APPLICATION DEVELOPMENT OF VIRTUAL WRITING USING COLOR FILTERING ALGORITHM

M Azhan Mizani[1], Nelly Indriani W[2]

[1,2] Teknik Informatika-Universitas Komputer Indonesia
Jalan Diaptiukur No.112-116 Bandung. 40132
E-mail : m.azhanmizani21@gmail.com[1], indiwidi@gmail.com[2]

ABSTRACT

Application development of virtual writing using color filtering algorithm is an application for writing letters and words virtually using object tracked as pointer. This application is built to fix previous research deficiencies. The previous research only detects red color and disturbs the result if there is another same color and size object to object marker.

Color filtering is an image processing method based on color. Camshift is a tracking object centroid method. In this research, "application development of detection and color tracking virtual writing" using webcam and object color as marker. Application is run by selecting object area and generating search window, then calculating color search window histogram to take color as a model object with color filtering method, then calculating search window centroid with camshift method. Application is built using addition library OpenCV. The purpose of this application is to write virtually without using mouse.

Result of testing application conducted with object size 4.5x2cm gets the good result on light 9-15 watt from distance 10 cm – 100 cm. the conclusion, object size will influence the result on certain distance, and distance will influence the detection ability based on object size. Unfavorable light will influence the detection of object.

Kata kunci : *Deteksi, Tracking, Objek, Color Filtering, CamShift.*

1. INTRODUCTION

Digital image processing is a process done on an image with the help of a computer to improve the quality of the image or image to be better and get the color value contained in it [1]. The link between digital image processing and computer vision is image processing in the steps of computer vision called preprocessing [2].

Computer vision is the transformation or change of data that can be either a still image or a camera video into another form or a new representation. Computer vision can be defined by means of image processing associated with image acquisition, processing, classification, and overall coverage, decision making followed by image identification [2].

One development of the use of computer vision techniques is the tracking of objects. The object tracking aims to detect and follow the position of a desired moving object. Many object tracking is required by various vision based applications such as human computer interface, video compression / communication and security systems [4]. Tracking objects must be capable of detecting moving objects, filtering noise, and other unnecessary movements. Research on object tracking has been done by Rio [5]. In the research results can be concluded, the application has been able to detect the color of an object and mark the position of pixels passed, the developed application still has a deficiency in the tracking results are still disjointed if the movement of objects quickly.

Color filtering is an image processing technique based on specific colors. How it works on color filtering is to compare the color component of each pixel image with the selected color. Color filtering is used in the early stages to detect color-based objects. After finding the specified color, the research can proceed to the tracking process of the object. the advantage of using the color filtering method is one of the simplest methods of color filtering in the RGB color space (Red, Green, Blue) that provides the fastest performance on RGB images [4].

The method used to search the centroid of a filtered object by color is the CamShift (Continuously Adaptive Mean Shift) Algorithm. The CamShift algorithm was originally developed for head and face tracking applications based on the color histogram of the target object. CamShift is the development of a mean shift algorithm that searches for the largest color probability produced by color histogram [6]. This algorithm is used, because it can change the size of the search window in the event of rotation, large and small objects causing the pixel distribution value to the object to change [13]. And also calculate the entire pixel value in the capture area to show the probability value that belongs to the object [6].

Based on what has been described then the color filtering and CamShift method will be used in research to develop previous research. Applications

that will be built on this research will be called virtual writing.

The purpose of this research is to know the accuracy of tracking object by using the method of CamShift.

2. RESEARCH CONTENTS

2.1 System Description

On the system will be made will do the color tracking, where this system will determine the midpoint on the object to be in tracking and will follow the movers of the object. In tracking, lighting is very influential and also the color of the object to be tracked jug important things, objects with black and white on this system, can not be tracked properly.

2.2 Image Processing

An image has rich information, but often the image that we have a degraded (degradation), for example, contains a defect or noise (noise), the color is too much contrast, less sharp, blur (blurring), and so on. Of course, this kind of image becomes more difficult to interpret because the information conveyed by the image becomes lessened.

In order for the image to be interrupted easily interpreted (both by human and machine), then the image needs to be manipulated into another image of better quality.

Image processing aims to improve image quality for easy interpretation by humans or machines (in this case the computer). Image processing techniques transform images into other imagery. So, the input is the image and the output is also the image, but the output image has better quality than the input image. Included in this field also is compressed image (image compression).

2.3 Computer Vision

Another terminology that is closely related to image processing is computer vision or machine vision. In essence, computer vision tries to imitate the workings of the human visual system. Human vision is as complex as it is. Man sees the object with the sense of sight (eye), then the image of the object is forwarded to the brain to be interpreted so that the human understands what objects appear in his eyes. The results of this interpretation may be used for decision making.

Computer vision is an automated process that integrates a large number of processes for visual perception, such as image acquisition, image processing, classification, recognition, and decision making. Computer vision consists of techniques for estimating the characteristics of objects in imagery, measuring characteristics related to object geometry, and interpreting the geometry information.

The processes in computer vision are divided into 3 (three) activities:

1. *Obtaining or acquiring digital imagery.*
2. *Perform computational techniques to processor modify image data (image processing operations).*
3. *Analyze and interpret Imagery using the results of processing for specific purposes, such as guiding robots, controlling equipment, monitoring manufacturing and others.*

2.4 Open Computer Vision (Open CV)

OpenCV (Open Computer Vision) is an open source library that contains more than 500 algorithms are optimized for image and video analysis. Since its introduction in 1999, OpenCV has been widely adopted as the primary development tool by the community of Researchers and developes in computer vision. OpenCV was originally developed at Intel by Gary Bradski as an initiative to promote research in vision and promote the development of vision-based rich, CPU-intensive applications. After several beta series were launched, version 1.0 was launched in 2006. The second major launch took place in 2009 with an OpenCV 2 version that aims to change specifically for the new C ++ interface. OpenCV announced as a non-profit foundation that relies on raising public funds for its development.

The features included in OpenCV include:

1. *Manipulation of image data (allocation, release, duplication, regulation, conversion).*
2. *Image and I / O video (and camera-based input file, the output image / video files).*
3. *Manipulation of vectors and matrices and linear algebra (products, solutions, eigenvalues, SVD).*
4. *Diverse dynamic data structures (lists, rows, graphs).*
5. *Basic image processing (filter, edge detection, suduk detection, sampling and interpolation, color conversion, morvologi operations, histogram).*
6. *Analysis of the structure (komponene related, contour processing, transformation distance, variations moment, the Hough transform, plygonal estimates, adjusting line, Delaunay triangulation.*
7. *Camera calibration (find and trace calibration patterns, calibration, basic matrix estimation, homograph estimation, stereo correspondence).*
8. *Analysis of motion (optical flow, motion segmentation, search).*
9. *Object recognition (eigen method, HMM).*
10. *Basic Graphical User Interface or GUI (displaying image / video, the handling of mouse and keyboard, scroll-bars).*

11. *Labeling image (lines, polygons, text images).The modules included in OpenCV include:*
12. *cv - the main function of OpenCV.*
13. *cvaux - OpenCV helper function.*
14. *cxcore - supporting data structures and linear algebra.*
15. *highgui - GUI functionality*

2.5 *RGB Color Processing*

There are basic colors that are like red, blue and green, from the color poko can doi combine into several other colors, here is a picture of RGB, can be seen in picture 1.

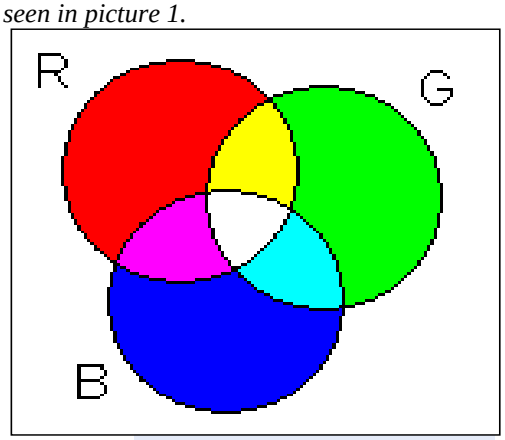


figure 1 Example of RGB Representation

Color processing using RGB model is very easy and simple, because the color information in the computer is packed in the same model [1]. The thing to do is to read the values of R, G, and B on a pixel. One way to calculate color is by normalizing the three components. Normalization is important especially when a number of images are captured with different illumination. The calculation results of each component of the normalized primary colors will eliminate the effect of lighting so that each component of the color can be compared with each other even though derived from images with unequal lighting conditions.

2.6 HSV Color Type

The HSV color space can be interpreted in Hue, Saturation, andValue. The HSV model describes colors the same as how the human eye sees color. RGB is a combination of 3 basic colors, whereas, HSV describes colors using comparisons such as color, intensity and lighting. Here are the formulas to change the color of RGB to HSV.

Change the RGB value range from 0..255 to 0..1:

$$R' = R/255 \quad G' = G/255 \quad B' = B/255 \tag{2.1}$$

Looking for Cmax, Cmin, and Delta values

$$Cmax = max(R', G', B') \quad Cmin = min(R', G', B') \tag{2.2}$$

$$\Delta = Cmax - Cmin \tag{2.3}$$

To find the value of H (hue) on an object can be done dean the following equation.

$$h(hue) \begin{cases} 60° \times \left(\frac{G' - B'}{max - min} mod\ 6\right), if\ max = R' \\ 60° \times \left(\frac{B' - R'}{max - min} + 2\right), if\ max = G' \\ 60° \times \left(\frac{R' - G'}{max - min} + 4\right), if\ max = B' \\ 0, \quad if\ max = min \end{cases} \tag{2.4}$$

2.7 Color FIltering

Color filtering is an image processing technique that is used to manipulate an image based on specific color [7]. The way it works is by comparing the color components of each pixel image with a specific color. If a specific color matches the color of the pixel component is left alone. However, if the color does not match the specific color of the pixel color components are converted to the background color, usually black.

The colors used in the Color Filtering can be represented in different color spaces. There are several known color spaces, such as RGB (Red, Green, Blue), HSV (Hue, Saturation, Value), YCbCr, etc. HSV is a very suitable color space for identifying basic colors, where the base color is used in research as a robot identification color. In addition, HSV tolerates changes in light intensity. This is the advantage of HSV compared to other color spaces [20].

2.8 Continuously Adaptive Meanshift (Camshift)

CamShift algorithms originally developed for head and face tracking application. Based on the color object histogram of the target [6]. CamShift is an improvisation of the mean-shift algorithm seeking the greatest probability of colors produced by the color histogram [6]. This algorithm is used as an important part of other applications such as video

surveillance, human computer interaction, head and face tracking application. The basis of the workflow CamShift illustrated in Figure 1. Algorithm CamShift originally developed for head and face tracking application. Based on the color histogram of the target object [6]. CamShift is an improvisation of the mean-shift algorithm seeking the greatest probability of colors produced by the color histogram [6]. This algorithm is used as an important part of other applications such as video surveillance, human computer interaction, head and face tracking application. The basis of the workflow CamShift illustrated in Figure 1.

Follow the steps of CamShift operation:

Step 1 set the coordinates (x, y) as the initial location of the search window.

Step 2 find the value 0, the following formula of the first and second moments of the image from the search window:

Where I (x, y) is the pixel value (probability) at position (x, y) in the image, and x and y is the range of the search window.

$$M_{00} = \sum_x \sum_y I(x,y) \quad (2.5)$$

$$M_{10} = \sum_x \sum_y xI(x,y); \quad (2.6)$$

$$M_{01} = \sum_x \sum_y yI(x,y); \quad (2.7)$$

Step 3 calculates the location of the mean (centroid) in the search window:

$$X_c = \frac{M_{10}}{M_{00}}; Y_c = \frac{M_{01}}{M_{00}} \quad (2.8)$$

Step 4 Repeat steps 2 and 3 sampaikonvergensi or parameter, unchanged or zero.

2.9 System Analysis

The system built is a color-based tracking system with marker by human movement to be captured using a webcam. A system that does control over the movement of human color and webcam with implementing the filtering method and camshift color. This system requires input in the form of image (video) captured in real time and the image will be processed by color filtering and CamShift to get the centroid of colors and objects to perform the tracking process. Here's a picture of the system flow to be built in Figure 2.

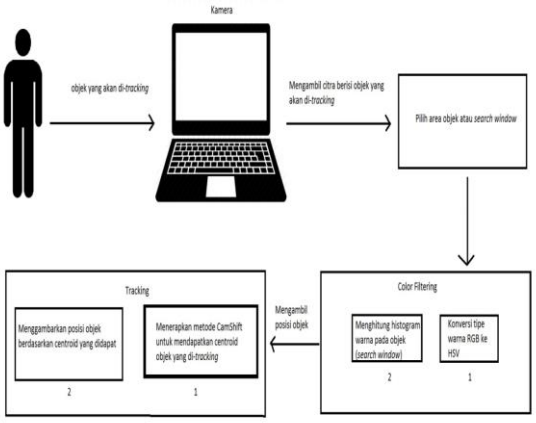


Figure 2 Flowchart Analysis System

2.10 Method Analysis

In this research, using color filtering and CamShift. Color filtering algorithm is used to determine the color that will be tracking after selecting the object area will be calculated histogram warnyanya. Then the object tracking is done using CamShift. The following outline of the process that occurs in the system can be seen in Figure 3:

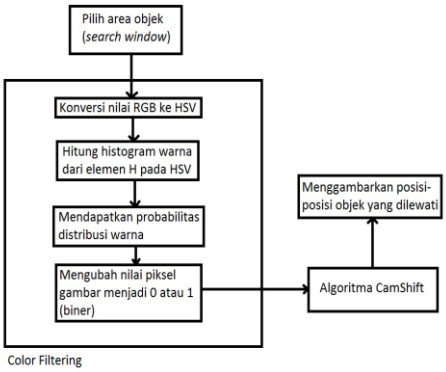


Figure 3 Flowchart Object Tracking Algorithm

Seen in Figure 3 is the whole process that will occur in this research. Here is an explanation diagram of the process on the system to be built:

1. *Select the object area (search window) to get the color of the object.*
2. *Convert RGB to HSV color space for each pixel in the image.*
3. *Make a color histogram of the element H (Hue) in HSV to be a reference to find the same color in the search window.*
4. *Calculate the color probability of the histogram already obtained.*
5. *Changed the pixel value to 0 or 1 called the image binary process.*
6. *Detect and get the middle value using the CamShift algorithm.*

7. *The middle value obtained is used as a reference to describe the position of the object being bypassed.*

2.10.1 RGB to HSV Conversion

After the system select area of objects to be in tracking then the next step is to convert the RGB value to HSV from every pixel that exist in the image. In this step the value taken is only the value of the element H (hue). At this stage, the image data used is the RGB value of the image 14 x 8 containing objects tracking di- (green book) starts from the coordinates (15.9) to (22.22) of simulation data 25 x 25 chosen to be calculated Histogram color. The steps taken are:

1. *Change the RGB value from 0..255 to 0..1 with the equation (2.1).*
2. *Then from the HSV result, the next step looks for Cmax and Cmin using equation (2.2).*
3. *Then look for vulnerable values from Cmax and Cmin (2.3.).*
4. *Finding niali H by using equation (2.4).*

2.10.2 Color Histogram

Creating a color histogram aims to group colors within a certain range of values. The color histogram created in this study had 8 bin for storing a variety of colors with a hue value range which is determined based on the smallest H (0) and largest (356) to determine the range of values both then divided by 8.

2.10.3 Image Binaryization

Image binaryization is the process of changing the value of the image to 0 or 1. The value of H converted to 1 is the value that goes into the largest frequency in the calculations performed on the calculation of the value of H..

2.11 CamShift

CamShift is a method for tracking on the object after HSV to RGB color conversion, color histogram calculation, and binerisasi image. How CamShift works is by finding the densest data points in the search window and then generate the middle value of the data points. The following flowchart and algorithm of CamShift, can be seen in Figure 4.

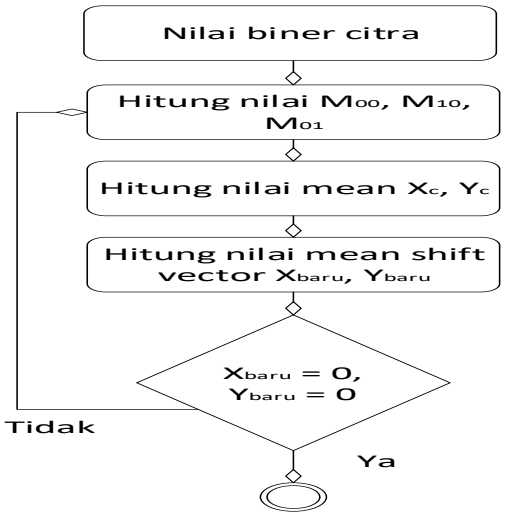


Figure 4 Flowchart algorithm Camshift

2.11.1 Calculation the values of 0 and 1 moments (M10, M01, M00)

Calculating the value of M00 that by adding all data points in the value of x, y coordinates that are on the search window. Determining the M00 value of the calculation area by using the equation (2.5). In the calculation below, I (x, y) is the value in the pixel of value 1. M10 is the value to find the distance between the new middle and middle center on the x axis, whereas M01 is value to find the distance between the middle and the new center on the y axis. Find the value of M10 and M01 by using equation (2.6).

2.11.2 Mean Xc and Yc

The mean of Xc and Yc aims to find the distance between the old and new coordinates. To find Xc and Yc we use equation (2.7).

2.11.3 Mean Shift Vetor

Mean shift vector aims to determine the value of x, y coordinates a new one by Xbaru = Xc + X and Ybaru = Yc + Y. Repeat this step until the new Y Xbaru and not berunbah. So when it is not riding that is the starting point in getting in the object tracking.

2.12 System Testing

This method is used to compare the performance of the software. Parameters used in this testing technique include:

1. *The state of light*
   Circumstances of light are grouped in good lighting and poor lighting.
   - *Parameters with good lighting if indoors contain light produced from the sun or from lights with a light output of between 9 and 15 watts.*
   - *Parameters with poor lighting if indoors lack the light produced by the sun or from*

lights with a light output of less than 9 watts.

- .

2. *background conditions*
   Background conditions captured by the webcam no premises marker of the same color and size objects larger than the size of the marker.
3. *The shape and size of the object (marker)*
   The object shape is viewed from a minimum length of 4.5 cm and a minimum width of 2 cm from the marker used for testing.
4. *Color object*
   The color of objects that can be tracking is other than black and white.
5. *Distance object with camera*
   Distance testing is done to find out how far the object will be detected. The test value parameter is calculated from the number of detected objects from each test. The following is an explanation of the assessment in Table 1:

Table 1 Eventing Objects Detected

No	*Number of Object Detected*	*Assessment*
1	*10*	*Very good*
2	*8-9*	*Good*
3	*7*	*Pretty good*
4	*1-6*	*Not good*
5	*0*	*Not detected*

2.12.1 Testing

Object detection tests are performed to test the light, distance, and size of objects that the application can detect. Distance tested from 10cm - 100cm (multiples of 10) in 10 times and using object size 1x1cm, 4.5x2cm, and 10x5cm on good light 9-15 watt lights and less good light smaller than 9 watt or greater Of 15 watt bulbs. Here are the results of the tests performed on the distance and size of different objects. Description on the detection test object T = detected and TT = undetectable.

2.12.1.1 First Testing

The first test was performed using the following tools:

- *Webcam 3MP*
- *The object size is 1x1cm*
- *9-15 watt lamp (Good lighting)*

The following test results are organized into tables:

Testing detection:

Table 2 Testing First Object Detection

No	***Object distance (cm)***	***Good Lighting***		***Bad Lighting***	
		Object		***Object***	
		T	***TT***	***T***	***TT***
1	*10*	√		√	
2	*20*	√		√	
3	*30*	√		√	
4	*40*	√		√	
5	*50*	√		√	
6	*60*	√			√
7	*70*	√			√
8	*80*	√			√
9	*90*	√			√
10	*100*	√			√
Number		*10*	*0*	*5*	*5*
Assessment		***Very Good***		***Pretty Good***	

Testing letters:

Table 3 First Test of Letter Writing

Huruf	***Jarak***	
	Good Lighting	***Bad Lighting***
A-Z	*10 cm – 40 cm*	*10 cm – 20 cm*

Testing words:

Table 4 First Test of Letter Writing

Word	***Object distance (cm)***					
	Good Lighting			***Bad Lighting***		
	10	***50***	***100***	***10***	***50***	***10 0***
SAYA	√	-	-	√	-	-
PERGI	√	-	-	√	-	-
MAKAN	√	-	-	√	-	-
Number of Words Read	*3*	*0*	*0*	*3*	*0*	*0*

2.12.1.2 Second Testing

The second test is carried out using the following tools:

- *Webcam 3MP*
- *The object size is 4.5x2cm*
- *9-15 watt lamp (Good lighting)*

The following test results are organized into tables:

Testing detection: :

Table 5 Testing Second Object Detection

No	Object distance (cm)	Good Lighting		Bad Lighting	
		Object		Object	
		T	TT	T	TT
1	10	√		√	
2	20	√		√	
3	30	√		√	
4	40	√		√	
5	50	√		√	
6	60	√			√
7	70	√			√
8	80	√			√
9	90	√			√
10	100	√			√
Number		10	0	5	5
Assessment		Very Good		Pretty Good	

Pengujian huruf :

Table 6 Second Test of Letter Writing

Alphabet	Distance	
	Good Lighting	Bad Lighting
A-Z	10 cm – 100 cm	10 cm – 50 cm

Testing words :

Table 7 Second Words of Writing

Word	Object distance (cm)					
	Good Lighting			Bad Lighting		
	10	50	100	10	50	100
SAYA	√	√	√	√	√	-
PERGI	√	√	√	√	√	-
MAKAN	√	√	√	√	√	-
Number of Words Read	3	3	3	3	3	0

2.12.1.3 Third Testing

The second test is carried out using the following tools:

- *Webcam 3MP*
- *The object size is 4.5x2cm*
- *9-15 watt lamp (Good lighting)*

The following test results are organized into tables:

Testing detection:

Table 8 Third Testing of Object Detection

No	Object distance (cm)	Good Lighting		Bad Lighting	
		Object		Object	
		T	TT	T	TT
1	10	√		√	
2	20	√		√	
3	30	√		√	
4	40	√		√	
5	50	√		√	
6	60	√			√
7	70	√			√
8	80	√			√
9	90	√			√
10	100	√			√
Number		10	0	5	5
Assessment		Very Good		Pretty Good	

Testing letters:

Table 9 Third Testing of Writing Letters

Alphabet	Distance	
	Good Lighting	Bad Lighting
A-Z	30 cm – 100 cm	30 cm – 50 cm

Testing words:

Table 10 Testing three Word Words

Word	Object distance (cm)					
	Good Lighting			Bad Lighting		
	10	50	100	10	50	100
SAYA	-	√	√	-	√	-
PERGI	-	√	√	-	√	-
MAKAN	-	√	√	-	√	-
Number of Words Read	0	3	3	0	3	0

2.12.2 Conclusion Testing

Based on the tests performed, the best results are obtained in the second test. The second test can detect the object on the illumination either within a distance of 10 cm - 100 cm, producing a readable AZ letter at a distance of 10 cm - 100 cm, and the resulting word ("MY", "GO", "EAT") can be read at Writing at a distance of 10 cm - 100 cm. The conclusion obtained that the size of the object affects the results of writing at a certain distance, the object is too large will result in poor results at close range. Distance affects detection capabilities at certain object sizes, the further distance detection capabilities will be reduced on small objects. Whereas in a less good light will affect the color of the object that affects the ability of object detection as well.

3. CLOSING

3.1 Conclusions

Based on the results of research, systems analysis, system design, and system testing on a virtual device of writing this it can be concluded that:

1. *This study has successfully detecting and tracking objects in realtime by using color filtering and camshift.*
2. *Applications can be used to write virtually without using the mouse.*

3.2 Suggestions

This study still lacks in object detection in case of poor light conditions. The next research is expected to be able to solve this problem with certain methods.

BIBLIOGRAPHY

[1] U. Ahmad, Pengolahan Citra Digital dan Pemogramannya, Yogyakarta: Graha Ilmu, 2005.

[2] R. Munir, Pengolahan Citra Digital dengan pendekatan algoritmik, Bandung: Informatika Bandung, 2004.

[3] Fadliansyah, Computer Vision dan Pengolahan Citra, Yogyakarta: ANDI, 2007.

[4] K. Nummiaro, E. Koller-Meier and L. V. Gool, "An Adaptive Color-Based Particle Filter," Preprint submitted to Elsevier Science, 2002.

[5] R. Pradhitya, "Pembangunan Aplikasi Deteksi dan Tracking Warna Virtual Drawing Menggunakan Algoritma Color Filtering," Universitas Komputer Indonesia, Bandung, 2015.

[6] A. D. Mohammed, T. Morris, " An Improved CAMShift Algorithm for Object Detection and Extraction," University of Manchester, Manchester, UK.

[7] A. R. Mulyanto, "Rancang Bangun Aplikasi Pendeteksi Bentuk Dan Warna Benda Pada Mobile Robot Berbasis Webcam", STIKOM Surabaya.

[8] D. Putra, Pengolahan Citra Digital, Yogyakarta, 2010.

[9] S. J. Prince, Computer Vision: Models, Learning and Inference, Cambridge University Press, 2012.

[10] M. H. Purnomo and A. Muntasa, Konsep Pengolahan Citra Digital dan Ekstrasi Fitur, Yogyakarta: Graha Ilmu, 2010.

[11] A. Basuki, "Aplikasi Pengolahan Citra Deteksi Warna," PENS-ITS, Surabaya, 2009.

[12] G. S. Nahla, S. and D. Pramadihanto, "Tracking Bola Menggunakan Robotino". 2014.

[13] L A Mushawwir, S. I Supriana, "Deteksi dan Tracking Objek untuk Sistem Pengawasan Citra Bergerak" Institut Teknologi Bandung, Bandung.