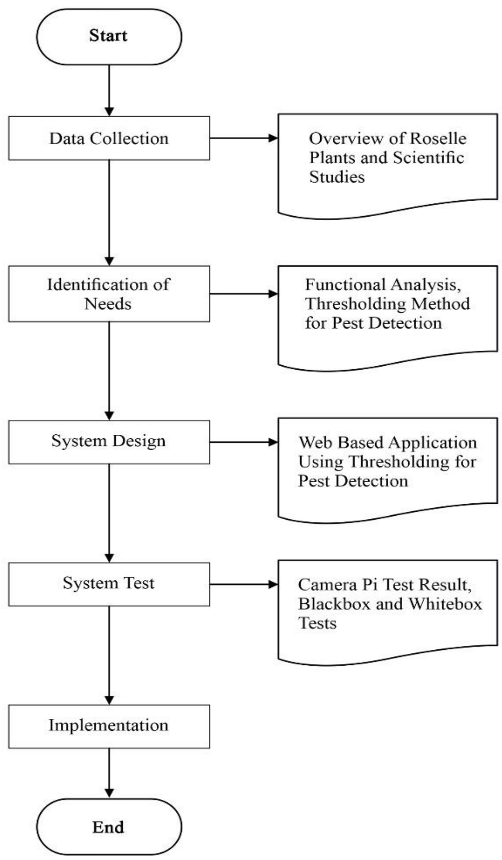


Figure 1. Research stages.

The first step is a field study to identify problems in the cultivation of Roselle plants in Cikondang Village, Cikijing District, Majalengka Regency, West Java

Province, Indonesia. Direct observation of roselle cultivation includes sowing seeds for four to five weeks, watering once every three to four days, neutralizing soil pH values, re-fertilizing, and harvesting roselle. One of the authors (Sri Fatimah) from Padjadjaran University is the initiator to the introduction of roselle flower planting in Cikondang village at the time this research, as part her community service program to develop Cikondang Village as rural tourism destination [19, 20]. As agriculture expert, she provides other authors an understanding of roselle plant maintenance, namely cultivation systems, nurseries, maintenance (watering, fertilization and pest control), harvesting and post-harvest handling.

In several leaf dataset sources that are available and accessible to the public, not many datasets have been found for the leaves of the Roselle plant. Table I is a number of web pages that contain leaf datasets:

TABLE I. LIST OF LEAF DATASET

Dataset Name	URL
Plant Image Analysis	https://plant-image-analysis.org/dataset
Swedish Leaf Dataset	https://www.cvl.isy.liu.se
Image Database of Plant Disease Symptoms (PPBD)	https://www.digipathos-rep.anptia.embrop.br/
UCI Machine Learning Repository	https://www.aarchies.ics.uci.edu/ml/datasets/plants
Plant Village	https://www.plantvillage.org/
Plant Phenotyping dataset	https://plant-phenotyping.org/dataset

The leaf dataset available on the website page above, has not found a dataset for the leaves of the Roselle plant. Likewise, the results of tracing several studies, not many have designed a Roselle plant disease detection tool using IoT technology. The main contributions of this research are:

1. Designing a pest detection and classification tool on Roselle plants by applying the Thresholding method to reduce the risk of errors in the handling or prevention process of detected pests.
2. Applying real-time notification technology to a system that is integrated with pest detection and classification tools on Roselle plants, so that information about pest detection will be received by farmers more quickly by utilizing IoT technology.
3. Designing a system that is able to integrate (embedded system) pest detection and classification systems on Roselle plants with other systems that support Roselle plant maintenance, such as Roselle petal maturity detection systems and soil fertility systems that are able to provide effectiveness in the Roselle plant maintenance process.

A. Thresholding Process

The use of image processing technology is expected to facilitate the process of analyzing the types of pests on roselle plants. Threshold technique is divided into two classes of methods, namely threshold based on the histogram of the entire image and histogram of edge pixels. Threshold assumes the image is presented on a number of value components that are almost homogeneous [21]. Thresholding is used to generate binary from an image [22]. In the threshold method there are several analyzes and stages to be carried out, namely the analysis of the scaling process, analysis of the grayscaling process, analysis of the gray level in the image to determine the thresholding threshold, analysis of the find contours and draw contours processes. Thresholding can be applied as long as there is a strong contrast between the image and the image background. The gray scale of the background or object image is a single entity with interconnected boundaries [23]. The image from the camera detection is processed to detect whether or not there are pests on the plant. Analysis of the stages of digital image processing using the thresholding method can be described by Fig. 2.

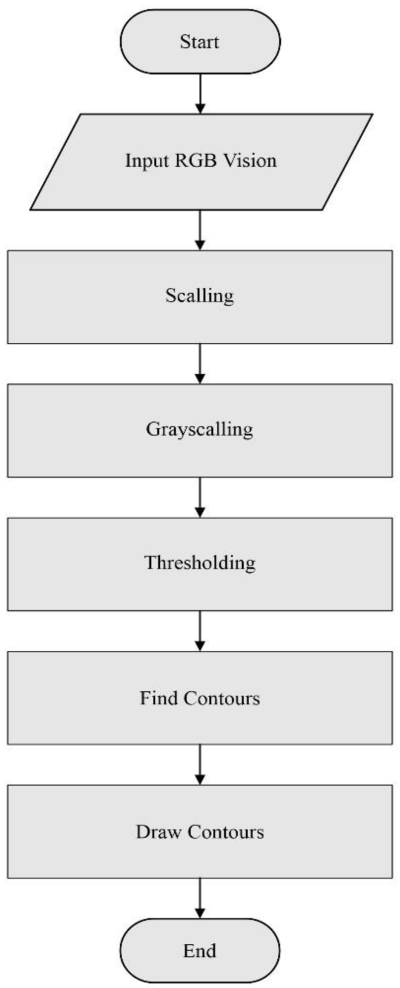


Figure 2. Flowchart thresholding method.

At this stage the implementation of the thresholding method is carried out, namely:

1. Determine the Red, Green, Blue (RGB) image that is the object of detection.
2. Scaling stage
   The digital image taken in real time by the webcam will be reduced by using the interpolation method. This method uses the average value of a region to represent the region. The pixel value in the coordinates of the interpolated image is obtained by calculating the average value of the 4 pixel values of the original image.
3. Grayscaling stage
   The representation of RGB values (Red, Green, Blue) is converted into an image consisting of white and black gradations which are commonly called grayscale. To convert RGB to greyscale, the following formula can be used:

$$grayscale = 0.299R + 0.587G + 0.114B \quad (1)$$

4. Thresholding stage
   Converts a greyscaled image to a binary image. Threshold value is calculated by dividing the value of the grayscale result in the previous stage by the value of the number of degrees of gray (0–256) divided by 256 (the value of the desired gray degree). The process of calculating the threshold value is as follows:

$$X = \frac{w}{b} \quad (2)$$

   where:
   X = threshold comparison value
   w = grayscaling result value
   b = 256/a (a = 256)
   To convert an RGB image into a binary image use the following rules:
   a. If the pixel value of the image = x then the pixel value becomes 1
   b. If the pixel value of the image = x then the pixel value becomes 0
5. The find contours stage is the process of finding the contours in the thresholding image, then taking the largest one.
6. The draw contours stage is the process of drawing an outline of the object, then drawing each extreme point, where the far left is red, the far right is blue, the top and bottom are teal.

B. Template Matching

Template Matching method is a method that is often used in the process of detecting objects or looking for object similarities, the basic technique of this method is the use of the thresholding method as the object search threshold value. The analysis of the template matching process is divided into three stages, the first stage is to process the RGB image input, the second stage is the grayscaling process and the third stage is thresholding. The template matching process flow can be seen in the image below:

In the template matching analysis process, it is carried out to detect and determine whether the detected object includes roselle plants or not. The stages in the template matching process are:

1. Determine the object of detection, in this study the roselle plant.
2. The representation of RGB values (Red, Green, Blue) is converted into an image consisting of white and black gradations which are commonly called grayscale as shown in Fig. 3.

Figure 3. Converting roselle RGB image to grayscale.

3. Next is the thresholding stage which is used to convert the greyscaled image into a binary image. Threshold value is calculated by dividing the value of the grayscale result in the previous stage by the value of the number of degrees of gray (0–256) divided by 256 (the value of the desired gray degree). Because the grayscaling value is known beforehand, to get the threshold value it is enough to divide the grayscale value and the value of the number of degrees of gray.

$$x \frac{w}{b}$$

4. x = Treshold Comparison Value Matching the detection object with the thresholding results with a predetermined template.

III. Result and Discussion

System design is the stage that is passed to facilitate the process of making the system. The system is made based on the design that has been described previously, namely the design of hardware and software. The hardware is made based on the design of a pest detection and classification tool on roselle plants with raspberry pi3 and the PiCamera module.

The software design is a design of the application that will be applied to the raspberry pi for the development of a pest detection and classification system on roselle plants and the RAOS webservice which will be applied as a liaison between the device and the RAOS robot. The embedded control program is a program for the raspberry pi module. Software design in python, this design contains software design that is able to detect roselle plant pests by taking pictures of roselle plants using thresholding and template matching methods. The language used to create the program is the Python programming language and is supported by the OpenCv library for processing images taken by the camera on a raspberry pi. This Python runs on raspberries as a replacement for a Personal Computer (PC) or laptop. The design of the system that runs on python software that will be installed on the Raspberry Pi, which is as follows:

1. Start by activating the system on the raspberry pi.
2. Then the program will check the raspberry pi connection with the internet. If it is connected, the system will be active, if not, the program will check the internet connection again.
3. After the system is active, the system will initialize the classes and modules that will be used.
4. The program in python will receive an action from the fire rest, namely checking the maturity of roselle or pests on roselle.
5. If Raspberry pi receives doneness check action
6. The picamera module will take pictures of roselle plants
7. The image is saved on the raspberry pi and made into an image.
8. The system will call the roselle checking class.
9. The grayscaling stage is the representation of the RGB values converted into an image consisting of white and black gradations which are commonly called grayscale.
10. Thresholding stage is the process used to convert grayscale images into binary images.
11. The next step is matching the thresholding image with training data or roselle flower template.
12. If roselle is detected, the next step is to call the maturity checking class.
13. The stage of determining the RGB image from the roselle image
14. The next process is the process of converting RGB images to HSV (Hue, Saturation, Value), with the aim of sharpening the color of the image taken on the rosella plant by utilizing the 3 color layers that exist in Hue, Saturation, and Value itself.
15. After that, the object detection process in the image is carried out based on the reference value (T) and the tolerance value (toll). With x as the HSV color in the existing pixels, the colors included in the range `T-tol < x < T+tol` will be marked in green.
16. If the roselle is ripe, it will send a message that it has detected a ripe roselle petal to the `save_detect` api rest, otherwise it will send a message that no ripe roselle petals have been detected to the `save_detect` api rest.
17. If Raspberry pi receives pest check action
18. The picamera module will take pictures of roselle plants
19. The image is saved on the raspberry pi and made into an image.
20. The system will call the roselle checking class.
21. The grayscaling stage is that the RGB (Red, Green, Blue) value representation is converted into an image consisting of white and black gradations which are commonly called grayscale.
22. Thresholding stage is the process used to convert grayscale images into binary images.
23. The next step is matching the teholding image with training data or roselle flower templates.
24. If roselle is detected, the system will call the pest checking class
25. Thresholding phase of pest screening is the process used to convert the grayscale image into a binary image.
26. The next step is matching the thresholding image with training data or roselle pest templates.
27. If a pest is detected, it will send information about a pest detected in roselle to the save detect API rest. Otherwise it will call the hole checking class.
28. The grayscaling stage is that the RGB (Red, Green, Blue) value representation is converted into an image consisting of white and black gradations which are commonly called grayscale.
29. Thresholding stage is the process used to convert the grayscale image into a binary image.
30. The edge detection stage is a process that is carried out to detect the edge line that limits two areas that have different brightness levels. At this stage is the detection of holes.
31. The detection blur stage is to give a blur effect to objects other than objects detected with edge detection.
32. If a hole is detected, it will send a description of the detected hole in the roselle to the `save_detect` API rest, if it is not detected, it will send a description of the undetected hole in the roselle to the `save_detect` fire rest.

The explanation of the System Flowchart in the Python Program above, is depicted in Fig. 4.

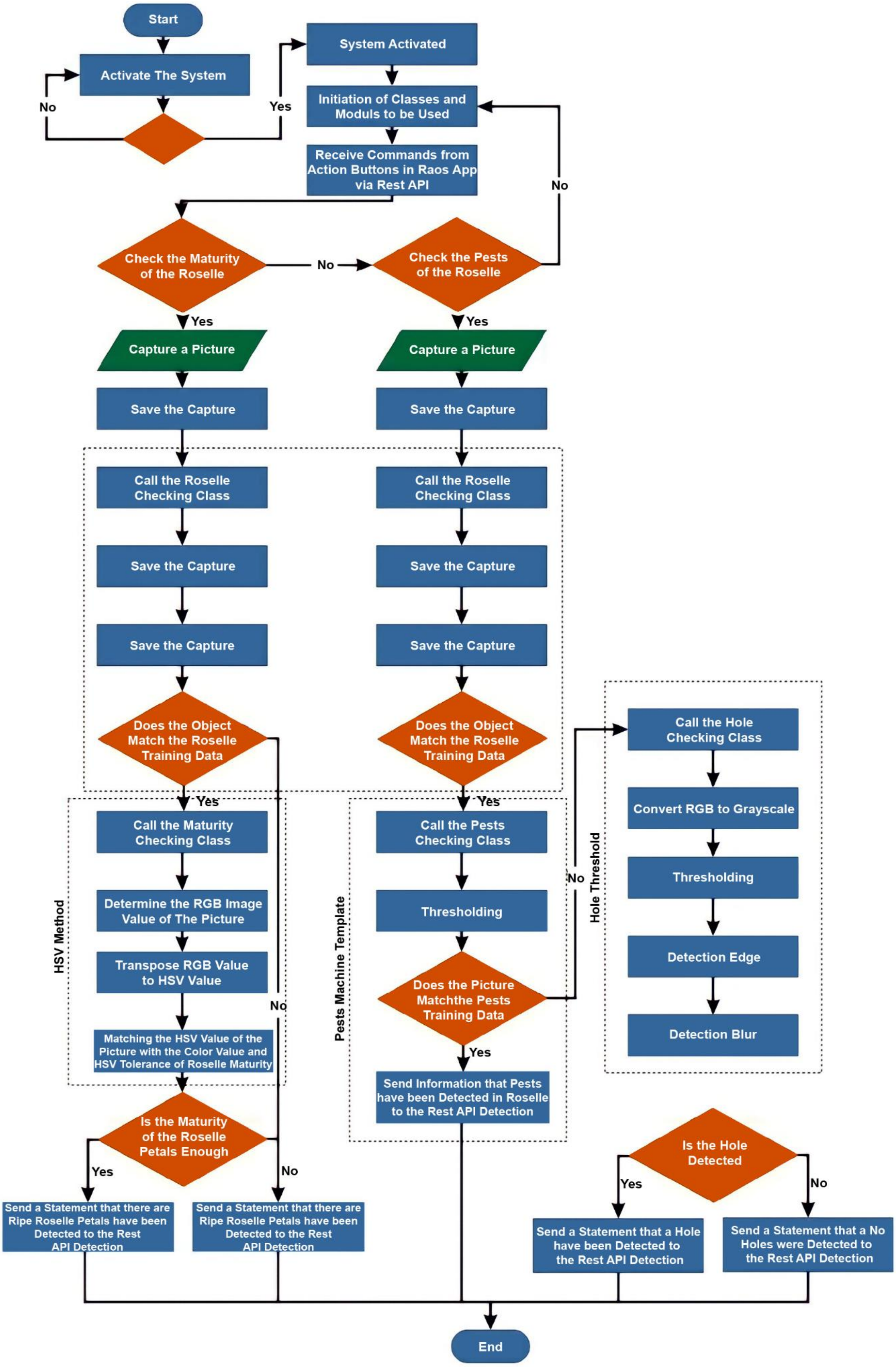


Figure 4. Flowchart of the system in the python program.

The CodeIgniter framework works with the MVC concept (Models, Views, Controllers) where the controller becomes the call center between the model and the view. By using the CodeIgniter framework, the process of making web-based applications is faster because the functions needed in the process of making websites and web services are provided. This system consists of 3 actors, namely Super Admin, Admin and Devices, consisting of 16 processes/use cases, namely user data processing, API Key data processing, notification data set processing, reference data processing, monitoring of nutrition check results, monitoring of pest check results, monitoring Petals check results, monitoring user activity logs, changing account identities, changing account passwords, forgetting passwords, rest API server, checking login status, validation, login and logout. Each process/usecase cannot be accessed directly unless the usecase forgets the password and rests the API server, to access the process/usecase it is required to successfully login first. The device has 1 process/usecase which is rest API server. Use case system diagram can be seen in Fig. 5.

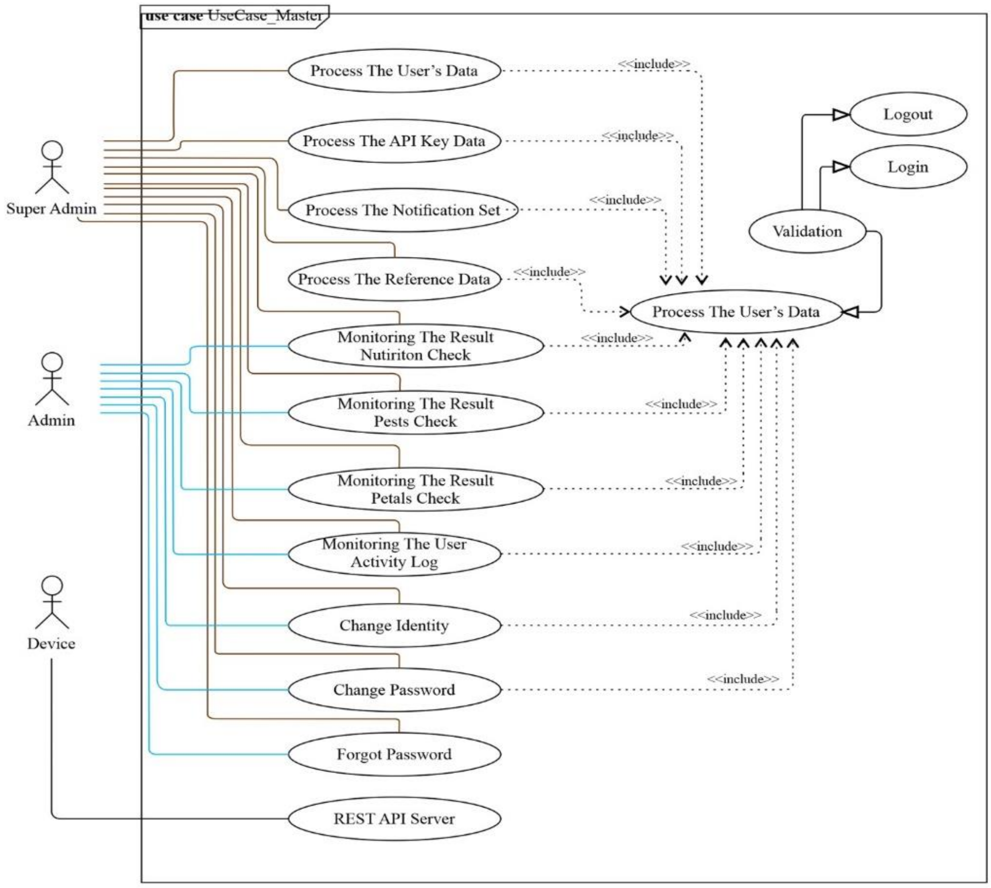


Figure 5. Use case diagram system.

The user, frontend system, and backend database interact to check plant pests in Fig. 6. Pest control involves: User chooses "monitoring" from the frontend system interface. The frontend system uses GET operations to deliver pest control search queries to the database. The frontend system queries the database, which does searches. The frontend system presents valid information to the user if the database finds the pest check. Frontend system information is accurate. Fig. 7 shows the plant pest monitoring technique. The admin launches the plant pest monitoring app on a browser. The admin picks "Monitoring" from the web browser application menu. The web browser requests monitoring from the web service. Web browser requests initiate the process on the web service. Pest data is requested from the database detection table by the web service. Web services request pest data from the table detection database. The online service receives pest data from the database. The web service transmits the database answer to the browser. The web service sends pest data to the browser. The web browser shows the admin pest statistics. The admin may access plant pest data after monitoring.

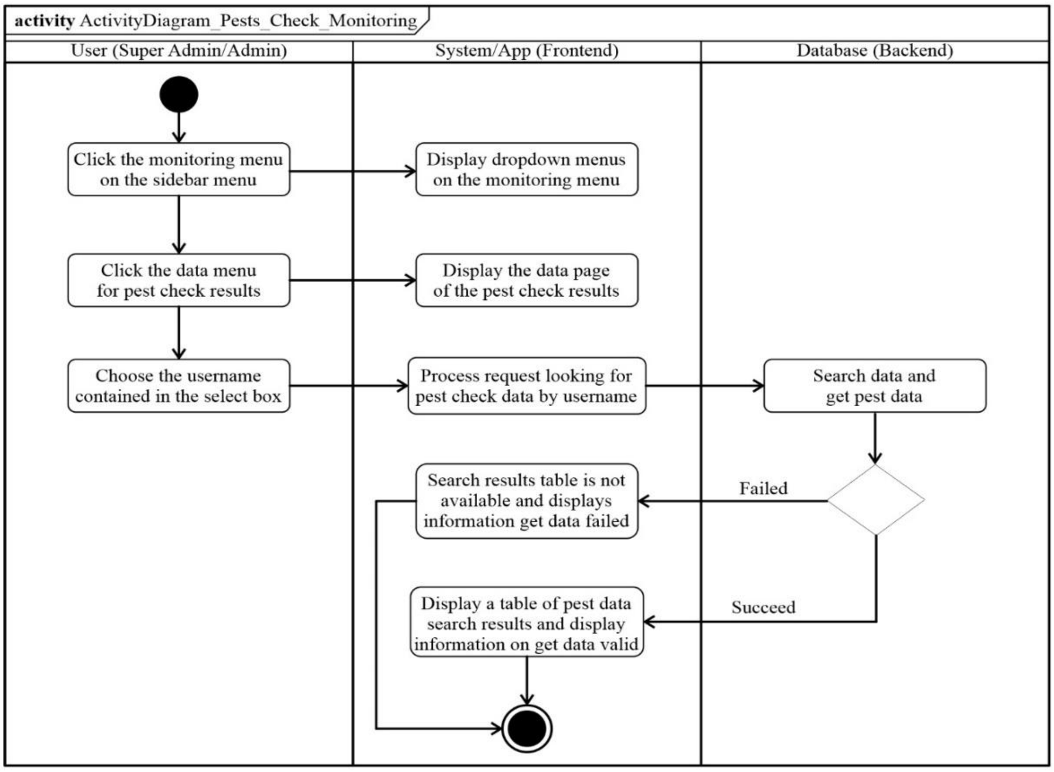


Figure 6. Activity diagram of monitoring pest check.

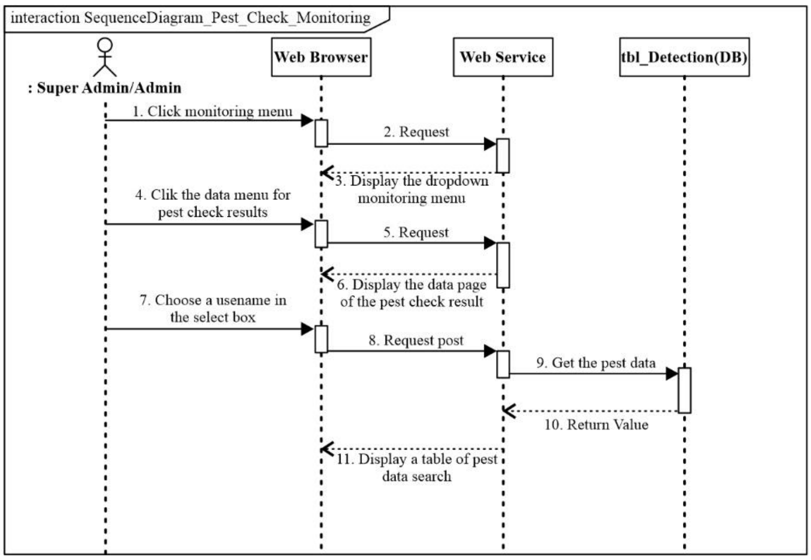


Figure 7. Sequence diagram of monitoring pest check.

The class diagram shows how database items relate. User, API, Land-set, Nutrition, Detection, Ref_device, Ref_subscriber, Log_act, Set_push, Ref_air, Ref_pupuk, Ref_kapur, Ref_button, Ref_wemos_1, and Ref_wemos_2. The primary object, the User table, includes properties like id_user, username, password, name, jk, address, email, nohp, level_user, and Photo. API, Land-Set, Nutrition, Detect, Ref_subscriber, and Log_act utilize User's primary key id_user. The main key links the entities to the User entity.

The Users table is connected to the API table through the id_user primary key. Each API has one User. The id_set_land primary key links the Nutrition table to the Land-Set database. Each land-Set corresponds to one Nutrient. Nutrition's id_nutrition field is linked to the Users table's id_user primary key. Each Nutrient has a User. The Land-Set table is connected to the Detect table by the id_set_land primary key. Each Detection has a Land-Set. The API table's main key links to the Ref_device table's id_ref_device property. Each Ref_device has one API.

This Fig. 8 shows database entity relationships and links, which aid system development and maintenance:

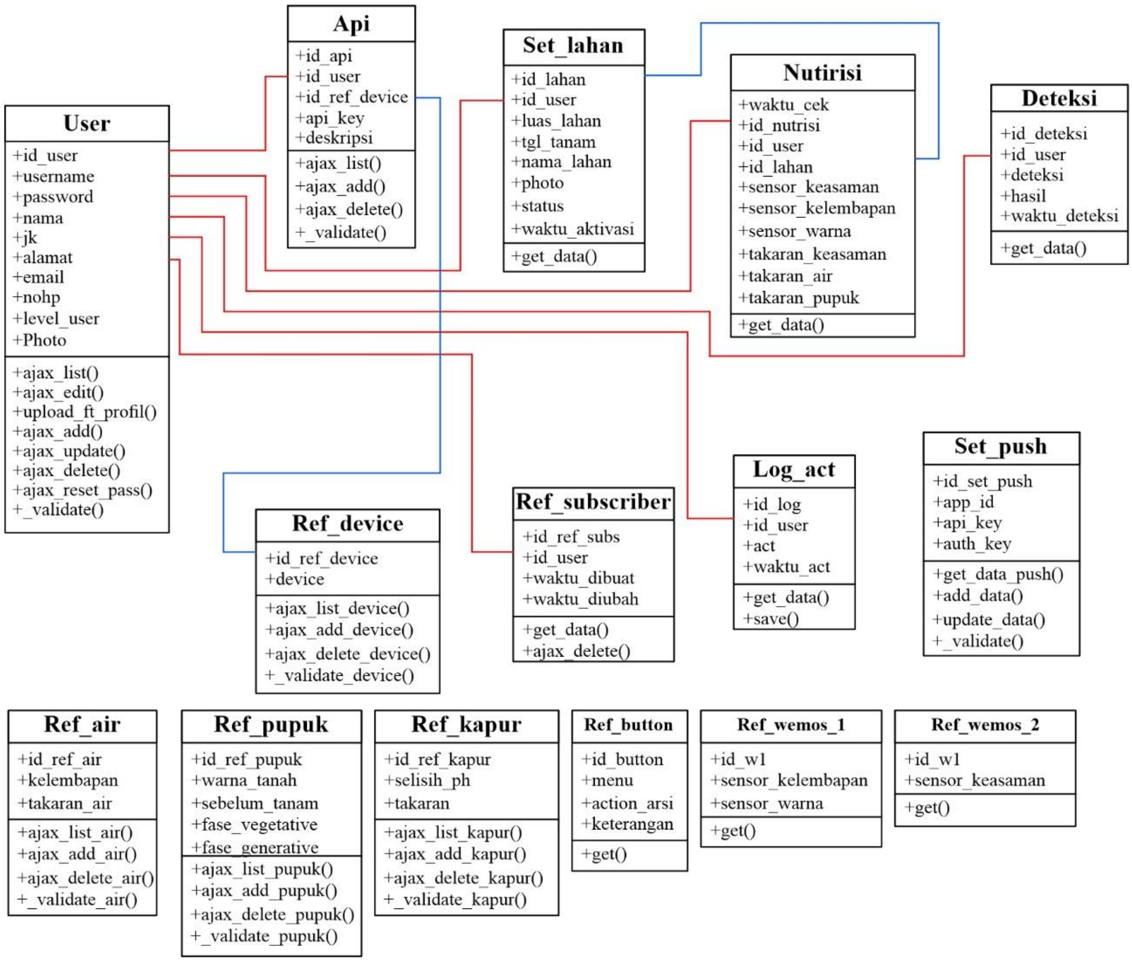
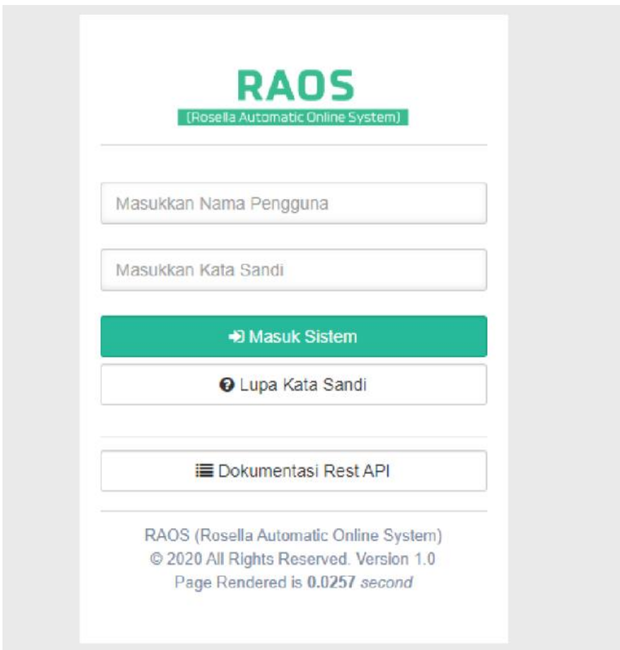


Figure 8. Class diagram.

The rosella pest detection app simplifies roselle plant management. The login page requires a username and password. Accessing program functionality requires credentials. Another feature lets users change their password if they forget it. After signing in, the application's homepage displays roselle pest detection options and information. Wemos Rest API is an application view. This page provides connections to Arduino, Wemos, and NodeMCU Rest APIs.

Fig. 9 shows the Wemos Rest API Menu, which connects Arduino, Wemos, and NodeMCU to applications. The user may connect with the device using Rest API by selecting the proper link. Users may manage, monitor, and configure roselle plants directly via the app. This rosella pest detection software provides Rest API documentation and ease of roselle plant management. This feature helps users learn how to utilize the application's Rest API to integrate and build new features. The rosella pest detection program makes caring for roselle plants and using linked gear easy and effective with a simple UI and valuable features.



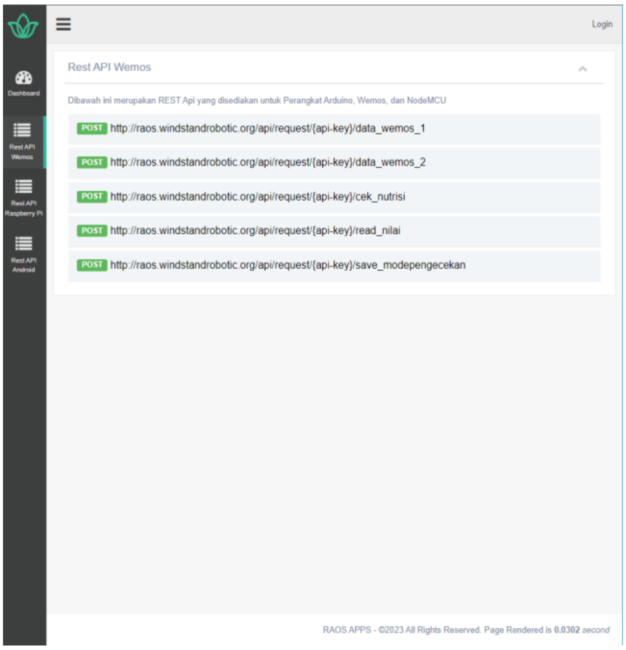


Figure 9. Web-based application display.

The design of the rest API server is a stage in designing a web service that can be accessed by various devices and of course connected devices can access data that is relevant to the needs of the device itself. Rest API Sever provides functions or functions that can be accessed by devices which of course have been previously registered and each connected device will have a special code or API key to be able to access the rest API that has been provided. This Api Key has many uses, such as recognizing user data, recognizing what functions are needed by each connected device and many more advantages with this API key. The following is a pseudocode for matching the thresholded image with training data or roselle flower templates:

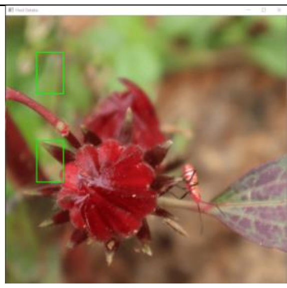
```
if(value > 0.01 and value < 0.258):

cv2.rectangle(test_image,top_left,bottom_
right, (0,255,0), 2)
   break
elif(value > 0.01 and value < 0.02):

cv2.rectangle(test_image,top_left,bottom_
right, (0,255,0), 2)
   continue
elif(value > 0 and value < 0.01):
   cv2.rectangle(test_image, top_left,
bottom_right, (0,255,0), 2)
   continue
elif(value < 0):

cv2.rectangle(test_image,top_left,bottom_
right, (0,255,0), 2)
   continue
else:
# cv2.imshow(test_image)
return None
print("Nilai : ",value,"\n")
```

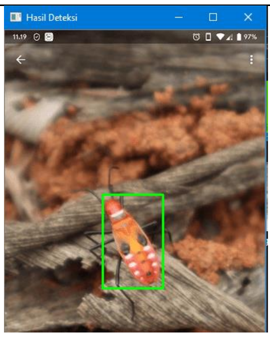
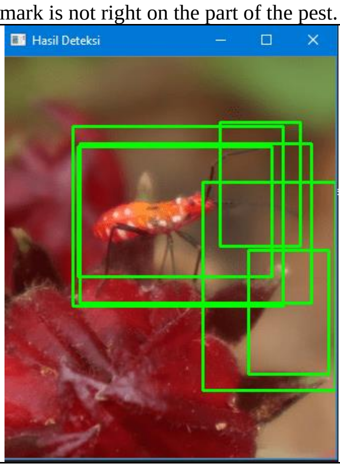
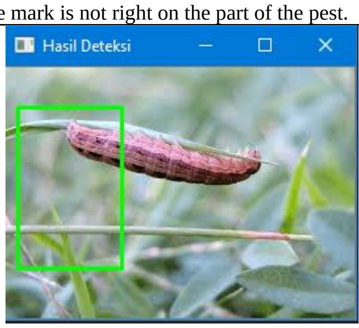
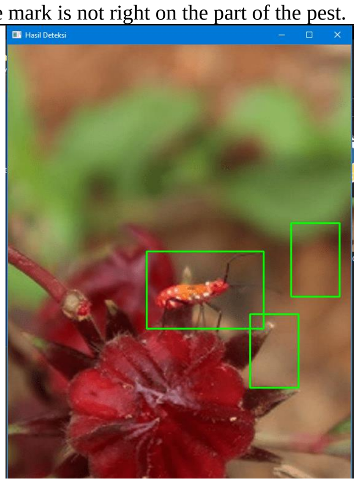
While the algorithm for matching the results of the thresholding image with training data or pest templates is as follows:

```
if(value > 0.02 and value < 0.06):

cv2.rectangle(test_image,top_left,bottom_ri
ght, (0,255,0), 2)
break
elif(value > 0.01 and value < 0.02):
continue
elif(value > 0 and value < 0.01):
continue
elif(value < 0):

cv2.rectangle(test_image,top_left,bottom_ri
ght, (0,0,255), 2)
continue
else:
return None
```

This object identification test is to determine the accuracy of the template matching method as a detection method in detecting similarities between the training data and the results of the detection of the Roselle object taken by the PiCamera module camera found on the raspberry pi. The detected object is changed using the thresholding method. Then it is matched with the template matching method where the training data has been stored in the database. Tests were carried out on Rosella plants cultivated by farmers in Cikondang Village, Cikijing District, Majalengka Regency with 12 tests. Each test hole or pest of Rosella plants is recorded with two results, namely detected or not detected. Tests were carried out by detecting pests and holes found in rosella plants which can be seen in Table II.

TABLE II. PEST/HOLE OBJECT DETECTION TEST

Camera Result Data	Pests/holes Identified Yes	Pests/holes Identified No
 Sample images taken directly from the Roselle field, the system managed to read/identify the presence of animals/pests in Roselle by marking a green box on the sample image, it is just that the mark is not right on the part of the pest.	✓	

 Sample images taken directly from the Roselle field, the system managed to read/identify the presence of animals/pests in Roselle by marking a green box on the sample image, it is just that the mark is not right on the part of the pest.	✓
 Sample images taken directly from the Roselle field, the system managed to read/identify the presence of animals/pests in Roselle by marking a green box on the sample image, it is just that the mark is not right on the part of the pest.	✓
 Sample images taken directly from the Roselle field, the system managed to read/identify the presence of animals/pests in Roselle by marking a green box on the sample image, it is just that the mark is not right on the part of the pest.	✓
	✓
 Sample images taken directly from the Roselle field, the system managed to read/identify the presence of animals/pests in Roselle by marking a green box on the sample image, it is just that the mark is not right on the part of the pest.	✓
 Sample images taken directly from the Roselle field, the system managed to read/identify the presence of animals/pests in Roselle by marking a green box on the sample image, it is just that the mark is not right on the part of the pest.	✓
 Sample images taken directly from the Roselle field, the system managed to read/identify the presence of animals/pests in Roselle by marking a green box on the sample image, it is just that the mark is not right on the part of the pest.	✓

This test is carried out to calculate the accuracy of the training data for the Roselle object. The percentage of success rate is calculated based on the Eq. (3):

$$\text{Percentage} = \frac{\text{Successful trial}}{\text{Number of trials}} \times 100\% \quad (3)$$

Identification of pests and holes on roselle plants from 12 trials there were 9 successful and 3 unsuccessful attempts to obtain 75% accuracy for the identification of pests and holes, which can be seen in Fig. 10.

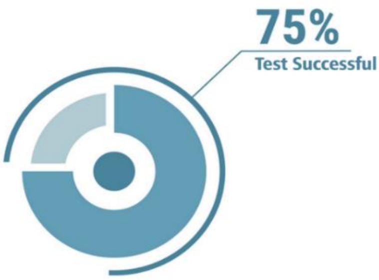


Figure 10. Identification test results.

The light that is on the object affects the process of identifying the object itself. The camera sensor used affects the quality of the captured image. In this study, the PiCamera module camera uses a sensor resolution of 5MP with a photo resolution of 640×480. To improve the functionality of this system, including the use of deep learning methods using the Convolutional Neural Network (CNN) algorithm for more accurate results. CNN proved to be much better than being able to recognize objects from blurs picture [24]. The use of Faster R-CNN can also be used for speed in detecting objects. The use of Raspberry Pi v4 and a camera module with a better and better resolution and sensor to get clearer images.

IV. Conclusion

Based on the test results of detection and classification of pests on Roselle plants and based on testing the web-based automated online rosella system application that has been designed and built. The detection and classification of pests on rosella plants was made using an embedded system based on python programming. Remote data communication can be achieved by utilizing internet of things technology. Using the thresholding method as the main method for detecting pests on roselle plants, to optimize the detection process, the template matching method to identify rosella and pests better because this method has the concept of equating the captured data with the training data that has been created. The accuracy of pest detection using thresholding and template matching methods based on test results reached 75%. The webservice created has also succeeded in responding to requests sent by sending real-time notifications regarding information on the detection results to farmers' smartphones.

Integration of the entire Rosella plant maintenance system is a future research work by completing advanced features such as soil moisture detection and the overall rosella plant maintenance process.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

A.B. and T.F.P. conducting research, quality control, and testing is intriguing. S.F. initial ideas, part of introduction and conclusion, text overview, investigation methodology and final checking. A.I.N. specializes in system architecture design, algorithm analysis, and system programming. And for A.B.'s composing of documents and correspondence. All authors had approved the final version.

Funding

This research was jointly funded by Universitas Padjadjaran (UNPAD) and Universitas Majalengka (UNMA), Indonesia.

Acknowledgment

Thanks to Universitas Majalengka (UNMA) for the support. Thank you also to the Rosella Farmers in Cikondang Village, Cikijing, Majalengka Regency, West Java, Indonesia for the co-operation. Our thanks also to Mr. Irawan Afrianto from the Universitas Komputer Indonesia (UNIKOM), for the supports. Thanks to the Ministry of Education, Culture, Research and Technology of the Republic of Indonesia (KEMDIKBUDRISTEK) for their assistance and guidance to the first author on behalf of Ade Bastian at the Scientific Article Writing Training held in Bandung, Indonesia, from June 6–11, 2022. Finally, thanks to Universitas Padjadjaran through Sri Fatimah, Ph.D. for funding this research.

References

[1] P. Wijayanti, *Red Roselle (Hibiscus Sabdariffa L) Cultivation and Its Secondary Metabolic Utilization*, Surakarta: Faculty of Agriculture Universitas Sebelas Maret Surakarta, 2010.

[2] W. G. A. Alshoosh, *Chemical Composition of Some Roselle (Hibiscus Sabdariffa) Genotypes*, Sudan: University of Khartoum, 1997.

[3] T. Astuti and S. Darmanti, "Development of roselle stem fiber (Hibiscus sabdariffa var.Sabdariffa) with different shade treatment and watering volume," *Buletin Anatomi dan Fisiologi*, vol. 18, no. 2, pp. 47–55, 2010.

[4] Y. Shi, Z. Wang, X. Wang, and S. Zhang, "Internet of things application to monitoring plant disease and insect pests," in *Proc. the 2015 International conference on Applied Science and Engineering Innovation*, 2015. doi: 10.2991/asei-15.2015.7

[5] C.-W. Wang, Y.-H. Yu, C.-H. Wu, R.-Y. Feng, K. Tandon, Y.-L. Chen, and S.-L. Tang, "Detection of pathogenic and beneficial microbes for roselle wilt disease," *Front Microbiol*, vol. 12, 756100, 2021. doi: 10.3389/fmicb.2021.756100

[6] C. Montiel-González, C. Montiel, A. Gómez, A. Pacheco, and F. Bautista, "Development and validation of climatic hazard indicators for roselle (Hibiscus sabdariffa L.) crop in dryland agriculture," *Ecological Indicators*, vol. 121, 2021. doi: 10.1016/j.ecolind.2020.107140

[7] N. Hassan, M. Shimizu, and M. Hyakumachi, "Occurrence of root rot and vascular wilt diseases in roselle (Hibiscus sabdariffa L.) in upper Egypt," *Mycobiology*, vol. 42, no. 1, pp. 66–72, 2014. doi: 10.5941/MYCO.2014.42.1.66

[8] G. UshaDevi and B. V. Gokulnath, "A survey on plant disease prediction using machine learning and deep learning techniques," *Inteligencia Artificial*, vol. 23, no. 65, pp. 136–154, 2020. doi: 10.4114/intartif.vol23iss65pp136-154

[9] A. S. H. Abdel-Moniem and T. E. Abd El-Wahab, "Insect pests and predators inhabiting roselle plants, Hibiscus sabdariffa L., a medicinal plant in Egypt," *Archives of Phytopathology and Plant Protection*, vol. 39, no. 1, pp. 25–32, 2006.

[10] D. R. Raju, *et al.*, "Image segmentation by using histogram thresholding," *International Journal of Computer Science Engineering and Technology (IJCSET)*, vol 2, issue 1, pp. 776–779, 2012.

[11] P. Guruprasad, *Overview of Different Thresholding Methods in Image Processing*, 2020.

[12] A. Rosenfield, "Connectivity in digital pictures," *Journal of the ACM*, vol. 17, no. 1, pp. 146–160, 1970.
[13] P. S. Khandare and M. A. Isalkar, "A survey paper on image segmentation with thresholding," *International Journal of Computer Science and Mobile Computing*, vol. 3, issue 1, 2014.
[14] F. Faithpraise, P. Birch, R. Young, J. Obu, B. Faithpraise, and C. Chatwin, "Automatic plant pest detection & recognition using k-means clustering algorithm & correspondence filters," *International Journal of Advanced Biotechnology and Research*, vol. 4, pp. 1052–1062, 2013.
[15] J. Chaki and R. Parekh, "Plant leaf recognition using shape based features and neural network classifiers," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 2, no. 10, 2011. doi: 10.14569/IJACSA.2011.021007
[16] R. M. Rakholia, J. H. Tailor, J. R. Saini, J. Kaur, and H. Pahuja, "Groundnuts leaf disease recognition using neural network with progressive resizing," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 13, no. 6, 2022. doi: 0.14569/IJACSA.2022.0130611
[17] D. R. Raju *et al.*, "Image segmentation by using histogram thresholding," *International Journal of Computer Science Engineering and Technology (IJCSET)*, vol 2, issue 1, pp. 776–779, 2012.
[18] Y. Liu, J.-M. Xue, and H. Li, "The study on the image thresholding segmentation algorithm," in *Proc. International Conference on Unteliggent Systems Research and Mechatronics Engineering (ISRME)*, 2015, pp. 2306–2310.
[19] S. Fatimah and Q. K. Wazni, "Development of tourism village based on Agro-creative and disaster consideration, case of Cikondang village, Cingambul District, Majalengka Regency," on *Proc. IOP Conf. Ser.: Earth Environ. Sci.*, 2020, vol. 466, 012022. doi: 10.1088/1755-1315/466/1/012022
[20] S. Fatimah, K. Ganjar, D. Gunardi, L. M. Farida, D. Supyandi, Rosalinda, and Y. Hidayat. (2019). Factors affecting the empowerment of rosella farmers to realize rural agroecotourism. [Online]. Available: https://www.atlantis-press.com/article/125909401
[21] L. Wang and J. Bai, "Threshold selection by clustering gray levels of boundary," *Pattern Recognition Letters*, vol. 24, pp. 1983–1999, 2003.
[22] F. M. Abubakar. (January 2013). Study of image segmentation using thresholding technique on a noisy image. *International Journal of Science and Research (IJSR)*. [Online]. 2(1). pp. 49–51. Available: https://www.ijsr.net/get_abstract.php?paper_id=IJSROFF130201024
[23] S. Khurana, "Comparative study on threshold techniques for image analysis," *International Journal of Engineering Research & Technology (IJERT)*, vol. 4, issue 6, June 2015. doi: 10.17577/IJERTV4IS060563
[24] M. A. Hosaain and M. S. A. Sajib, "Classification of image using Convolutional Neural Network (CNN)," *Global Journal of Computer Science and Technology: Neural & Artificial Intelligence*, vol. 19, issue 2, 2019.