TẠP CHÍ KHOA HỌC
TRƯỜNG ĐẠI HỌC QUY NHƠN

Ứng dụng công nghệ xử lý ảnh trong phân loại cà chua sau thu hoạch

Nguyễn Trường Duy[1], Trần Đình Huy Đức[1], Bùi Văn Vũ[2,*], Đỗ Văn Cần[2]

[1]Sinh viên ngành Kỹ thuật điện - Điện tử, Khoa Kỹ thuật và Công nghệ, Trường Đại học Quy Nhơn, Việt Nam

[2] Khoa Kỹ thuật và Công nghệ, Trường Đại học Quy Nhơn, Việt Nam

Ngày nhận bài: 29/05/2020; Ngày nhận đăng: 20/07/2020

TÓM TẮT

Bài viết này trình bày về việc ứng dụng công nghệ xử lý ảnh trong việc phân loại cà chua theo màu sắc và kích thước quả. Việc xác định mức độ chín và kích thước của cà chua một cách tự động sẽ được đưa ra phân tích chi tiết và cụ thể trong bài báo này. Cụ thể, hệ thống sẽ thu thập hình ảnh từ máy ảnh được đặt trên băng chuyền. Hình ảnh sau khi được chụp bằng máy ảnh sẽ được đưa về bộ xử lý trung tâm để tính toán và phân tích. Các thuật toán về xác định biên và màu sắc của đối tượng sẽ được ứng dụng. Những quả cà chua không đạt yêu cầu sẽ được phát hiện dựa trên số điểm ảnh và kích thước đường biên của đối tượng được chụp. Mục tiêu của hệ thống này là để thay thế khâu phân loại cà chua theo hình thức thủ công vốn còn tồn tại các nhược điểm như độ chính xác chưa cao, năng suất lao động thấp.

Keywords: *Xử lý ảnh, cà chua, màu sắc, kích thước, tự động.*

*[*]Tác giả liên hệ chính.
Email: bvvu@ftt.edu.vn*

JOURNAL OF SCIENCE
QUY NHON UNIVERSITY

Applying of image processing technology in classifying tomatoes after harvest

Nguyen Truong Duy[1], Tran Dinh Huy Duc[1], Bui Van Vu[2,*], Do Van Can[2]

[1]Undergraduate student majoring in electrical and electronic engineering, Faculty of Engineering and Technology, Quy Nhon University, Vietnam
[2] Faculty of Engineering and Technology, Quy Nhon University, Vietnam

Received: 29/05/2020; Accepted: 20/07/2020

ABSTRACT

This paper applies image processing technology in classifying tomatoes by color and size. An automatic determination of ripeness and size of tomatoes will be analyzed in detail in this paper. Actually, the system will collect images from cameras placed on a conveyor belt. The images will be taken to the central processor for calculation and analysis. Algorithms is used to estimate the boundary and color of the object will be applied. Tomatoes that have low quality will be detected based on the number of pixels and border size of the subject being captured. The goal of this system is to replace manual grading of tomatoes that still have shortcomings such as low accuracy, low labor productivity.

Keywords: *Image processing, tomatoes, color, size, automatic.*

1. INTRODUCTION

Vietnam is a country with a high proportion of agricultural sectors. However, the labor productivity of this industry is not high. Therefore, the application of science and technology in the field of production and export of agricultural products is very necessary. Currently, the post-harvest classification of agricultural products, particularly tomatoes in our country, is mainly done by hand. When using human power, tasks that require high concentration and repetition make the eyes strained and tired, so it is difficult to guarantee accurately, especially in the case of high-frequency labor. These forms of manual labor need to be replaced by automatic or semi-automatic models to overcome the above weakness. This improves the accuracy, labor productivity and product quality so that Vietnam's agricultural products can compete with the agricultural products of developed countries in the world.

Today, one of the technologies that is supposed to replace the manual classification of agricultural products after harvest is image processing.[1-7]

In this study, the author will apply image processing technology to classify tomatoes after harvest. The technology mentioned by the author can overcome the shortcomings of the visual classification of tomatoes used in our country today such as: high accuracy, high labor productivity, quality of production, good product. Different image processing algorithms will be used in author's study.[8-12]

2. STUDY METHODS AND TECHNIQUES

2.1. Method of classifying tomatoes according to its color

In order to classify the ripeness of tomatoes,

*Corresponding author.
Email: bvvu@ftt.edu.vn

the system relies on the images obtained from the camera. First, images captured from the camera are converted to the HSV color space. Next, the image will be divided into 3 separate color channels H, S, V and then a series of comparison analysis algorithms are used to output the results. The analysis algorithm for tomato fruit color is shown in detail in Figure 1.[4,12]

Figure 2 shows a photo of tomatos taken from a camera. This image is represented under the RGB color space.[2,3,12,13] RGB color spaces use a complementary pattern where red, green, and blue light are combined each other in different ways to form different colors.

The RGB color model is depicted in Figure 3. The advantage of the RGB model is that it is convenient for color correction, but this color model is not suitable for image processing. Therefore, the image needs to be converted to the HSV color space (figure 4).[3,12,13,14]

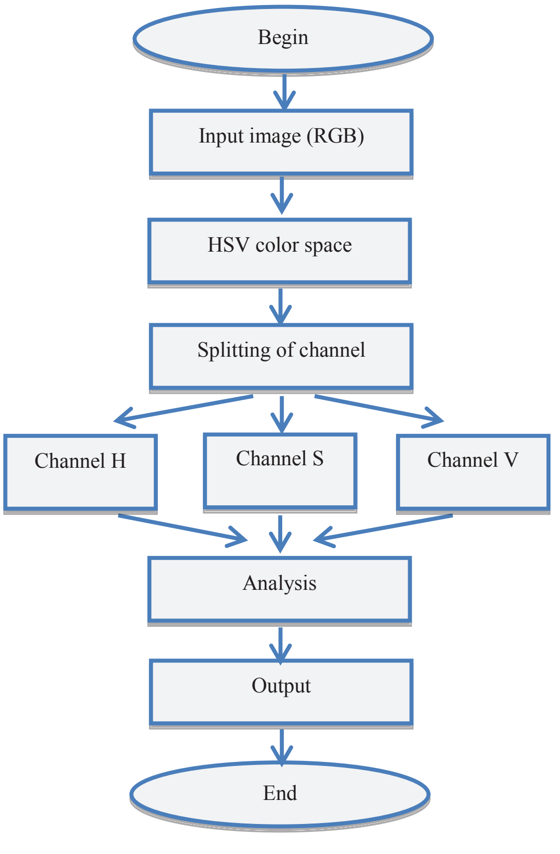


Figure 1. The algorithm determines the maturity level of the tomatoes.

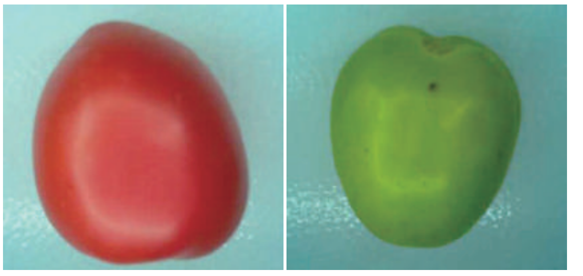


Figure 2. Tomato images were taken from the camera.

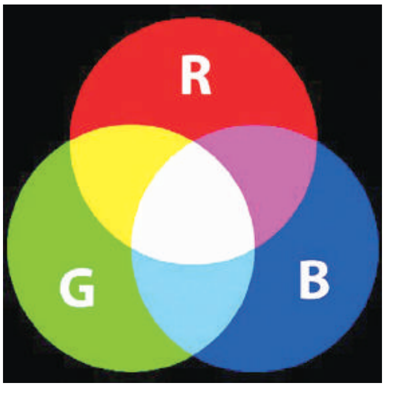


Figure 3. RGB color space.

To convert from RGB color space to HSV color space, formulas (1) - (5) are used.[8,9]

$$H = \arccos\left(\frac{\frac{1}{2}(2R - G - B)}{\sqrt{(R-G)^2 - (R-G)(G-B)}}\right) \quad (1)$$

$$H = H_1 \quad \text{if} \quad B<G \quad (2)$$

$$H = 360^o - H_1 \quad \text{if} \quad B>G \quad (3)$$

$$S = \frac{\max(R,G,B) - \min(R,G,B)}{\max(R,G,B)} \quad (4)$$

$$V = \frac{\max(R,G,B)}{255} \quad (5)$$

Where,

R, G, B: are values representing the intensity of red, green, and blue in the RGB color space, respectively (0 -255).

H: The value represents the color area.

S: value that represents color saturation.

V: value represents luminous intensity.

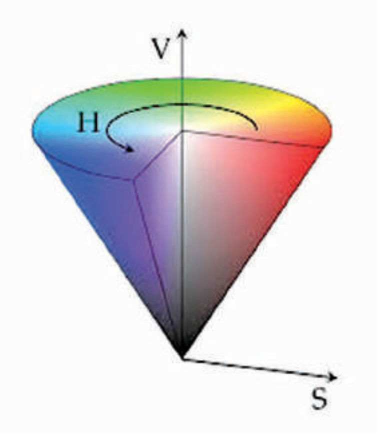


Figure 4. HSV color space.

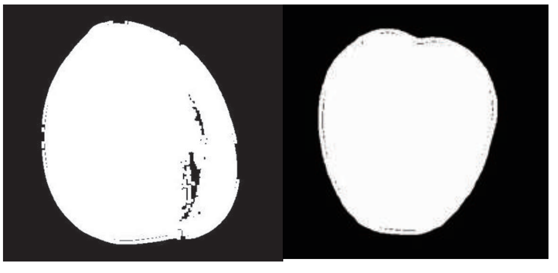


Figure 5. The binary image of the object.

After the object image is converted from the RGB color space to the HSV color space, we obtain the values of the color channels. From the values of these color channels, we obtain a binary image of the object (see Figure 5). Based on this binary image, we determine the number of pixels in the each object image, the number of white pixels of each color in the object image and the number of black pixels. Once these data are available, the author can determine the proportions of each color and use this result to determine the maturity level of the fruit.

2.2. Method of classifying tomatoes according to fruit size

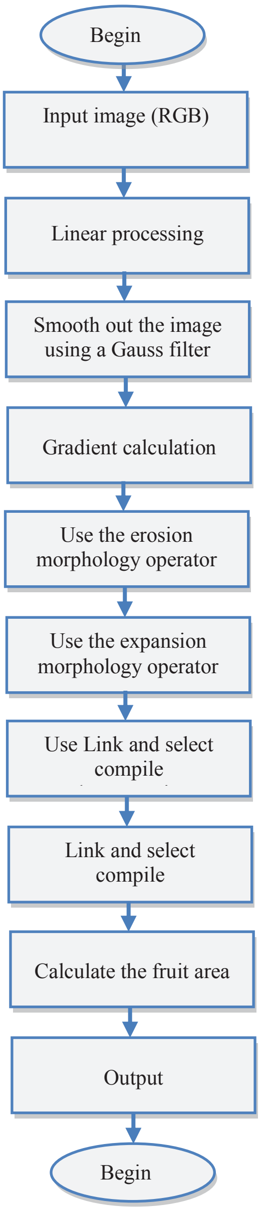


Figure 6. Diagram of the fruit size classification process.

To be able to remove tomatoes that do not satisfy the size requirements, the authors used a method of determining the boundary of the object image captured from the camera.[10,11] The size of the analyzed tomato is the area of the figure surrounded by the boundary.

Derived from related concepts about the boundary point and boundary,[1] there are two methods used to detect the boundary of an object. The first method is the direct method. This method aims to enhance the boundary based on the variation in the pixel's luminance value. The direct method using mainly derivative techniques (Gradient, Sobel, Prewitt, Laplace, Canny, ...).[1,7,15] And, the second method is the indirect method. This method often relies on the surface texture of the image to divide the image into regions, the border between those regions is the boundary need to be dectected. Indirect edge detection method is difficult to set up but works well when there is small variation in luminance.[1,7,15]

Here, the author uses the Canny boundary detection method, which is commonly used because it has the advantage of being less affected by noise.[1] The process diagram of fruit size determination is shown in Figure 6.[11] Canny boundary detection method is shown in formula (6) - (12).[1,7,10,11]

We have the derivative of a filtered image: $\nabla f = \nabla(G \otimes I) = f_x + f_y$, where f_x and f_y is the partial derivative of f in terms of x and y respectively.

Hence, we obtain formula (6):

$$\nabla f = \nabla(G \otimes I)_x + \nabla(G \otimes I)_y = \nabla(G_x \otimes I) + \nabla(G_y \otimes I) \quad (6)$$

Taking the partial derivative of G for x and y, we get:

$$G_x(x,y) = \frac{-x}{\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}} \quad (7)$$

$$G_y(x,y) = \frac{-y}{\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}} \quad (8)$$

Where,

x, y: are directions

σ : width control parameter

Furthermore, the Gaussian filter is decoupled so that convolutional charges can be performed separately for x and y. See equations (9) and (10):

$$G_x(x,y) = G_x(x) \otimes G(y) \quad (9)$$

$$G_y(x,y) = G_y(y) \otimes G(x) \quad (10)$$

From (6), (9) and (10), we have (11) and (12):

$$f_x(x,y) = G_x(x) \otimes G(y) \otimes I \quad (11)$$

$$f_y(x,y) = G_y(y) \otimes G(x) \otimes I \quad (12)$$

After Gaussian blurring, the image pixel values are extracted for low threshold and high threshold values. The authors use the Gradient operator to calculate the local gradient of amplitude and direction.

Assuming that g_1 and g_2 are gradients in x and y directions, respectively, then the amplitude of the gradient at the point (m, n) and the direction of the boundary at the point (m, n) are calculated by equations (13) and (14):[1,11]

$$A_0 = g(m,n) = \sqrt{g_1^{\,2}(m,n) + g_2^{\,2}(m,n)} \quad (13)$$

$$\theta_r(m,n) = \arctan \frac{g_2(m,n)}{g_1(m,n)} \quad (14)$$

After using the Gradient operator to calculate the local gradient, the authors use the erosion morphology operator to remove white noise.[1] White points with a low pixel value are removed. Because the morphology operator removes white noise, the thickness and size of the boundary will be reduced. Therefore, the authors continue to use the expansion morphology operator with the aim of increasing the size of the boundary after using noise and erosion removal algorithms. Now we get a curve connecting all continuous points of the same color or intensity. This curve is the boundary of the object. Figure 7 shows a picture of the tomato after marginal extraction.

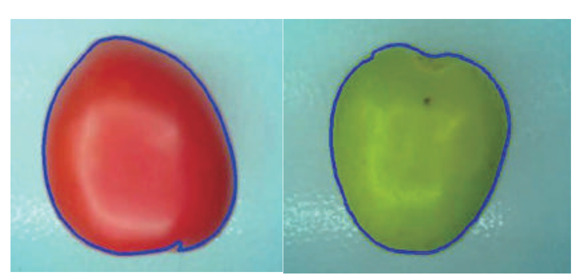


Figure 7. Photo after determining border.

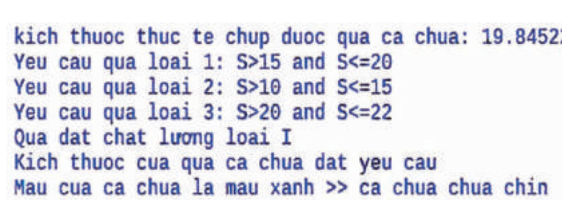


Figure 8. Cross-sectional area of above red tomato.

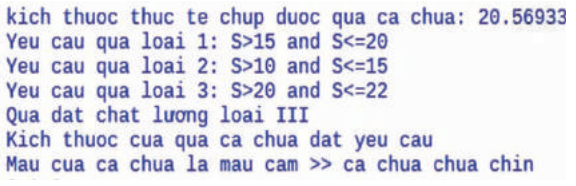


Figure 9. Cross-sectional area of above green tomato.

After we have the boundary of the object image, we compute the cross-sectional area of the fruit. The cross-sectional area of the fruit is here understood as the area delimited by the boundary of the object image. Figures 8 and 9 show cross-sectional area data of two tomatoes taken from the experimental model. Figure 8 shows data for red tomatoes (see Figure 7) and Figure 9 shows data for green tomatoes (see Figure 7).

2.3. Building of the system program

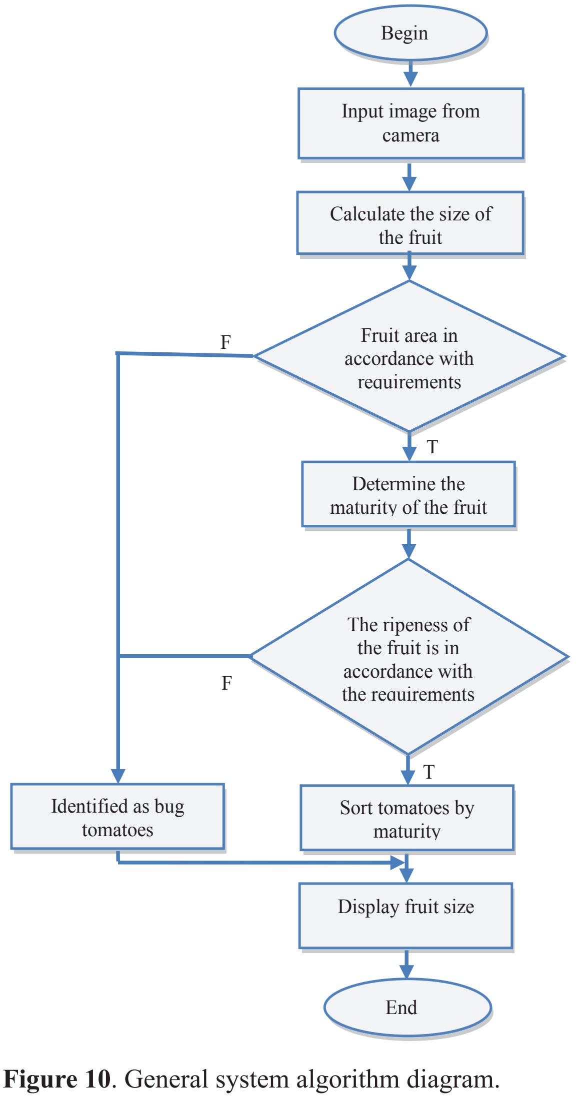


Figure 10. General system algorithm diagram.

From the analytical results in sections 2.1 and 2.2, a general algorithmic scheme is built for an automatic tomato classification system. This algorithmic automatic tomato classification scheme is shown in Figure 10.

From the general algorithm diagram in Figure 10, a block diagram of the classification system is proceed to built. The central controller of the system is raspberry, Arduino Uno, and PLC S7-200 devices (see Figure 11). From the diagram shown in Figure 11, authors select the types of equipment to build the experimental model as shown in Figure 12 and Figure 13.

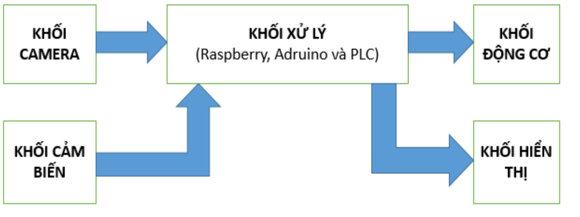


Figure 11. Diagram of the system's blocks.

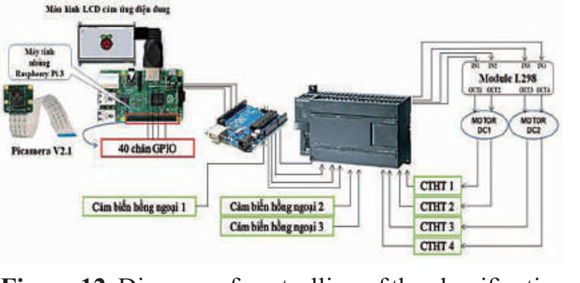


Figure 12. Diagram of controlling of the classification process.

2.4. Experimental results

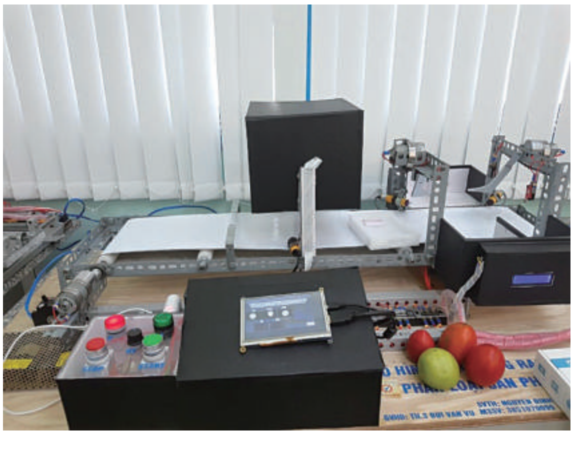


Figure 13. Experimental model.

The experimental model includes the following blocks:

a. Processing block

Kit Raspberry Pi 3 Model B +: Used to receive image signals from the camera, perform image processing according to pre-programmed program, and send signals to PLC.

PLC S200: Receiving signals from Raspberry and transmits signals to actuators of a sorting system.

b. Camera block

The Camera block has the function of collecting real-world image signals and converting them to electrical signals and sending data to the Raspberry Pi unit in the model using Raspberry Pi camera.

c. Sensor block

The sensor block used to detect the fruit has reached the position of the camera in the model uses infrared sensor E18-D80NK.

d. Actuator block

This unit receives signals from S7 -200 PLC to control the swing arms to bring products with different degrees of ripeness to different boxes.

e. Display block

The display block has the function of displaying information about the color and size of the fruit. In the model using a 7 inch LCD (Raspberry Pi) screen.

The entire product classification process is as follows:

Step 1: When starting up the system, the conveyor belt brings the sorted tomatoes to the camera's location. When the sensor detects that the fruit has reached the camera position. Raspberry will take the image signal of the object and process the image (see Figure 14).

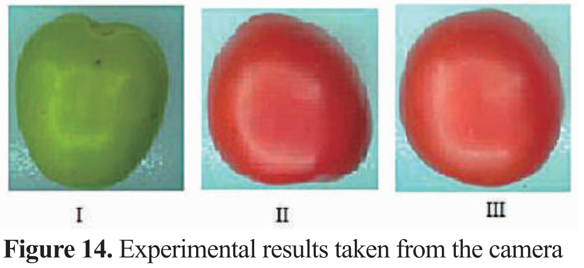


Figure 14. Experimental results taken from the camera

Step 2: An algorithm to determine the size of the fruit will be done by raspberry. The size of the fruit will be determined based on the area delimited by the boundary of the object's image. The experimental results of the boundary determination process are shown in Figure 15. In the system developed, the tomato subjects were classified into four groups based on fruit size. Particularly, Group 1, the fruit has a size of 15-20 (cm^2); Group 2, the fruit has a size of 10-15 (cm^2); Group 3, the fruit has a size of 20-22 (cm^2), and Group 4, the fruit has a size of smaller than 10 (cm^2) or larger than 22 (cm^2) (see Figure 17, Figure 18 and Figure 19). In case of unsatisfied fruits (a size of smaller than 10 (cm^2) or larger than 22 (cm^2)), the system will sort immediately without any color determination. The fruits of these different groups are classified and put in different containers.

After classifying tomatoes according to fruit size, the system will determine the maturity level of the fruit based on the color of the fruit. In terms of color, the system only selects fruits from light red to red. Fruits of other colors are rejected because the fruit is still green or the fruit is overripe.

Figure 16 shows an experimental result taken from Raspberry's memory when determining the color of an object based on the number of white pixels of each color.

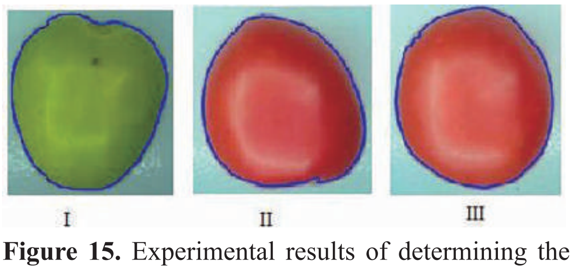


Figure 15. Experimental results of determining the object boundary.

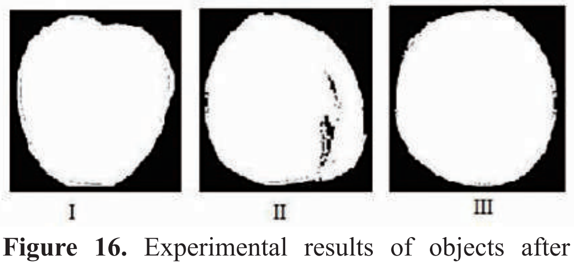


Figure 16. Experimental results of objects after converting to binary images.

Step 3: Display information on the LCD screen.

In this step, the maturity and fruit size information will be displayed on an LCD screen connected directly to the Raspberry (see Figures 17, 18, and 19).

Figure 17 shows the first tomato's data (see Figure 15) displayed on the Raspberry screen. This fruit is analyzed by the system as an unripe tomato. At the same time, the system also analyzes this fruit with a cross-sectional area of 19.84522 (cm^2), while the experimental measured data was about 19 (cm^2).

Figure 18 shows the second tomato's data (see Figure 15) displayed on a Raspberry screen. The fruit is analyzed by the system to be orange in color. At the same time, the system also analyzes this fruit with a cross-sectional area of 20.56933 (cm^2), while the experimental measured data was about 21.4 (cm^2).

Figure 19 shows the third tomato's data (see Figure 15) displayed on a Raspberry screen. This fruit was analyzed by the system to have a cross-sectional area of 25.32344 (cm^2), while the experimental measured data was about 25.5 (cm^2). Because in the program programmed for Raspberry, tomatoes are too big, larger than 22 (cm^2), are removed without analyzing its color. Therefore, the system does not need to analyze its color for this fruit (see Figure 19).

```
kich thuoc thuc te chup duoc qua ca chua: 19.84522
Yeu cau qua loai 1: S>15 and S<=20
Yeu cau qua loai 2: S>10 and S<=15
Yeu cau qua loai 3: S>20 and S<=22
Qua dat chat lương loai I
Kich thuoc cua qua ca chua dat yeu cau
Mau cua ca chua la mau xanh >> ca chua chua chin
```

Figure 17. Information display about maturity level and size of the first fruit.

```
kich thuoc thuc te chup duoc qua ca chua: 20.56933
Yeu cau qua loai 1: S>15 and S<=20
Yeu cau qua loai 2: S>10 and S<=15
Yeu cau qua loai 3: S>20 and S<=22
Qua dat chat lương loai III
Kich thuoc cua qua ca chua dat yeu cau
Mau cua ca chua la mau cam >> ca chua chua chin
```

Figure 18. Information display about maturity level and size of the fruit.

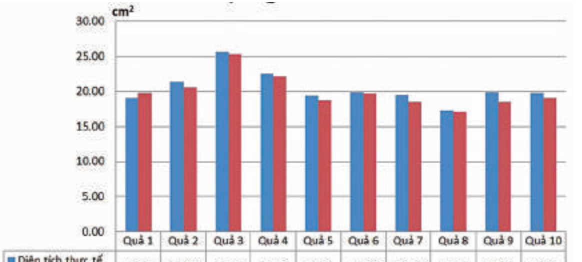
```
kich thuoc thuc te chup duoc qua ca chua: 25.32344
Yeu cau qua loai 1: S>15 and S<=20
Yeu cau qua loai 2: S>10 and S<=15
Yeu cau qua loai 3: S>20 and S<=22
kich thuoc cua qua ca chua khong dat yeu cau
loi
```

Figure 19. Information display about maturity level and size of the third fruit.

The results of fruits' size analysis in Figures 17, 18, and 19 show that the built-in automatic tomato classifying system can determine the size of tomatoes with a maximum error of 5%.

In the study, the author has tested on 200 tomato samples on the experimental model built. The experimental results show that, in terms of color grading criteria, the system accurately classifies the colors of 200 experimental tomato samples. Regarding the criteria of product classification according to the size of the fruit, the system determines the size of the fruit with an error of not more than 1.32 (cm^2).

Figure 20 shows analytical data of 10 samples per 200 samples of tomatoes that were tested. The statistics show that the maximum size error of the system is 1.32 (cm^2) and there is no error in determining fruit color. Also, the process from taking pictures to getting classification results on raspberry for a tomato sample is approximately 0.6 seconds.



	Quả 1	Quả 2	Quả 3	Quả 4	Quả 5	Quả 6	Quả 7	Quả 8	Quả 9	Quả 10
Diện tích thực tế (cm2)	19.10	21.44	25.58	22.55	19.47	19.89	19.50	17.30	19.87	19.85
Diện tích tính toán (cm2)	19.85	20.57	25.32	22.14	18.70	19.72	18.49	17.11	18.55	19.05

Figure 20. Analytical data of 10 tomato samples.

3. CONCLUSION

The paper presents the application of image processing technology in the automatic tomato grading system after harvest. In the paper, authors have applied image processing algorithms to identify the maturity level and size of tomatoes to classify. Tomatoes of different sizes and satisfactory maturity will be classified and sent to different containers. Beside, fruits that are too large or too small, or too ripe or green will be rejected by the system. Furthermore, the author has also successfully built an experimental model of the system. The experimental results show that the system is stable, has high accuracy and high productivity.

4. REFERENCES

1. Lương Mạnh Bá, Nguyễn Thanh Thủy. *Nhập môn xử lý ảnh số*, Nhà xuất bản Khoa học và Kỹ thuật, Hà Nội, 2006.
2. P. Koirala. *RGB Color Space*, Department of Computer Science and Statistics, University of Joensuu, 2017.
3. D. Cardani. Adventures in hsv space, *Laboratorio de Robótica*, Instituto Tecnológico Autónomo de México, 2001.
4. R. R. Mhaski, P.B. Chopade and M.P. Dale of MES College of Engineering. *Determination of Ripeness and Grading of Tomato using Image Analysis on Raspberry Pi,* International Conference on Communication, Control and Intelligent Systems (CCIS), 2015.
5. Nguyen Duc Thong; Nguyen Truong Thinh; Huynh Thanh Cong. *Mango Classification System Uses Image Processing Technology and Artificial Intelligence*, 2019 International Conference on System Science and Engineering (ICSSE), IEEE, 2019.
6. Andrés Felipe Sánchez-Aguiar, Alberto Mario Ceballos-Arroyo and Albeiro Espinosa-Bedoya. *Toward the recognition of non-defective coffee beans by means of digital image processing,* Symposium on Image Symposium on Image (22th), Signal Processing and Artificial Vision (STSIVA), 2019.
7. G. Xin, C. Ke, H. Xiaoguang. *An improved Canny edge detection algorithm for color image*, 10th International Conference on Industrial Informatics, 2012.
8. P. Li, Y. Huang and K. Yao. *Multi-algorithm Fusion of RGB and HSV Color Spaces for Image Enhancement*, In Proceedings of the International Conference on 37th Chinese Control Conference (CCC), 2018.
9. K.N. Plantaniotis and A.N. Ventsanopoulos. *Color image processing and applications,* Springer, 2000.
10. Vinh Du Mai, D. Miao, R. Wang and H. Zhang. *An Improved Method for Vietnam License Plate Location,* 2011 International Conference on Computation and Information Sciences, 2011.
11. Ali, M. H. Mouad, P. Yannawar, and A. T. Gaikwad. *Study of edge detection methods based on palmprint lines*, 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), 2016.
12. P. Ganesan, V.Rajini, B.S. Sathish Khamar and Basha.Shaik. *HSV Color Space Based Segmentation of Region of Interest in Satellite Images,* International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2014.
13. L. Feng, L. Xiaoyu, C. Yi. *An efficient detection method for rare colored capsule based on RGB and HSV color space*, International Conference on Granular Computing (GrC), 2014.
14. R. P. Salunkhe & A. A. Patil. *Image Processing for Mango Ripening Stage Detection: RGB and HSV method,* International Conference on Image Information Processing (3rd), 2015.
15. J. Zhang. *Edge detection in glass fragmentation images based on One Order Differential operator*, Second International Conference on Computer Engineering and Applications, 2010.