International Journal of Engineering & Technology, 7 (3) (2018) 1439-1445

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet.v7i3.11862
Research paper

Fruit monitoring system using multi-layered neural network

Rui Xu [1], Tae-Hyun Cho [1], Chang Kil Kim [2], Bonghwan Kim [3], In Soo Lee [1] *

[1] School of Electronic Engineering, Kyungpook National University, Daegu 41566, Korea
[2] Department of Horticultural Science, Kyungpook National University, Daegu 41566, Korea
[3] School of Electronic and Electrical Engineering, Daegu Catholic University, Gyeongbuk 38430, Korea
**Corresponding author E-mail: insoolee@knu.ac.kr*

Abstract

A fruit monitoring system based on image processing technology and multi-layer neural network is proposed. The advantage of the proposed fruit monitoring system allows it to be remotely controlled by PCs and the graphical user interface (GUI) program by LabVIEW which has been designed for more intuitive and convenient operation of this system. In addition, the neural network can reduce nonlinearity of the system compared to the calculation based system. Therefore, experienced workers and novices can easily judge the ripeness of the fruits using the GUI program without necessarily going to the orchards. In this study, the color is used as a criterion to judge the maturity of tomatoes. Ripe tomatoes will appear to be red, while the unripe tomatoes will be green in color. The region of interest (ROI) function and Canny edge detection are applied to crop the image and remove the background, then the pixel data obtained are to supply the use of neural network. After that the maturity level of tomatoes is judged by the neural network. In laboratory test, 50 experiments have been down, 48 of which were successful, 2 of which failed, so the recognition rate was 96%. The experiments of this fruit monitoring system in the greenhouse on real growing tomatoes has been conducted. Therefore, 10 experiments on the red and green tomatoes has been conducted, respectively. As a result, the recognition rate of the red tomatoes is 100%, and recognition rate of the green tomatoes is 90%. The experimental results show that the proposed mobile fruit monitoring system has a very high recognition rate of accuracy.

Keywords: *Fruit Monitoring System; GUI; Image Processing; Neural Network.*

1. Introduction

Recently, the number of experienced workers in the countryside has decreased because of the rise in the age of the population and the increase of novices in the countryside. In addition, an increasing number of people are suffering from the heat of the ever-increasing hot weather [1-2]. Presently, increasingly intelligent products have been applied in agriculture, especially for judging the maturity of the fruits. Generally, the maturity of the fruits is judged by their shapes [3] and colors [4]. Given that the size and shape of ripe fruits vary, the colors of the fruits have become important criteria in judging their maturity. Therefore, proposed method is to judge the ripeness based solely on the color of the fruit. For example, in this study, color is used as criteria to judge the maturity of tomatoes. Ripe tomatoes will appear to be red, while the unripe fruit will be green in color. Zhao and Hou [5] wrote a paper that uses the Hue saturation intensity (HIS) color model and back propagation (BP) neural network to grade apples. The system has a good performance of about 95% accuracy. Ji and Yuan [6] wrote a paper that uses the HIS color model and particle swarm optimization (PSO) neural network to grade apples. Hu et al. [7] wrote a paper that uses the HSI color model and Hamming neural network to grade apples. These three papers utilized the HSI color model and neural network to grade the fruit. However, they only consider the Hue component and ignore the saturation component correlated to the brightness, which require the light to be constant. Therefore, these devices can only be operated in the lab and cannot be moved, reducing their practicality and convenience.

To obtain the images of the fruits, an image processing technology is used, which has been mainly used in the field, such as factory automation, military, medical treatment, remote prospecting, monitoring security and many others. To facilitate a role in the intelligence camera and intelligent transportation system (ITS), internet of things (IoT) technology, and to make the users recognize the surroundings, the related research is being conducted [8]. The appearance of the OpenCV (Open Source Computer Vision) makes it possible for the developer to apply various algorithms, such as image recognition and face recognition, for easier development of the image processing system [9]. Neural networks have proven to be a promising paradigm for intelligent systems. It has been trained to perform complex functions in various fields of application, such as pattern recognition, identification, and classification [10]. The ability to learn complex nonlinear input output relationships, the use of sequential training procedures, and their adaptability to data are the three outstanding characteristics of the neural network. Some popular modules of neural network have been shown the ability of associative memory and learning [11-13]. In order to allow the network to perform efficiently a specific classification/clustering task, the learning process is comprised of updating the network architecture and modifying the weights between the neurons. Many applications of neural networks have been reported to be used in the images in agriculture. For example, the accuracy of the classification of potted plants can be greater than 99% [14]. The accuracy rate is 95% when apples are graded by color [15]. The accuracy for the classification of logs for defects when apply the computed tomography imagery can be 95% [16]. The classification of wheat kernels by color can be greater than or equal to 98% accurate [17]. In general, neural networks can efficiently model a variety of input and output relationships. Compared to a procedural model, it has the advantage of less execution time [18-19].

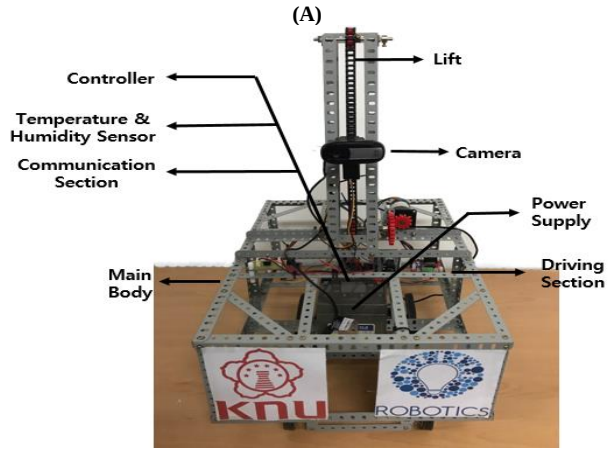
In this study, a fruit monitoring system is developed. The advantage of this fruit monitoring system is that it can be remotely controlled by PCs. Without going to the orchard by themselves, skilled workers, as well as novices, can easily judge the maturity level of the fruits from afar. We have incorporated temperature and humidity sensors in this system to obtain the real-time temperature and humidity readings to monitor the surroundings better. The graphical user interface (GUI) program by LabVIEW has been designed for more intuitive and convenient operation of the system. To achieve this target, the image processing technology and multi-layer neural network to help in judging the maturity level of the fruits is used. Then, the fruit monitoring system is applied to conduct the experiments of color recognition rate and tomato recognition. The results of the experiments show that this system has a high recognition rate. In section 2, the fruit monitoring system will be introduced. First, it is the explanation of the operating principle of this system, the temperature and humidity sensor, and the communication system. Then, the GUI program is discussed in section 2.2. And section 2.3 discusses the Hue saturation value (HSV) color mode, and then the ROI function and Canny Edge detection is used to crop the image and remove the background. Finally, the Multi-layer neural network algorithm is introduced to judge the color of the fruit, where the recognition result was obtained on the GUI directly. Section 3 discusses the experimental results and discussion, and the paper concludes in section 4.

2. Fruit monitoring system

The proposed fruit monitoring system, which measures 40 cm × 32 cm × 55 cm, is shown in Figure 1. The whole system is made up of a camera, the main body, the temperature and humidity sensors, the control section of the control sensor and communication sensor, and the communication section, which communicates with the users, the battery, and the motor.

2.1. Composition of the fruit monitoring system

The composition of the proposed fruit monitoring system is displayed in Figure 2. Using Bluetooth technology, users can directly control the ATmega128, the camera, the image processing model, neural network training, and view the judgment results by GUI program. The recognition of the statistics of the temperature and humidity by the temperature and humidity sensors is synchronous, real-time, and continuous. Using H-bridge circuit, the system can move front and back, and left and right, and, at the same time, take advantage of the lift and drop functions of the camera and the motor can make it move up and down and left and right. The lift and drop of the camera can be realized by the rise and fall principle. The left and right rotation can be realized using the servomotor HS-311.



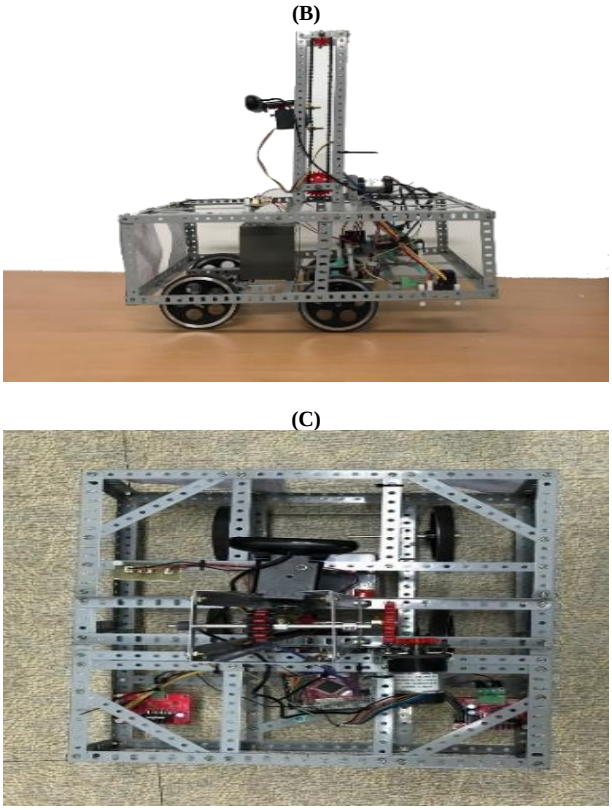


Fig. 1: Fruit Monitoring System. (A) Front View, (B) Side View, and (C) Top View.

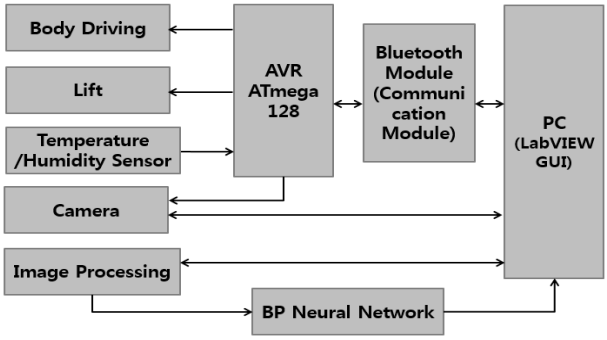


Fig. 2: Fruit Monitoring System Structure.

The communication between VDD and GND, MCU (Micro Controller Unit), and the sensor is synchronous. Using the temperature and humidity sensor measuring data, the statistics of the temperature and humidity can be transferred to the numerical value through A/D converter. According to Equations (1) and (2), the temperature and humidity value will be inputted as MCU, namely ATmega128 [20].

$$T = -40.00 + 0.01 \cdot SO_T \quad (1)$$

$$H = -0.4 + 0.0405 \cdot SO_{RH} + (-2.8 \cdot 10^{-6} \cdot SO_{RH}^2) \quad (2)$$

where, SOT is the voltage for temperature and SORH is the voltage for humidity.

In this study, the greatest communication distance of the Bluetooth for the experimental fruit monitoring system, which uses the FB155BC Bluetooth module and the FB200AS Bluetooth Receiver, is 15 m [21]. Assuming that the module is the SLAVE MODE and the receiver is the MASTER MODE, all of the temperature/humidity sensor, motor section, lift and drop section, and the camera's knob can be remotely controlled by the GUI program.

2.2. GUI of the fruit monitoring system

The PC GUI of the proposed fruit monitoring system is shown in Figure 3, and is achieved by the GUI program of LabVIEW [22]. The LabVIEW connects the multiple functions by wire. Compared to the other languages, the designed image based on the graphical programming language is more intuitional and the programming is faster. In addition, the LabVIEW can be displayed in various forms through the front panel [23]. The power control section of the PC GUI comprises the start and stop button of the program. The status display section uses an image-processing engine created using the OpenCV environment bank to output one of the two states of fruit maturity, namely GOOD or BAD. The control section uses different buttons to control the movement directions of the motor and the rotation, rise, and fall of the camera. The setting part is composed of the MCU, namely ATmega128, the corresponding Bluetooth receiver ports, the camera storage port, image-processing test data files, and the settings and temperature-setting window. The current temperature and humidity are displayed using the temperature and humidity sensors. The time display section shows the current date and time. The video output section shows the images photographed by the camera.

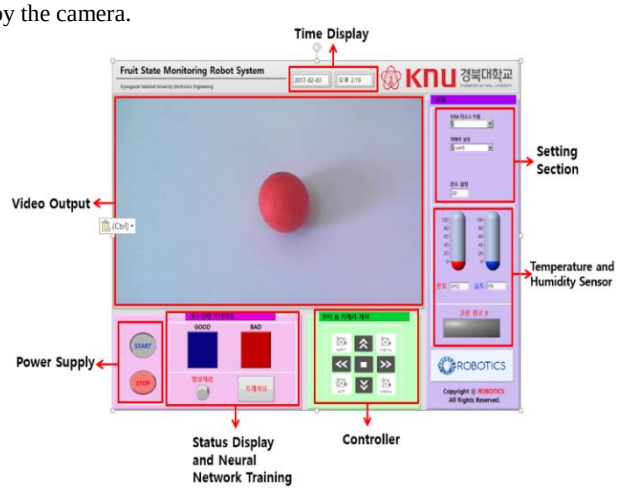


Fig. 3: GUI Screen in Lab View.

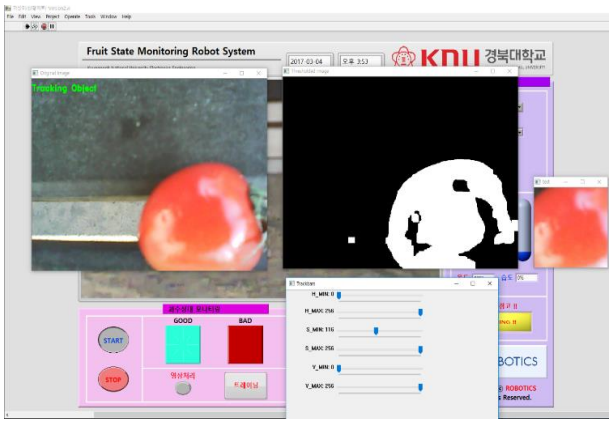


Fig. 4: Image Processing GUI Screen.

When running the program, the ATmega128 and the paired Bluetooth receiver, as well as the camera's port, are connected with the GUI. Pressing the "START" button will display real-time images. The control section, through the use of respective buttons, can change the control value of the ATmega128 motor and the camera, control their movements, and continuously display the current date, time, temperature, and humidity. If the measured temperature is higher than the value being set, the high-temperature warning LED will light up and, at the same time, the LED will display the word "WARNING". If we press the training button, the neural network will be trained (here we will save the training samples in the specified folder beforehand). Then, if we press the image-processing button as shown in Figure 4, the original image, Binary image, and HSV window will pop up. Finally, we can judge the maturity level of the fruits and show the result by turning on either the" good" or "bad" light in the GUI panel in real time.

2.3. Image processing using open CV

The process of Image processing using OpenCV is shown in Figure 5, which is comprised of the image processing output screen (which outputs the actual image), the HSV setting screen, the binary code pictures obtained using the setting value of HSV, and the OBJECT window (which removes the background and only shows the fruits).

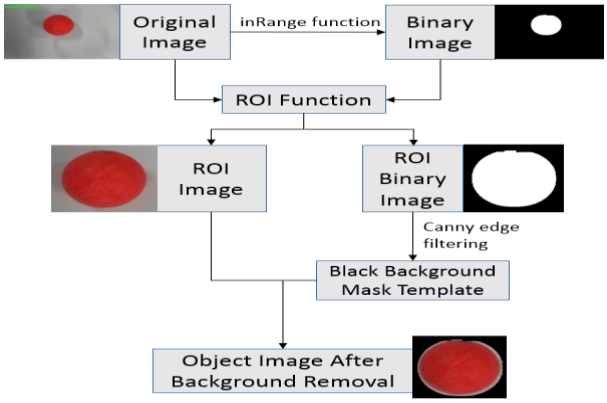


Fig. 5: Image Processing.

When the image processing program begins, the original image as well as the binary image based on the original image will be outputted, as shown in Figure 6(a) and Figure 6(b) (each channel values are set in the Trackbars window and the specified HSV values are inputted in the inRange function so that the binary image can be obtained). The S value is adjusted through the Trackbars window. The outline of the object will gradually appear in the binary image. At this point, by use of the moment function in OpenCV, it will determine the center coordinate of the object in the binary image. ROI (region of interest) function will use this center coordinate as the base to intercept a 200px × 200px region. Because the position of the center coordinate in the original image and the binary image is the same, in the original image, a region in the identical position and of the same size can also be cut out. Next, the object part in the ROI is removed by the Canny edge filtering which is an edge detection algorithm which can recognize the outline of the object in the binary image. And only the black background of the binary image is kept. Then the black background in the binary image will cover the original one. Finally, the ROI with its background removed is obtained as shown in Figure 6(d).

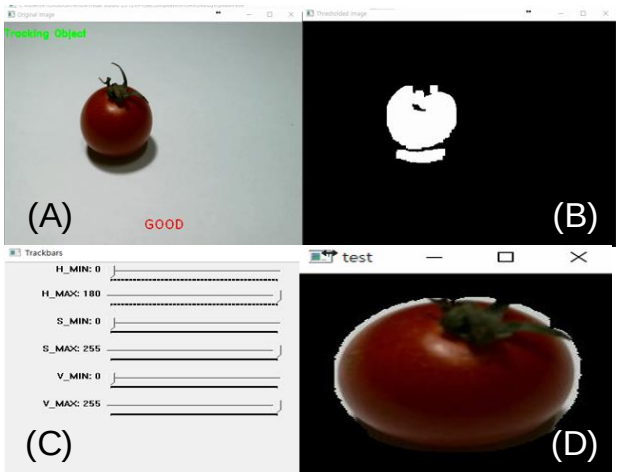


Fig. 6: Image Processing Screen (A) Image Output Windows, (B) Binary Image, (C) HSV Setting Windows, and (D) Object Windows.

2.3.1. Extracting the fruits area using HSV color space

HSV color space is a color model that is based on and is closest to the human eye observation. The model is comprised Hue, Saturation, and Value [24]. Hue is the visible fiber optic spectrometer values, which find their places in the color table, with the longest wavelength red as standard. 0° denotes the relative configuration angle, and the maximum range is up to 360°. Saturation indicates the degree of the hue's brightness, which ranges from 0% to 100%. Popularity (Value) represents the luminance of the object, which also ranges from 0% to 100%. The input image is obtained in the form of RGB, which is used to make the obtained image into HSV as shown in Equation (3).

$$H = \begin{cases} \theta & ,if\ \ B \le G \\ 360 - \theta, if\ \ B > G \end{cases}$$

$$S = 1 - \frac{3}{(R+G+B)}[\min(R+G+B)] \quad (3)$$

$$V = \frac{1}{3}(R+G+B)$$

The θ in Equation (3) is based on red color. The specific definition is shown in Equation (4). In Equation (3), if the change of hue is B ≤ G, then the range of hue is 0°-180°. If B > G, then the range of hue is 0°-360°. S and V are the value between 0 and 1 [25]

$$\theta = \cos^{-1}\{\frac{0.5[(R-G)+(R-B)]}{[(R-G)^2+(R-B)(R-G)]^{\frac{1}{2}}}\} \quad (4)$$

Using this method, we make use of the Trackbars of the adjustable HSV value to extract the right form of the fruits as shown in Figure 7. Each channel of the HSV has 1 byte. Therefore, we mark it as numbers 0- 255. The range 0°-360° is used to describe H by a value of maximum 1 byte. We should reduce the range in half to 0°-180°. S and V are marked within the range 0°-255°.

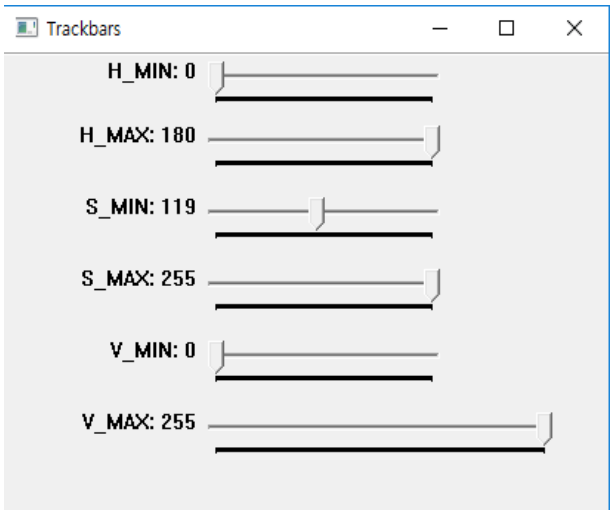


Fig. 7: HSV Screen.

2.3.2. Using ROI to cut the image

ROI is mainly applied to photos that only partly need to be processed. To obtain the ROI photos, we need to use the ROI function with a specified range. In addition, this function needs to use the central coordinates of the input red ball. To obtain the central coordinates of the red ball, we apply the moment function, which can detect the objects extracted from the binary code pictures. The moment function is shown in Equation (5).

$$M_{ji} = \sum_{x,y}[array(x,y) \cdot x^j \cdot y^i] \quad (5)$$

Where,

$$x = \frac{m_{10}}{m_{00}},\ y = \frac{m_{01}}{m_{00}}$$

where m_{00} is the length of the outline, and m10 and m01 divided by m_{00} presents the x, y coordinates, which are the center coordinates of the measurement area. Eventually, the center coordinates are used in the ROI function and the ROI can be extracted. If the center coordinates are used in the ROI function, regardless of which position, tracking can be carried out to extract the region of interest.

2.3.3. Removal of the background by canny edge filtering

Canny Edge filtering is the contour extraction algorithm proposed by John F. Canny in 1986, which is a representative contour detection algorithm [26]. The basic idea of the Canny algorithm is to find the position where the gray scale intensity changes the strongest. First, use Gaussian filtering to smooth the image. The purpose is to remove the noise. After that, combined with the Sobel operator, the gradient intensity and gradient direction of each pixel are calculated. Use the Non-maximum suppression technique to make the blurred boundary sharper. Next use double thresholding technology to set a threshold's upper limit and threshold's lower limit to determine the boundary further. Then we fill the area between the outline of the object and the boundary of ROI image to make the mask template with black background. After we cover the ROI Image by the mask template, and we get the object image without the background, as shown in Figure 8, we use Canny Edge filter to extract the contour of the image and obtain a fruit region.

To obtain the sample data, we continue to apply the ROI function to the background-removed images. Given that we have already confirmed the central point of this image, when we obtained the interest area of the image using the ROI function, we continue to use this central point. We apply the ROI function in the horizontal direction in the range of 8 × 50px and in the vertical direction in the range of 50 × 8px to obtain the interest area. Next, to read the data information from the images, we apply the for language to the 8 × 50px ROI area and the 50 × 8px ROI area. While the pixels in the ROI area are being read individually, when we apply the cvGetReal2D function, we can obtain the pixel data. Finally, the 400 data in the horizontal direction and the 400 data in the vertical direction will be saved in two TXT files to supply the use of neural network.

Fig. 8: Background Removal Using Canny.

2.4. Multi-layer neural network

The neural network is composed of a large number of simple processing units, and is a parallel distributed system connected by variable weight. A neuron is the basic processing unit of the artificial neural network, which is a multi-input and single output nonlinear device. The most famous neural network learning algorithm is the back-propagation (BP) learning algorithm for BP network error that is the gradient algorithm. This algorithm belongs to the tutor learning, whose principle is to modify the weight of the network from the gradient direction of the square of the error and the weight of

the network so that the BP neural network can achieve the desired learning effect quickly.

In 1986, Rumelhart, Hinton, and Williams completely and concisely put forward the error back-propagation training algorithm for artificial neural network (BP algorithm) [27]. It systematically solves the problem of learning in multilayer networks with unit weights. The algorithm forms a network called BP network, which is a kind of forward feedback network that is most widely used. The structure of the multilayered neural network (MNN) with feedforward connections is depicted in Figure 9.

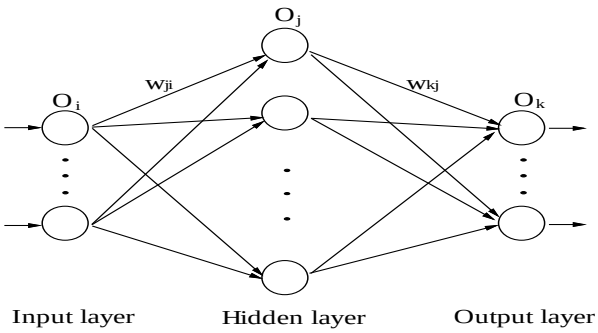


Fig. 9: Structure of Multilayered Neural Network.

The input output node relations of the MNN are described as follows: At the output node k,

$$O_k = f_k(net_k) \tag{6}$$

$$net_k = \sum_{j=1}^{J} w_{kj} O_j \tag{7}$$

Where O_j is the output of the hidden node j, $f(\cdot)$ is a sigmoid activation function, J is the number of the hidden nodes at the hidden node j

$$O_j = f_j(net_j) \tag{8}$$

$$net_j = \sum_{i=1}^{I} w_{ji} O_i \tag{9}$$

Where Oi is the output of the input node i, I is the number of the input nodes.

In the BP method, the weights of the neural network are adjusted by adding a value proportional to the negative gradient of a cost function. Let the cost function be defined as

$$E = \frac{1}{2} \sum_{k=1}^{N} (d_k(t) - O_k(t))^2 \tag{10}$$

Where d_k is the desired value, Ok is the output of the output node k, and N is the number of the output nodes. The subscripts i, j, and k refer to any node in the input, hidden, and output layers, respectively. The updated weights and the incremental change of output weights are computed as follows:

$$w_{kj}(t+1) = w_{kj}(t) + \Delta w_{kj} + \alpha(w_{kj}(t) - w_{kj}(t-1)) \tag{11}$$

Where

$$\Delta \text{ Wkj} = \eta \cdot \delta_k \cdot O_j \tag{12}$$

H is the learning rate, α is the momentum term to increase the learning speed, and w_{kj} is the weight from output node k to the hidden node j. The error signal δ_k at the output node k is obtained from the chain rule and defined below:

$$\delta_k = (d_k(t) - O_k(t)) \cdot \dot{f}_k(net_k) \tag{13}$$

Where $\dot{f}_k(\cdot)$ is the derivative of the sigmoid activation function.

Similarly, the weight-update equation of the hidden layer and the error signal is written as follows:

$$w_{ji}(t+1) = w_{ji}(t) + \Delta w_{ji} + \alpha(w_{ji}(t) - w_{ji}(t-1)) \tag{14}$$

Where,

$$\Delta w_{ji} = \eta \cdot \delta_j \cdot O_i \tag{15}$$

$$\delta_j = \sum_{k=1}^{N} \delta_k \cdot w_{kj} \cdot \dot{f}_j(net_j) \tag{16}$$

3. Experimental results and discussion

MATLAB is applied to construct the BP Neural Network as shown in Figure 10. The network is shown to have 800 input nodes, 10 hidden nodes, and 2 output nodes (which is good or bad). Figure 11 shows the test result of the neural network. The red line "O" is the actual output of the test sample. The blue line * describes the test output of the neural network. The result shows an accuracy rate of 100%.

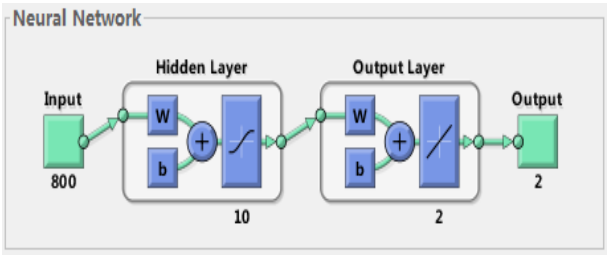
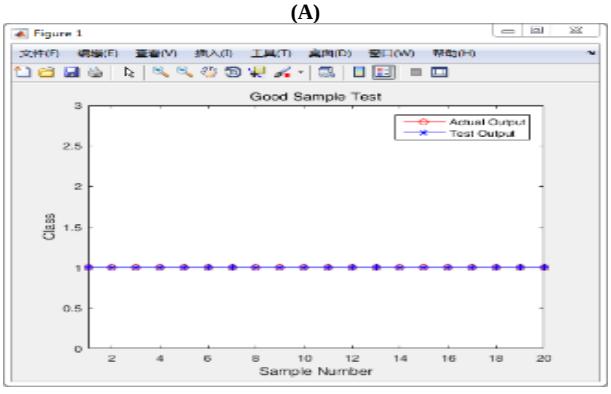


Fig. 10: BP Neural Network on MATLAB.

(A)



(B)

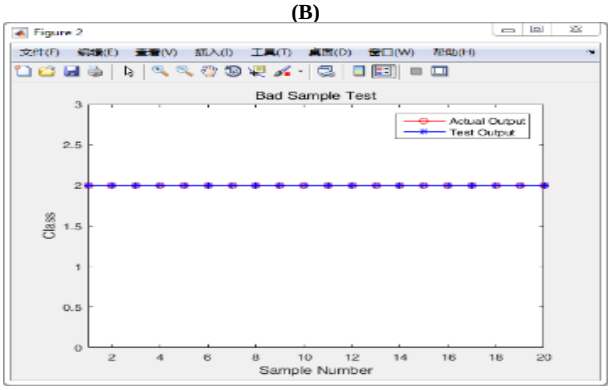


Fig. 11: Test Result of the Neural Network. (A) Good Sample Test and (B) Bad Sample Test.

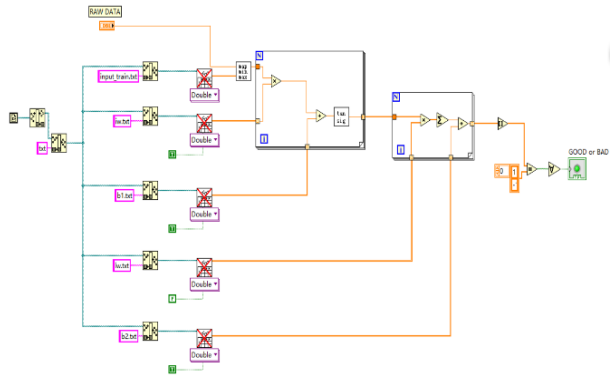
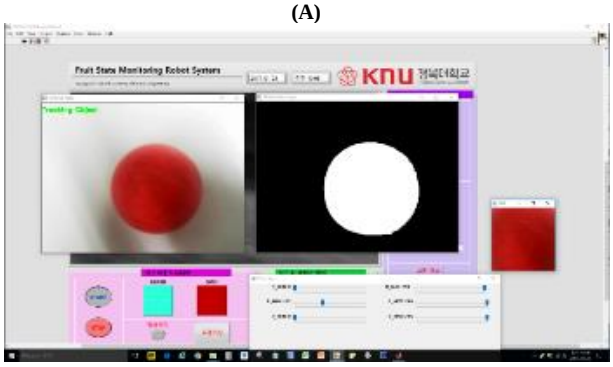


Fig. 12: BP Neural Network Program in GUI by Lab View.

(A)



(B)

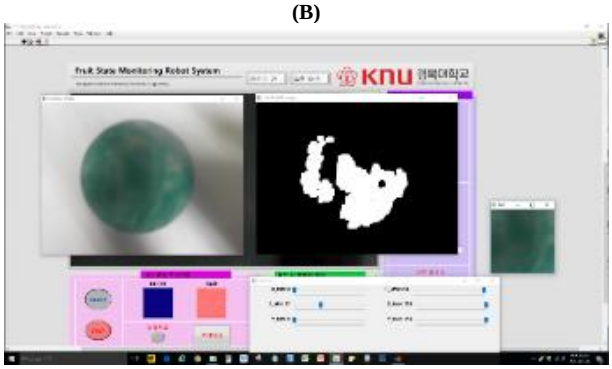


Fig. 13: Using the GUI Program to Judge the Red Ball and Green Ball in the Lab. (A) Recognition Result of the Red Ball and (B) Recognition Result of the Green Ball.

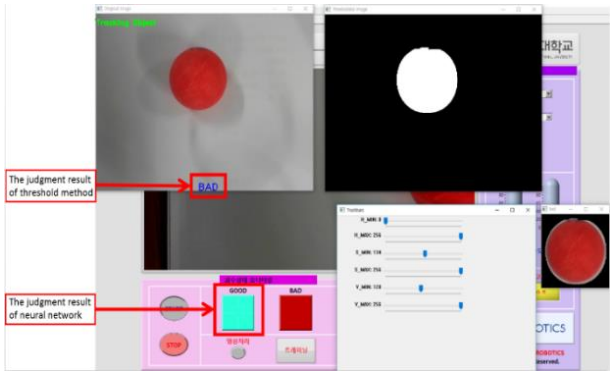


Fig. 14: Compare the Threshold Method's Result with Neural Networks.

Then, the neural network is placed in the GUI by LabVIEW as shown in Figure 12. A neural network training program on the GUI is also made to directly call the MATLAB program to train the neural network, which is more convenient and increases the practicality. Next, we have the image of the object obtained by the camera after ROI and Canny Edge filtering. Only the region of the object to be measured has been cut. After that, the GUI program is applied to assess the red and green balls as shown in Figure 13 in the lab. Here, it shows that when the ball is red, the GOOD light is turned on, and when the ball is green, the BAD light is turned on. Here 50 experiments have been down, 48 of which were successful, [2] of which failed, so the recognition rate was 96%. The experimental results have met the expectations.

In the experiment, the status of the ball is also judged by threshold method [28] using Equation (17).

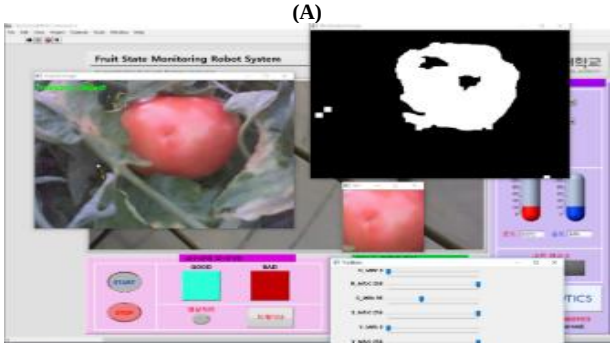
$$\frac{|red|}{|red|+|green|} \times 100 \geq d_T$$

$$\begin{cases} if \quad ture, & Output \quad "GOOD" \\ else, & Output \quad "BAD" \end{cases} \tag{17}$$

Where |red| and |green| are number of red pixels and green pixels, respectively. And d_T is threshold value for decision and it is determined by heuristic method.

Fig. 15: Experiments Conducted in the Greenhouse.

(A)



(B)

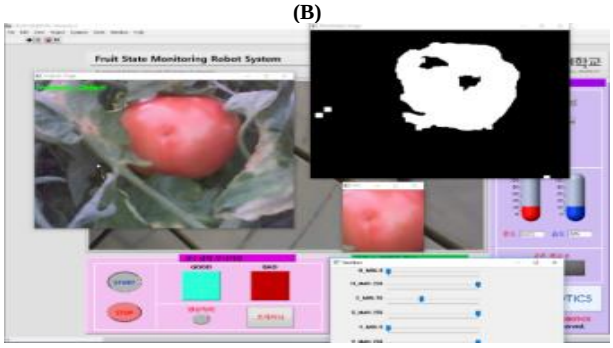


Fig. 16: Using the GUI Program to Judge the Red and Green Tomatoes in the Greenhouse. (A) Recognition Result of Red Tomato and (B) Recognition Result of Green Tomato.

Because of the sensitivity of the threshold to the environment, the value of the threshold d_T should be adjusted according to the environment. For example, as shown in Figure 14, here the real-time image is a red ball, then the correct judgment should be good. But

when the threshold d_T is 90%, the threshold method made a wrong (bad) judgement. However, the neural network made a correct (good) judgement at the same time. Thus, compared to the threshold method, the neural network has a wider applicability and better accuracy.

The experiments of this fruit monitoring system is conducted in the greenhouse as shown in Figure 15. Using the GUI program, the intuitively the real time temperature can be obtained, which is 21°C and the real time humidity, which is 23%. In addition, the experiments on real growing tomatoes is conducted. As shown in Figure 16, when red tomatoes are in the images, the recognition result is GOOD. However, when green tomatoes are in the images, the recognition result is BAD. Therefore, 10 experiments were conducted on the red and green tomatoes, respectively. As a result, the recognition rate of the red tomatoes is 100%, and recognition rate of the green tomatoes is 90%. The results met the expectations and show that the proposed fruit monitoring system has a very high recognition rate.

4. Conclusion

The age of farmers in the countryside is increasing annually, while the percentage of professional staff is decreasing because of the increase in the number of novices. Moreover, increasing number of people suffer from sunstroke due to the hot weather. So, a fruit monitoring system have been proposed by us to help the novices on the farm to determine when to harvest.

In this study, we utilized the image processing technology and multi-layer neural network to develop a fruit monitoring system. The GUI program was also designed which makes proposed system more intuitively and more conveniently. The GUI program was used to assess the red and green balls in the lab and the experimental results have met our expectations. Then the experiments on real growing tomatoes were conducted. The accuracy in the green house is 95%. Therefore, the results of the experiments show that this system has a high recognition rate.

In the future, WiFi is planned to add to this system so that the wireless transmission of the images can be utilized. This would greatly improve the practicability of the fruit monitoring system. We also hope to use a caterpillar band instead of wheels in order to overcome difficult topographic conditions. In addition, the main point of future research will focus on how to reduce the influence of the illumination intensity on the results of the experiments. For example, the size of the lens hoods can be increased, use the filter lens or improve the image processing algorithms.

Acknowledgement

This research was also supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. NRF-2016R1D1A3A03919627).

References

[1] Seok, H. J.; Lee, J. H.; Lee, S. H.; Lee, E. P.; Oh, B. H.; Rui X.;, Lee, I. S. Implementation of an Image Processing System Based on Mobile Robot for Fruit State Monitoring, Proceedings of the 2016 KIIT Summer Conference, Korea, 2016, pp.394-396. (In Korean).

[2] Yang, J. H.; Ha, J. S. Estimation of Future Death Burden of High Temperatures from Climate Change, Korean society of Environmental Health, 2013, 39(1), pp. 19-31. (In Korean).

[3] Morimoto, T. ; Takeuchi, T. ; Miyata, H. ; Hashimoto, Y. Pattern recognition of fruit shape based on the concept of chaos and neural networks, Computers and Electronics in Agriculture, 2000, 26(2), pp. 171-186. https://doi.org/10.1016/S0168-1699(00)00070-3.

[4] Nakano, K. Application of neural networks to the color grading of apples, Computers and Electronics in Agriculture, 1997, 18(2-3), pp. 105-116. https://doi.org/10.1016/S0168-1699(97)00023-9.

[5] Zhao, M.; Hou, W. Method of apple automatic grading based on neural network, Journal of Nanjing Forestry University (Natural Sciences Edition), 2009, 33(1), pp. 136-138.

[6] Ji, H.; Yuan, J. The application Study of Apple Color Grading by Particle Swarm Optimization Neural Networks, Intelligent Control and Automation, Proceedings in the Sixth World Congress, 2006, pp. 2651-2654.

[7] Hu, H.; Deng, J.; Zhang, T. Study on color Sorting for Apples Based on Hamming Neural Networks, Journal of Agricultural Mechanization Research, 2006, 11.

[8] Kim, U. C.; Lee, S. W.; Choi, Y. Y.; Kim, G. L.; Lee, Y. I. On the self-driving of a model car using image processing, Institute of Control, Robotics and Systems, 2011, 2011(5), pp. 660-663. (In Korean).

[9] Jeong, J. S.; Jang, Y. M.; Park, Park, E. C. H.; Cho, S. B. Vehicle License Plate Recognition System using a Noise-cancelling based on Square Feature and OpenCV, The Institute of Electronics Engineers of Korea, 2015, 2, pp. 784-786. (In Korean).

[10] Johnson, J.; Picton, P. Concepts in artificial intelligence, Butterworth-Heinemann LTD, 1995, Ch(4), pp. 95-96.

[11] Schurmann J, Pattern classification, a unified view of statistical and neural approaches, John Wiley and Sons, New York, 1996.

[12] Kohonen T, Self-organizing maps, Springer, Berlin, 1997. https://doi.org/10.1007/978-3-642-97966-8.

[13] Fausett L, Fundamental of neural netwroks, Prentice Hall, 1994.

[14] Timmermans, A. J. M. Hulzebosch, A.A. Computer vision system for on-line sorting of pot plants using an artificila neural network classifier, Computer and Electronics in Agriculture, 1996, 15(1), pp. 41-55. https://doi.org/10.1016/0168-1699(95)00056-9.

[15] Nakano, K. Application of neural networks to the color grading of apples, Computers and Electronics in Agriculture, 1997, 18(2-3), pp. 105–116. https://doi.org/10.1016/S0168-1699(97)00023-9.

[16] Schmoldt, D.L.; Li, P.; Abbott, A. L. Machine vision using artificial neural networks with local 3D neighbourhoods, Computer and Electronics in Agriculture, 1997, 16(3), pp. 255-271. https://doi.org/10.1016/S0168-1699(97)00002-1.

[17] Wang, D.; Dowell, F. E.; Lacey, R. E. Single wheat kernel color classification using neural networks, Transactions of the ASAE-American Society of Agricultural Engineers, 1999, 42(1), pp. 233-240. https://doi.org/10.13031/2013.13200.

[18] Yang, C.-C.; Prasher, S.O.; Lacroix, R.; Sreekanth, S.; Madani, A.; Masse, L. Artificial neural network model for subsurface-drained farmlands, Journal of Irrigation and Drainage Engineering, 1997, 123(4), pp. 285-292. https://doi.org/10.1061/(ASCE)0733-9437(1997)123:4(285).

[19] Yang, C.-C.; Pasher, S.O.; Mehuys, G. R.; An artificial neural network to estimate soil temperature, Canadian Journal of Soil Science, 1997, 77(3), pp. 421-429. https://doi.org/10.4141/S96-062.

[20] SHT-11 Data Sheet http://www.sensirion.co.kr/.

[21] Oh, B. H.; Kim, T. H.; Lee, I. S. Implementation of Safety Buoy System Using PIR Sensors, Journal of Korean Institute of Information Technology, 2015, 13(4), pp. 9-15. (In Korean). https://doi.org/10.14801/jkiit.2015.13.4.9.

[22] LEE, I. S. Diagnostic system development for state monitoring of induction motor and oil level in press process system, Korean Institute of Intelligent Systems, 2002, 19(5), pp. 706-712. (In Korean).

[23] Lee, K. H.; Kim, I. J.; Choi, J. Y.; Lee, S. K. Design of Real-Time Power System Simulator for Education using LabVIEW, The Korean Institute of Illuminating and electrical Installation Engineers, 2012, 24(6), pp. 177-182. (In Korean).

[24] Park, H. S. Vehicle Tracking System using HSV Color Space at nighttime, Korea Information Electron Communication Technology, 2015, 8(4), pp. 270-274. (In Korean). https://doi.org/10.17661/jkiiect.2015.8.4.270.

[25] Kim, J. H.; Jeon, S. H.; Park, K. H.; Kim, Y. H. The Method of the Identity-Discriminant using HSV Color Model for Moving Object Tracking, Korean Institute of Information Technology, 2014, 5, pp. 292-295. (In Korean).

[26] Choi, J. H.; Ko, K. J.; Han, D. I. Hardware Design and Implementation of Advanced Edge-Detection engine based on Canny Edge Filter for Robustness to Quantization Noise, The HCI Society of Korea, 2011, 11(1), pp. 582-584. (In Korean).

[27] Rumelhart, D. E. McClelland, J. L. Parallel Distributed Processing: Explorations in the Microstructure of Cognition, 1986, 1, MIT Press, Reading, MA.

[28] Xu, R.; Seok, H. J.; Lee, J. H.; Lee, S. H.; Lee, E. P.; Cho, T. H.; Lee, I. S. The Mobile Fruit Monitoring System Using Image Processing Technology and Temperature/Humidity Sensors, Korean Institute of Information Technology, 2017, 15(5), pp. 37-45. (In Korean). https://doi.org/10.14801/jkiit.2017.15.5.37.