Website : jurnal.umj.ac.id/index.php/icecream

Implementation of Digital Image Processing for Raspberry Pi-Based Warehouse Layout Settings

Deska Anas[1], Fadliondi [1*]

[1]*Department of Electrical Engineering, Universitas Muhammadiyah Jakarta, Indonesia*

[*]*Corresponding author: fadliondi@umj.ac.id*

ABSTRACT

Warehousing is a place for storing goods that are formed in such a way. Efficient and effective warehousing has the ability to adapt to demands to increase the speed of processes starting from receiving, storing and shipping. Good conditions and arrangements in the warehouse are expected to avoid company losses, reduce costs incurred and speed up activities and services at the warehouse. Therefore, we need an application that can support warehousing. In this case, we analyzed the warehousing system to support the warehousing process and designed a "Digital Image Processing Implementation for Raspberry Pi-Based Warehouse Layout Settings". This design uses image processing. At this stage, the camera reads the object and the object image is processed by the microcontroller which will then be arranged according to the shape of the object. The objects are then separated using a servo motor according to a predetermined path so that the objects can be arranged according to size and not mixed up.

Keywords: digital image processing, raspberry pi, threshold, warehousing.

1. Introduction

Warehousing activities must have good storage in order to support the smooth process in warehousing. A warehouse is said to be effective and efficient if the storage and release of goods from and to the warehouse is smooth. Warehousing is all management efforts that include receipt, storage, maintenance, distribution, control, destruction and reporting of logistics and logistics equipment so that quality and quantity are guaranteed. Warehousing activities include:

1. Maintaining the smooth receipt and expenditure of logistics.
2. Maintaining order in warehousing administration to ensure the safety of goods as well as the provision of accountability tools for warehousing management.
3. Proper storage of logistics so that logistics are easy to check, find and retrieve.
4. Proper arrangement of goods so that the safety and security of goods is guaranteed.
5. Good care of goods so that the goods in the warehouse are not only as inventory items but also goods that are ready to be used.

Image processing is an activity to process the signal with the input of image and transform into another image with certain techniques [1]. Image processing is a set of methods to get high quality image or to obtain some useful information from it [2]. A warehouse is a place for storing items where inventories are used to fulfill future needs and requirements [3], [4]. Raspberry Pi is one of the most well-known electronic boards used for prototyping the applications such as industry, research, agriculture, etc [5].

2. Methodology

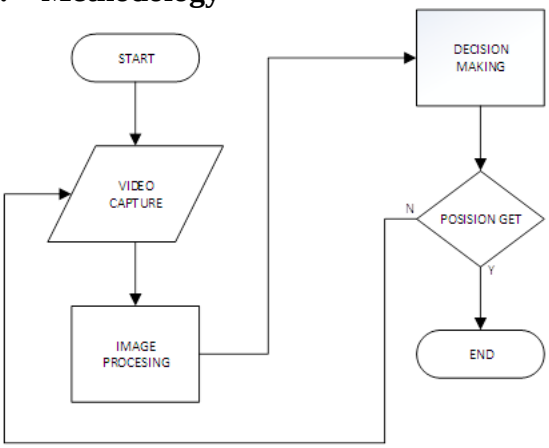

Figure 1. System Flow Diagram

In this flow chart, we can see how the tool works. First the camera is activated. The image is taken from the camera and processed in the image processing block. The next stage is decision making.

At the time of capture, if the object matches the image, the tool will direct the object to the appropriate place. If the system has been running but can't find a suitable place, then the system will loop until it finds a place that matches the object.

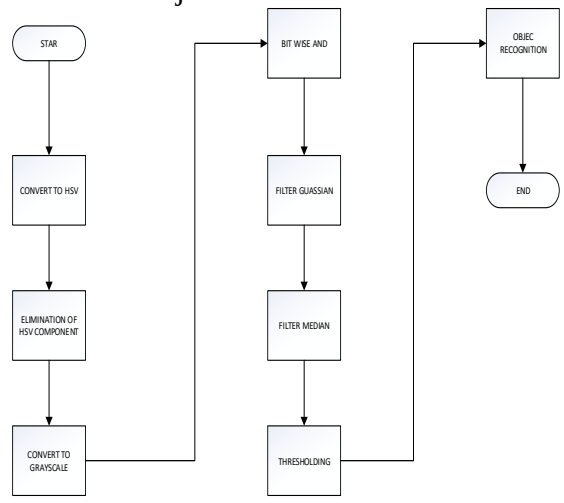

Figure 2. Image Processing Flow Chart

This flow chart shows how image processing works. When the image has been taken, the image will be processed in the rotate image block. After that, the image is converted to an HSV format image so that the obtained image can be processed based on the Hue value so that it is possible to process the saturation value and value in a wide range. HSV is a technique to represent points in the RGB color space in an inverted cone [6]. Hue Saturation Value is commonly used as for the image detection it is same as what natural eye observes compared to a RGB picture [7]. Furthermore, the HSV image will be eliminated and each component separated, namely hue, saturation and value. Meanwhile, the HSV image will be limited to a certain hue value so that it will only produce the specified color and ignore other colors. After getting a limited HSV format image and a grayscale format image. The image will be combined with a bitwise operator process so that the result is a grayscale image by ignoring the color in the hsv image other than the predetermined hue value. To reduce the noise that occurs, the image will be filtered using two filters, namely gaussian and median. Gaussian filter is a method to filter the image before the classification process. This technique is a linear filter with a weighted value and is selected based on the form of the Gaussian function [8].

Median filter focuses on the middle value of the total number of overall values of pixels around it and serves to smooth the image to reduce noise or interference with the image [9].

After the image is filtered, the image will enter the development process using the Otsu method with the final result in the form of a binary image, each pixel is worth 1 bit. The Otsu method is one of the efficient techniques for image thresholding using image histograms that can be used to separate the background pixels of the image [10]. After the image is in binary format, the object in the image will be identified with an object recognition function that will identify the appropriate shape of the object.

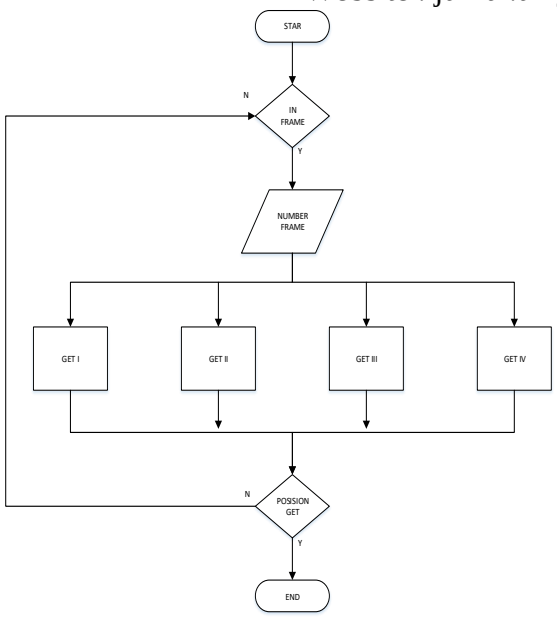

Figure 3. Decision Making Flow Chart

This flow chart shows how the decision-making system works based on the results of the previous image processing. When the size of the target taken is in accordance with the position of one of the paths, a decision will be made by adjusting the position of the target so that the system directs the object according to the appropriate path. Servo motor is the vital part of automatic production line and its performance and reliability have great role on the normal operation of the whole automatic production line [11].

3. Results and Discussion

Tests are carried out on both software and hardware. Then automatic testing is carried out to obtain appropriate results and determine the validity of this system. This test is carried out with the aim of obtaining test results whether it is in accordance with what is expected. The testing includes software testing and hardware testing. In software testing, detection of boxes based on color and size is carried out. Then, the calculation of the number of boxes that have been detected will be carried out. In the object detection test, there are box A, box B, box C and box D. Each box will be tested 10 times. The test results can be seen in Table 1, Table 2, Table 3 and Table 4.

Table 1. Test Results Of Box A

Test	Radius (pixel)	Description
1	319	Detected
2	318	Detected
3	315	Detected
4	305	Detected
5	307	Detected
6	301	Detected
7	320	Detected
8	308	Detected
9	301	Detected
10	322	Detected
Average radius = 311.6 pixels		

Table 2. Test Results Of Box B

Test	Radius (pixel)	Description
1	245	Detected
2	262	Detected
3	266	Detected
4	226	Detected
5	297	box A detected
6	251	Detected
7	285	Detected
8	238	Detected
9	277	Detected
10	240	Detected
Average radius = 258.7 Pixels		

Table 3. Test Results Of Box C

Test	Radius (pixel)	Description
1	188	Detected
2	188	Detected
3	204	Detected
4	186	Detected
5	207	Detected
6	177	Detected
7	154	Box D

		detected
8	217	Detected
9	200	Detected
10	219	Detected
Average radius = 194 Pixels		

Table 4. Test Results Of Box D

Test	Radius (pixel)	Description
1	128	Detected
2	143	Detected
3	158	Detected
4	155	Detected
5	147	Detected
6	153	Detected
7	148	Detected
8	132	Detected
9	157	Detected
10	135	Detected
Average radius = 145.6 pixels		

In testing the number of objects detected, there are 3 boxes. The test results can be seen in Table 5.

Table 5. The Results Of Testing The Number Of Objects Detected

Test	Box detected	Number of boxA	Number of box B	Number of box C	Number of box D
1	C	0	0	1	0
2	A	1	0	1	0
3	D	1	0	1	1
4	B	1	1	1	1
5	B	1	2	1	1
6	C	1	2	2	1
7	A	2	2	2	1
8	A	3	2	2	1
9	B	3	3	2	1
10	D	3	3	2	2
11	D	3	3	2	3
12	C	3	3	3	3

Based on the decision-making flow diagram in Figure 3, the movement of the gate making up the box layout will be determined based on the results of object detection. When the detected object is box A, gates B, C and D will open. If the detected object is box B, gate B will be closed and gates C and D will be open. If the detected object is box C, then gate B will not move or follow the previous command and gate C will close and gate D will open. If the detected object is box D, then gates B and C will not move or follow the previous command and gate D will be closed. Clearer results can be seen in Table 6.

Table 6. Gate Movement Test Results Based On Detected Objects

Detected box	Gate B	Gate C	Gate D
Box A	Open	Open	Open
Box B	Close	Open	Open
Box C	Last command	Close	Open
Box D	Last command	Last command	Close

Figure 4. Gate Testing Based On Detected Objects

This system uses a Raspbery Pi as a microcomputer to input images and process digital image processing and uses Arduino nano as a microcontroller to control actuators based on the results of digital image processing. Communication between Raspberry Pi and Arduino uses serial communication in the form of TX (Transmitter) and RX (Receiver) pins. The data sent by Raspberry PI is in the form of byte data type and will be converted by Arduino to char data type which can be seen in Table 7.

Table 7. Test Results, Sending And Receiving Serial Data Between Raspberry Pi And Arduino Nano

Detected box	Raspberry Pi	Arduino nano
A	byte(1)	char(1)
B	byte(2)	char(2)
C	byte(3)	char(3)
D	byte(4)	char(4)

From the test results with different box sizes, the digital image processing process has been successful. From a total of 12 tests, all of them were successful and the number of boxes that had been detected was counted well.

4. Conclusion

Based on the test results, it can be concluded as follows:

1. The radius value on box A is between 297 and 322 pixels, on box B it is between 285 and 226 pixels and on box C it is between 219 and 177 pixels, and on box D it is between 128 and 154 pixels.
2. Failure appears in the detection of box B on the 5th test with a radius value of 297 pixels so that the system detects it as box A. In addition, failure appears in the detection of box C on the 7th test with a radius value of 128 pixels so that the system detects it as box D .
3. From the analysis that has been done, the failure occurred because the position of the box at the time of detection was too tilted.
4. The quality of pixels and the amount of FPS in the test has decreased due to the Raspberry PI microcomputer hardware factor so that the microcomputer hardware adjusts the pixel quality and the amount of FPS according to its own capabilities.
5. The process of sending serial data from the Raspberry PI in the form of a byte data type to the Arduino in the form of a data type that has been converted into a char data type has been successfully sent and received without any disturbance.

References

[1] H. Pranamurti, A. Murti, and C. Setianingsih, “Fire detection use CCTV with image processing-based Raspberry Pi,” in Journal of Physics: Conference Series, 2019, vol. 1201, no. 1, p. 012015.

[2] K. V. Deepak and K. Sasikala, “A Brief Review on Various Image Segmentation Techniques,” J. Homepage Www Ijrpr Com ISSN, vol. 2582, p. 7421.

[3] N. Azadi Zadeh, “Subject: Evaluation of warehouse management and warehouse location and security, in offices,” Geogr. Hum. Relatsh., vol. 3, no. 1, pp. 42–75, 2020.

[4] A. Novyrmansyah, F. Esra Muhammad, H. Faisal, S. Kurnia, and V. Hartati, “Products Classification in the Finished Good Warehouse (Case Study of Pharmacy Industry in Bandung),” Solid State Technol., vol. 63, no. 3, pp. 5321–5332, 2020.

[5] S. E. Mathe, M. Bandaru, H. K. Kondaveeti, S. Vappangi, and G. S. Rao, “A survey of agriculture applications utilizing raspberry pi,” in 2022 International Conference on Innovative Trends in Information Technology (ICITIIT), 2022, pp. 1–7.

[6] J. Quan, H. Jin, Z. Li, and Z. Wen, “Low Illumination Image Enhancement Algorithm

Based on HSV-RNET," in 2022 7th International Conference on Image, Vision and Computing (ICIVC), 2022, pp. 531–536.

[7] M. P. Reddy, M. F. Mohiuddin, S. Budde, G. Jayanth, C. R. Prasad, and S. Yalabaka, "A Deep Learning Model for Traffic Sign Detection and Recognition using Convolution Neural Network," in 2022 2nd International Conference on Intelligent Technologies (CONIT), 2022, pp. 1–5.

[8] M. R. A. Yudianto and H. Al Fatta, "The effect of Gaussian filter and data preprocessing on the classification of Punakawan puppet images with the convolutional neural network algorithm.," Int. J. Electr. Comput. Eng. 2088-8708, vol. 12, no. 4, 2022.

[9] S. Sundaramurthy, A. Wahi, L. P. Devi, and S. Yamuna, "Cardiac cycle phase detection in echocardiography images using ANN," in 2014 International Conference on Intelligent Computing Applications, 2014, pp. 275–279.

[10] A. T. H. Al-Rahlawee and J. Rahebi, "Multilevel thresholding of images with improved Otsu thresholding by black widow optimization algorithm," Multimed. Tools Appl., vol. 80, no. 18, pp. 28217–28243, 2021.

[11] J. Huang, L. Qi, J. Gu, Z. Lu, J. Sun, and C. Yu, "Servo Motor Fault Diagnosis Based on Data Fusion," in 2021 33rd Chinese Control and Decision Conference (CCDC), 2021, pp. 6737–6743.