

## OBJECT DETECTION WITH A WEBCAM USING THE PYTHON PROGRAMMING LANGUAGE

Ali Basrah Pulungan<sup>1\*</sup>, Zhfranul Nafis<sup>2</sup>, Muhammad Anwar<sup>3</sup>, Hastuti<sup>4</sup>, Hamdani<sup>5</sup>, Dwiprima Elvanny Myori<sup>6</sup>

Department of Electrical Engineering, Universitas Negeri Padang<sup>1,2,4,5,6</sup>

Department of Electronic Engineering, Universitas Negeri Padang<sup>3</sup>

[alibp@ft.unp.ac.id](mailto:alibp@ft.unp.ac.id)<sup>1</sup>

Received : 07 May 2021, Revised: 27 May 2021 , Accepted : 27 May 2021

\*Corresponding Author

### ABSTRACT

Technology developed rapidly along the times, various ways are done to make works easier. One of them is by utilizing artificial intelligence, likes the use of a webcam as a sensor in detecting an object through several stages of image processing. There was research on object recognition previously using three measurement parameters based on color, shape, and size of objects. It is used a webcam as the sensing sensor, and image processing is processed with python programming. In this article, writer described about image processing for the detection of objects used in the research. The results of this device have been tested and are able to detect objects properly based on predetermined color, shape and size. Object detection using a webcam can work properly according to what the author wants.

**Keywords :** Webcam, Python, Image Processing, Automation.

### 1. Introduction

Technology is developing so rapidly along the times, so various ways are done to make it easier for humans to do a task. One of which is by utilizing artificial intelligence (Prayitno & Harianto, 2011; Mubarrok & Rahmawati, 2020). Many people are competing to make new innovations to be able to create tools that can make the task easier. One of the innovations in the newest technology is the use of a webcam. Currently, webcams have been widely used in industry; ranging from quality control, sorting, security (Juliando & Husin, 2019). Moreover, it can also be developed as a sensor in detecting an object through several stages of image processing (Prayitno & Harianto, 2011).

The stage of image processing is also called object detection. Object detection is the process of recognizing objects in a digital image. The image is obtained from a camera with certain specifications which processed by the object detection method, so that the classification of the object can be recognized (Najmurrohman et al., 2017). In several previous studies, the detection of this object only used two parameters for its recognition, namely shape-color recognition and color-size recognition. These two object recognition parameters are worth already. If object recognition used three recognition parameters, i.e. color, shape, and size; then the object sorting process is more selective accordingly (Purwanti, et. al., 2021).

The previous research used three measurement parameters that is already available, i.e. measurement parameters based on color, shape, and size of objects using a webcam as a sensing sensor, and the image processing is processed with python programming (Pulungan & Nafis, 2021). Thus, the author will account about image processing for the detection of objects used in research with three object recognition parameters using a webcam as a sensing sensor, and image processing is processed with the Python programming language.

### 2. Literature Review

#### 2.1. The Digital Image

A digital image is a two-dimensional image that can be displayed on a computer monitor screen as a finite (discrete) set of digital values called pixels (picture elements). The pixel is an image element that has a value that indicates the intensity of the color (Norfazlinda Binti Daud,

2007). The image processing is the process for processing an image quality or image that has been taken by a camera or webcam so that the image can be recognized and has certain values. The values that have been obtained are then processed to classify the images(Alfaro-Almagro, et. al., 2018).

The digital image is composed of a digital number that represents the intensity of light represented by a 2-dimensional array at a certain point in a grid. The points in this 2-dimensional grid are called pixels or mops, which stands for the word picture element.

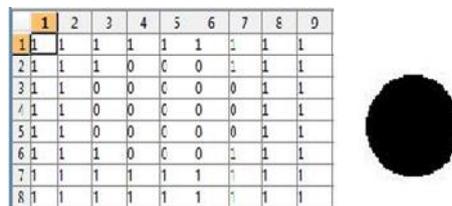


Fig. 1. Pixels in an Image

**2.2. HSV (Hue, Saturation, and Value )**

The image processing for color recognition is RGB to HSV color segmentation. HSV is a color model which derived from RGB color, so to get HSV color which have to do a color conversion process from RGB to HSV (Fauzi and Mayasari, 2011). HSV stands for Hue Saturation and Value. Hue is a measure of the wavelength of the color, hue has a size ranging from 0 until 255. The number of 0 represents the red color until through a spectrum it returns 256 or returns to red again. The sturation is a process to increase the brightness of the color based on the amount of pure hue in the final color. The value is a measure of how much brightness a color is.

If the color has a size of 100%, then it will look very bright. If it is 0%, then it will look dark (Aditya *et al.*, 2020). Thus, the process will form an area segment with the desired color tolerance (Zarwani, 2019). A literature review is a survey of scholarly sources (such as books, journal articles, and theses) on a particular topic. It gives an overview of key findings, concepts and developments in relation to a research problem or question.

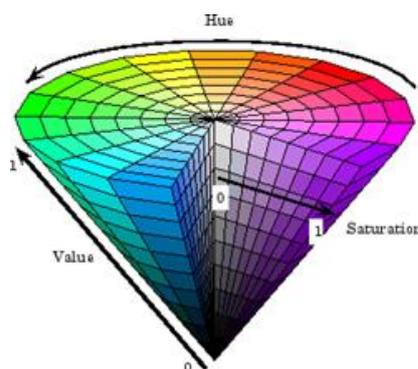


Fig. 2. HSV Color Space

**2.3. Thresholding**

Thresholding is the conversion of a color image to a binary image which is done by grouping the grayscale value of each pixel into 2 classes, black and white. In the floating operation, the pixel intensity value is mapped to one of two values,  $\alpha_1$  or  $\alpha_2$  (Handoyo, 2006). Contour is a state caused by changes in intensity at different pixels. Because of this change in intensity, the edges of the object in the image are detected (Andrizal *et al.*, 2016).

**2.4. Python**

Python is a multipurpose interpretive programming language that is easy to read and understand, and it places more emphasis on code readability to make it easier to understand the syntax. The modest and implemented code of python make it easy to learn, both for beginners and those who have mastered the programming language. Various operating system platforms

and software development requirements can use Python. Python programming language is defined as a language that combines capabilities, which with a very clear code syntax and also comes with a large comprehensive standard library functionality. Python is a free tool, there are no restrictions on how to distribute it. Equipped with automatic memory management, python is a dynamic programming language. Python is a programming language that is easy to understand because of the clear form of the language used (Aviolita, 2019).

### 2.5. OpenCV

OpenCV (Open Source Computer Vision Library) is a software that aim to process dynamic images in real-time (Januar, 2016). OpenCV is an API (Application Programming Interface) library which is very familiar in image processing designed for real-time applications, has good acquisition functions for image / video (M. Syafriandi, 2017). OpenCV has many functionalities that can help in making image processing applications easier and less complicated (Edward, Ghanie and Setiawan, 2020). OpenCV can be used with various programming languages such as Python, Java and C language (Bradski and Kaeler, 2008).

### 2.6. Webcam

This study used the Night Hawk a80 webcam with HD960p resolution. The functions is as a sensing sensor for image capture of objects to be detected in color, size and shape. The output of the data image obtained will be carried out by the object detection process using the PYTHON programming language OpenCV library.



Fig. 3. Webcam

## 3. Research Methods

### 3.1. Blok Diagram



Fig. 4. Diagram block

The function of each block diagram as follow:

- Objects are objects that will detect color, shape, and size.
- Webcam functions as a sensing sensor that will take pictures of objects.
- Python is a programming language that has been installed on a PC which functions to proceed image data on objects by detecting objects based on the object's color, shape, and size. The nomenclature must be appeared at the end of the paper (if included). The SI units are highly recommended for this journal.

### 3.2. Procedure

The first process is color recognition of object through the HSV segmentation process, where the color range processed is red. If the object's color is in the red range, then it will be detected. Thus proceed with the thresholding process, which is a segmentation technique to change the RGB color to the bine color. After that, the process is carried out to find out the outline of the object by drawing the contour to detect the edges of the object in the image. The last process is to describe the mid-point on each contour line to find out the size of the object. After the color,

shape and size of the object are known, it can be concluded whether the object conforms to the classification or not. The result of this detection will be in the form of a logic 1 or 0 which will be sent to the actuator controller via a USB connection. The range of colors detected by this tool is objects in red. If the object is red, then it will be processed for shape and size detection. Conversely, if the object is not red, then the object will be considered a reject without detection of its shape and size.

**3.3. Flowchart System**

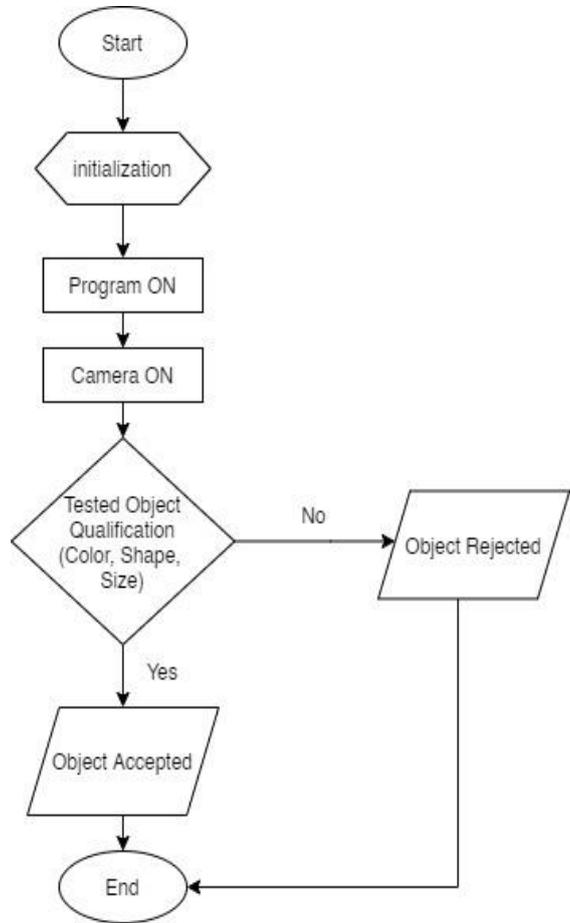


Fig. 4. Flowchart

**3.4. Design of Hardware**

The device is designed with an overall length and width of 20 x 20 x 30 cm, with the webcam stand at a height of 22 cm.

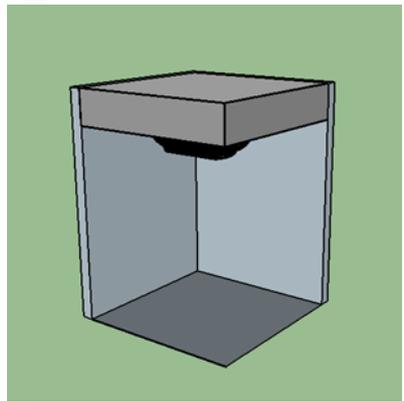


Fig 5. The device view

The device is tested using 8 different objects. The classification of objects to be tested can be seen in Table 1.

Table 1 - The classification of Object to be Tested

| No | Object   | Color | Shape          | Size (cm) | Result   |
|----|----------|-------|----------------|-----------|----------|
| 1  | Object1  | Red   | Square (edge)  | 6         | Accepted |
| 2  | Object 2 | Blue  | Square (edge)  | 6         | Reject   |
| 3  | Object3  | Red   | Square (edge)  | 7         | Reject   |
| 4  | Object 4 | Blue  | Square (edge)  | 7         | Reject   |
| 5  | Object 5 | Red   | Cirle (radius) | 6         | Reject   |
| 6  | Object 6 | Blue  | Cirle (radius) | 6         | Reject   |
| 7  | Object7  | Red   | Cirle (radius) | 7         | Reject   |
| 8  | Object 8 | Blue  | Cirle (radius) | 7         | Reject   |

### 3.5. GUI Display

Graphic User Interface (GUI) or graphical user interface is a display system of a program or application that functions to provide information or access to users. In this study also uses a GUI which serves as a display of information from the detection of objects that are being carried out.

There are several parts of the GUI in this article:

- The start button functions to activate the camera
- Stop button functions to turn off the camera
- Track button functions to start object detection
- Object display window which functions to provide information on object detection.
- Display binary image of detected objects in red.

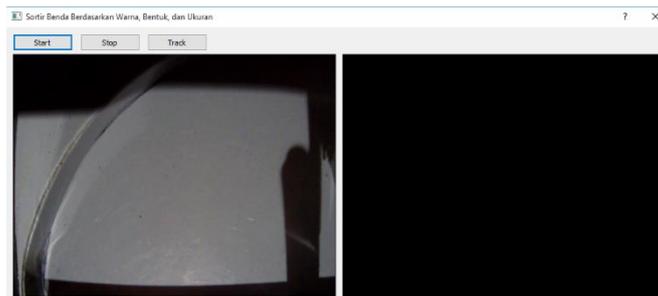


Fig 6. GUI Display

## 4. Results and Discussions

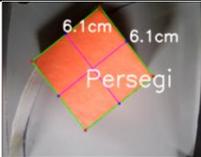
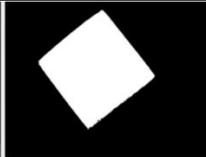
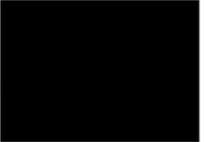
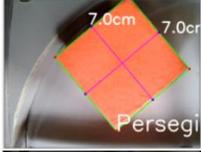
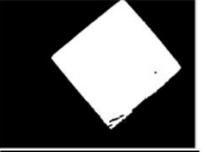
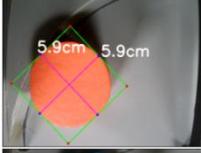
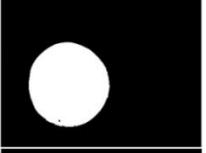
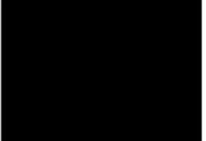
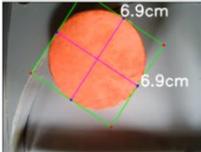
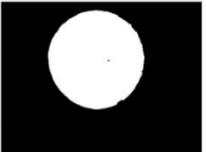
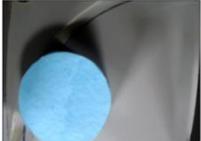
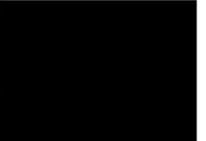
The following are some tests that have been carried out several times to obtain data and evidence that can be analyzed on the performance process of the device. The test of device using 8 objects of different colors, shapes, and sizes. The classification of the objects to be tested can be seen in Table 1.

### 4.1 Object Detection

This object detection is the result of object detection by the webcam using the python programming language with the OpenCV library. Testing was carried out with 8 different objects, where this detection includes detecting the color, shape, and size of the object. The results obtained are in Table 2 below

Table 2 - Object Detection Result

| No | Object | Object Detection Result |
|----|--------|-------------------------|
|----|--------|-------------------------|

|   |          |   |  |
|---|----------|---|--|
| 1 | Object 1 |    |    |
| 2 | Object 2 |    |    |
| 3 | Object 3 |    |    |
| 4 | Object 4 |    |    |
| 5 | Object 5 |    |    |
| 6 | Object 6 |   |   |
| 7 | Object 7 |  |  |
| 8 | Object 8 |  |  |

From the results of the detection of objects, data can be taken from each object, which will be grouped based on the detection data of color, shape and size in Table 3, Table 4, and Table 5.

#### 4.2. Color Detection

The color detection in this article is based on a predetermined range of HSV values, while the specified HSV ranges can be seen in Table 3.

Table 3 - Value range of HSV

| No | Limit Value | H  | S   | V   |
|----|-------------|----|-----|-----|
| 1  | Upper       | 30 | 255 | 255 |
| 2  | Lower       | 0  | 80  | 135 |

If the color of an object has an HSV value in the range of values in the table above, then the object will be detected, and will carry out the next process for shape and size recognition. With the listing program:

```
hsv = cv2.cvtColor(self.image, cv2.COLOR_BGR2HSV)
mask = cv2.inRange(hsv, lower_red1, upper_red1)
```

Fig.7 Listing program

The results of color detection on this device using the HSV color segmentation process can be seen in Table 4.

Table 4 - Color Detection Result

| No | Experiment | Result     |
|----|------------|------------|
| 1  | Object 1   | Detected   |
| 2  | Object 2   | Undetected |
| 3  | Object 3   | Detected   |
| 4  | Object 4   | Undetected |
| 5  | Object 5   | Detected   |
| 6  | Object 6   | Undetected |
| 7  | Object 7   | Detected   |
| 8  | Object 8   | Undetected |

This color recognition works as expected, because this device was able to detect red objects. It can be seen in the object classification table. Red objects are objects 1,3,5,7 these objects are detected and can be processed to recognize the shape and size of objects .

**4.3. Shape Detection**

After color detection, then the shape recognition processed. In this system device can only recognize square objects, with the following listing program.

```
if len(approx) == 4:
    cv2.putText(img, "Persegi", (x, y),
```

Fig. 8. Listing program

The results of shape detection are in Table 5 below.

Table 5 - Results of Shape Detection

| No | Object   | Shape  |
|----|----------|--------|
| 1  | Object 1 | Square |
| 2  | Object 2 | -      |
| 3  | Object 3 | Square |
| 4  | Object 4 | -      |
| 5  | Object 5 | -      |
| 6  | Object 6 | -      |
| 7  | Object 7 | -      |
| 8  | Object 8 | -      |

The recognition of object shapes can work as expected. The objects that can carry out the next process are only red objects, so objects detected by squares are only objects 1 and 3.

**4.4. Size Detection**

In the process of object detection, the last process is the size of the object. After the color and shape known, the next step is to test the size of the object. Objects that comply with the classification are objects with sides of 6x6cm with a tolerance of 0.2cm. This detection uses listing program like the one below.

```
dA = dist.euclidean((tltrX, tltrY), (blbrX, blbrY))
dB = dist.euclidean((tlblX, tlblY), (trbrX, trbrY))
```

Fig. 9. Listing program

The results of size detection are as shown in Table 6.

| No | Experiment | Size        |
|----|------------|-------------|
| 1  | Object 1   | 6,1 x 6,1cm |
| 2  | Object 2   | -           |
| 3  | Object 3   | 7,0 x 7,0cm |
| 4  | Object 4   | -           |
| 5  | Object 5   | 5,9 x 5,9cm |
| 6  | Object 6   | -           |
| 7  | Object 7   | 7,9 x 6,9cm |
| 8  | Object 8   | -           |

Same as the previous object shape detection, only objects that are red in the process of detecting the size of this object. So, the recognition of the size of this object can only work on objects 1,3,5, and 7. In the object classification table, the the size detection of the object can work as expected, because the results obtained are in accordance with the classification of objects.

#### 4.5. Object qualification have been tested

From some of the tables above, it can be seen the results of testing objects, from testing the color, shape, and size of objects. Next is to determine the object according to specifications or not with the listing program below.

```
len(approx) == 4 ) and (area > 5000) and (dimA > 6.8 )
and (dimA < 7.2) and (dimB > 6.8 ) and (dimB < 7.2) ):
else : ()
```

Fig. 10. Listing program

As for the results of the qualification of objects that have been tested can be seen in Table 7 below.

| No | Experiment | Result   |
|----|------------|----------|
| 1  | Object 1   | Accepted |
| 2  | Object 2   | Reject   |
| 3  | Object 3   | Reject   |
| 4  | Object 4   | Reject   |
| 5  | Object 5   | Reject   |
| 6  | Object 6   | Reject   |
| 7  | Object 7   | Reject   |
| 8  | Object 8   | Reject   |

## 5. Conclusion

Based on testing and analysis that has been carried out on object detections based on color, shape and size with a webcam, it can be concluded that:

- The manufacture of hardware can be made in accordance with the planning device that have been made.
- This device can work properly as expected, because in testing all these components can work in accordance with a predetermined system.
- Object detection by webcam can detect objects based on color, shape, and size.
- This test used 8 objects that have been prepared with different colors, shapes and sizes each other, and have been tested several times. There is minimum error, either the object recognition error by the webcam or other tests

## References

- Aditya, M. R. V., Husni, N. L., Pratama, D. A., & Handayani, A. S. (2020). Penerapan Sistem Pengolahan Citra Digital Pendeteksi Warna pada Starbot. *TEKNIKA*, 14(2), 185-191.
- Andrizal, Hidayat, A., Susanti, R. R., Chadry. (2016) .Computare Vision Berbasis Camera dan Mini PC untuk Identifikasi Kecacatan Penutup Kemasan Minuman Kaleng. *Jurnal Ilmiah Poli Rekayasa*, 12(1).
- Alfaro-Almagro, F., Jenkinson, M., Bangerter, N. K., Andersson, J. L., Griffanti, L., Douaud, G., ... & Smith, S. M. (2018). Image processing and Quality Control for the first 10,000 brain imaging datasets from UK Biobank. *Neuroimage*, 166, 400-424.
- Aviolita, E. (2019). *Pendeteksi Penyakit Tanaman Buah Naga Menggunakan Kamera Berbasis Raspberry Pi* (Doctoral dissertation, Institut Teknologi Nasional Malang).
- Bradski, G., & Kaehler, A. (2008). *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc."
- Daud, N. B. (2007). *Application of colors sensor in an automated system*. Technical University Malaysia.
- Ghanie, C. E. S., & Setiawan, F. B. (2020). Penerapan Sistem Pan-Tilt Camera untuk Deteksi Objek berdasarkan Warna menggunakan Raspberry Pi. In *Seminar Nasional Teknik Elektro 5*(1), 92-96.
- Handoyo, E. D. (2006). Perancangan mini image editor versi 1.0 sebagai aplikasi penunjang mata kuliah digital image processing. *Jurnal Teknik Informatika dan Sistem Informasi*, 2(2), 219230.
- Januar Insani, O. (2016). *Implementasi Lane Detection Dengan Metode Hough Transform Untuk Penilaian mengemudi Berdasarkan Marka Jalan (Studi Kasus Sukses Mandiri)* (Doctoral dissertation, Universitas Komputer Indonesia).
- Juliando, J., & Husin, Z. (2019). Rancang Bangun Alat Penyortir Minuman Kaleng Menggunakan Camera Dan Deteksi Warna (Doctoral dissertation, Sriwijaya University).
- Mubarrok, A. R., & Rahmawati, D. (2020). Rancang Bangun Timbangan Buah Anggur Digital Otomatis Berbasis Webcam Menggunakan Transformasi Hough. *Science Electro*, 12(2).
- Najmurokhman, A., Nugraha, A., Kusnandar, U. K., & Wibowo, B. (2017). Perancangan dan Realisasi Sistem Pendeteksi Objek Menggunakan Perangkat Lunak. in *Prosiding Seminar Nasional Ilmu Pengetahuan dan Teknologi*. Cimahi: Jenderal Achmad Yani (SNIJA)2017.
- Prayitno, Y. P., & Harianto, M. C. W. (2011). *Rancang Bangun Aplikasi Pendeteksi Bentuk dan Warna Benda Pada Mobile Robot Berbasis Webcam*.
- Pulungan, A. B., & Nafis, Z. (2021). Rancangan Alat Pendeteksi Benda dengan Berdasarkan Warna, Bentuk, dan Ukuran dengan Webcam. *JTEIN: Jurnal Teknik Elektro Indonesia*, 2(1), 49-54.
- Purwanti, S., Febriani, A., Mardeni, M., & Irawan, Y. (2021). Temperature Monitoring System for Egg Incubators Using Raspberry Pi3 Based on Internet of Things (IoT). *Journal of Robotics and Control (JRC)*, 2(5), 349-352.
- Syafriadi, M. (2017). Rancang Bangun Alat Pendeteksi Warna Menggunakan Kamera Dan Output Suara Berbasis Komputer (Doctoral Dissertation, Politeknik Negeri Sriwijaya).
- Yulian, F., & Mayasari, Z. M. Pengembangan Teknik Pengolahan Dan Analisis Citra Penginderaan Jauh Melalui Perancangan Tapis Morfologi Matematik.
- Zarwani, H. (2019). *Rancang Bangun Alat Penyortir Buah Tomat Berdasarkan Ukuran Dan Warna Menggunakan Metode Segmentasi Hsv Berbasis Raspberry Pi 3b+*.