

# Smartphone Technology Applications for Milkfish Image Segmentation Using OpenCV Library

<https://doi.org/10.3991/ijim.v14i08.12423>

Akhmad Qashlim (✉), Basri, Haeruddin, Ardan  
Universitas Al Asyariah Mandar, Polewali Mandar, Indonesia  
qashlim@unasman.ac.id

Inggrid Nurtanio, Amil Ahmad Ilham  
Hasanuddin University, Makassar, Indonesia

**Abstract**—This research presents the use of smartphone technology to assist fisheries work. Specifically, we designed an Android application that utilizes a camera connected to the internet to detect RGB image objects and then convert them to HSV and gray scale. In this paper, Android-based smartphone technology using image processing methods will be discussed, a digital tool that provides fish detection results in the form of length, width, and weight used to determine the price of fish. This application was created using features provided by the OpenCV library to produce binary images. Three main challenges highlighted during application design including C++ QT were used to build the user interface, the contour-active method was used to divide and separate image objects from the background, while the clever edge edge method was used to improve the outline appearance of objects. Both methods are implemented on the Android platform and utilize smartphone cameras as an identification tool. This application makes it possible to provide many benefits and great benefits for farm farmers but on the other hand will create technological gaps.

**Keywords**—Library OpenCV, Citra Segmentation, Android Application, Technology Smartphone.

## 1 Introduction

Having the best smartphone for various needs is important but still depends on the software or application that is owned according to work requirements. Various mobile applications that can help human work [1], [2] and the very trend right now [3] growing so fast and constantly updated and tends to connect hardware into applications such as cameras, sensors, com-passes, Bluetooth, Wi-Fi to meet user needs [4]. The integration of Mobile device is an innovative process to increase labor mobility and flexibility on the workforce [5] and and caused desktop computing to be replaced by the emerging mobile computing technology [6].

Mobile application innovation is now moving towards the digitalization of processes that aim to create quality [7] and the most interesting thing is image processing

technology mobil based [8] to detect product quality through image objects [1], [8] besides also used for visual identification and video analysis [2], [9] and used as solutions to help people with color blindness [10]. Various cloud-based mobile computing applications have been developed and presented to users such as, cellular learning, cellular health services, mobile games or industrial needs [5] The need for mobile technology with the ability to image segmentation in the freshwater fish industry is used to identify the quality and type of freshwater fish [11]. This research succeeded in designing an image segmentation technology based on an android mobile application using the active contour method. [12], [13]. The technology proposed in this research is a form approaching the digitization process used computer vision and image processing to segmentation. The OpenCV library is used for application development because it can work on various operating systems and across platforms [14] as well as having complete algorithms and functions for image processing function [8] besides that it has been implemented in several image processing projects [4], [14]. The focus of this research is to look at and demonstrate the utility of an android mobile application and find out how efficient an image processing application can be used to check and assess the benefits of a product.

### **1.1 OpenCV and platform android**

OpenCV Library (Open Computer Vision) is a library used primarily for image and video processing needs so that it can extract information in it. The OpenCV library has 2500 algorithms first introduced by the Intel Research initiative to advance CPU-intensive applications in 1999. OpenCV library can be built with a variety of sump as programming such as C, C ++, Java, Python that massing one has sample code for real-time computer vision and support a variety of platforms such as windows, Linux [4], Mac OS, iOS and Android [15]. OpenCV Library Version 2.2 is the first version that can be built on the Android operating system and in version 2.3.1 has been equipped with a program code for image processing on the Android system [4]. The data type in OpenCV is IplImage derived from the word Intel Image Processing Library which is the main image structure in OpenCV [8]. Much other research carried out by using OpenCV to create a system capable of capturing, identifying and analyzing the features in the image [14] such as face detection and tracking features that use a combination of Java and C ++ to build programming code [4]. Basically, the image detected will be converted to binary values and OpenCV provides a library for doing this like inRange, threshold, Adaptive Threshold, Canny and others to make binary images to grayscale or color.

Building image processing applications on the Android platform requires an Android software Native Development Kit (NDK) as an API (Application Programming Interface) tool [16]. NDK has a set of features that help use C ++ and C programming codes.

## 1.2 Citra RGB and HSV

The digital image is defined as a function  $f(x, y)$ . The value  $f$  represents the value of the gray degree of the object at points  $x$  and  $y$ , while  $(x, y)$  is the position of the pixel where the value is  $x$  represents the row number while they value represents the column value [17]. RGB ( Red, Green, Blue ) imagery consists of three image fields where each color has a value of each pixel [18]. Object Pendetksian dap at done in several ways one of them by way of color-based HSV. HSV Digital Image defines color in the terminology of Hue (True Color), Saturation and Value (Color Brightness) [19] HSV image contains colors that n is equal to that captured by the human senses. While the colors formed by other models such as RGB are a mixture of primary colors [17]. RGB images are more dominant in the original image of an object.

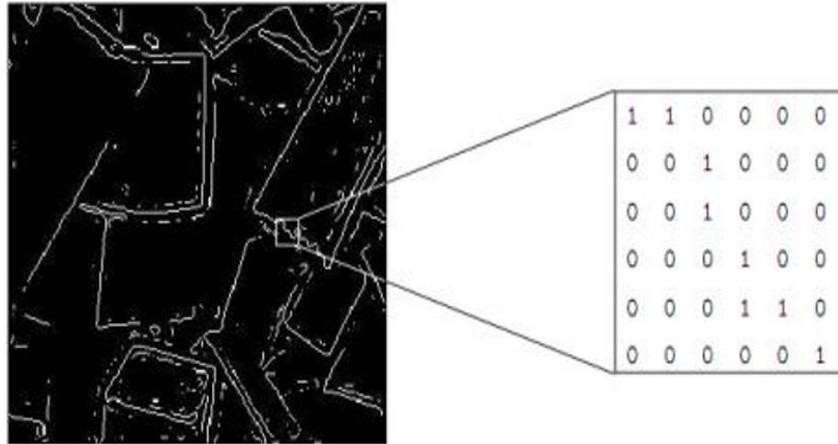
Building image processing applications on the Android platform requires an Android software Native Development Kit (NDK) as an API (Application Programming Interface) tool [16]. NDK has a set of features that help use C ++ and C programming codes.



Fig. 1. RGB type Digital Image

## 1.3 Thresholding technique and binary image segmentation

Threshold is a process segmentation image to change the grayscale image [20] that will produce a binary image [20]; [21] and has two possible pixel values, namely the intensity value of the image that is more than or equal to the threshold value of 1 (white or foreground) and if it is less than the threshold value or value 0 (black or background ) that you want to remove . Each pixel in the binary image has a size of 1 Bit. Processing binary images are done by refer to (1).



**Fig. 2.** Digital Image and Binary Value

Processing binary images with refer to (1) and in floating-point form done with refer to (2):

$$I_{Bin}(x, y) = \begin{cases} 0 & I_{BW}(x, y) < T \\ 255 & I_{BW}(x, y) \geq T \end{cases} \quad (1)$$

$$I_{Bin}(x, y) = \begin{cases} 0 & I_{BW}(x, y) < T \\ 1 & I_{BW}(x, y) \geq T \end{cases} \quad (2)$$

Where :

$I_{Bin}(x,y)$  = Binary pixel value at point (x,y).

$I_{BW}(x,y)$  = Pixel value at point (x,y) in a grayscale image.

T = Threshold value

An image with a grayscale type and model in the arrangement of pixels consisting of rows and columns shown in figure 3. Each pixel has a degree grayscale with a numerical value in the range of 0 (Black) - 255 (White), and each pixel has a size of 8 Bit or 1 Byte. With such a digital image symbolized as a matrix of pixels with a value of 0-255

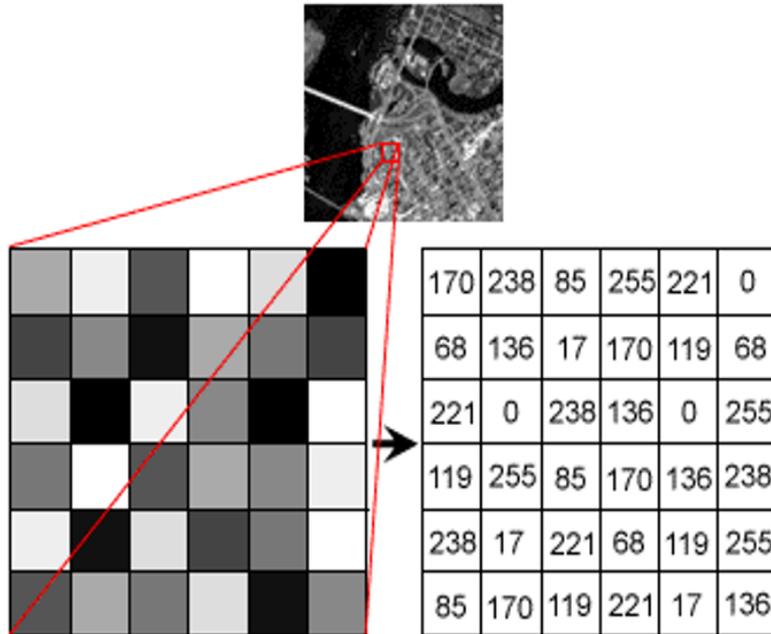


Fig. 3. Grayscale Type Digital Image

## 2 Method

In this section, we will introduce the proposed scene image detection system. We use the Android platform because it is free and open source and is widely used to develop mobile applications [10]. This research uses Java and C++ as the basis for programming code. To create an Android application, the C++ section will be built first by using the Native Android Development Kit (NDK) to improve application performance, while the OpenCV library is used for processing image coding, all programming code is built on the operating system Windows then converse I to APK for Android smartphone applications and uses the camera as a detection tool. The program is run by using the camera on a smartphone [6]. Object samples that will be detected in this study are banding and the results of image processing will display information about weight estimation and price calculation. To get the best object detection results, the minimum Android specification uses version 17 with 1 GB of RAM and a 5 Megapixel Camera. The system workflow in this application can be explained, see Figure 4.

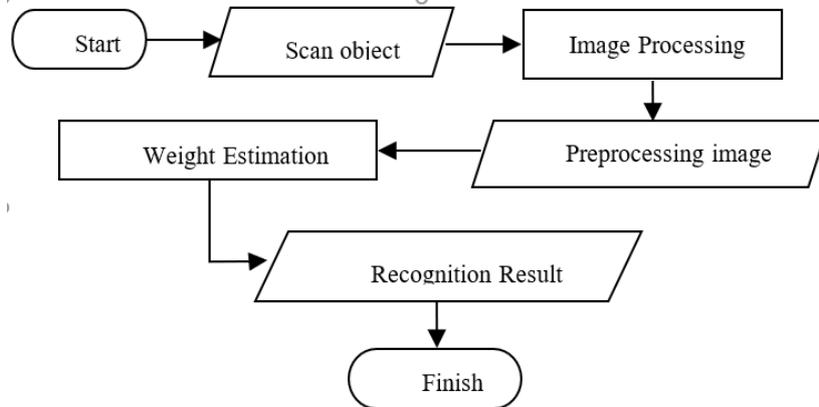


Fig. 4. The Flow of Each Image Processing Level

The first stage, the camera on the android anchor will detect the object and then use it as an input image after the input image has been successfully captured, the next step is *preprocessing the image*. [20], that is, first - the first image with the JPG type and the original color model obtained from the camera is RGB will be converted into two color models namely HSV and Grayscale. Process conversions are explained as follows. Conversion the image from RGB to HSV is done by using the *In Range* feature in the OpenCV library, this is a color-based detection process after which color segmentation is done, this function uses refer to (1) and (2). to get a *pixel* value binary at each color point, then operating a morphological image is a process to reduce noise and also to eliminate black holes on objects. The next stage of detection contour on objects using *canny edge detection* to identify the boundary edges of objects, the data contour then stored in *an array a rotatories' (rectangle accompanied by rotation)*, this process has a detection parameter or *pixel* value. Meanwhile, image conversion from RGB to Grayscale is also carried out by taking detection parameters from the database. Thresholding technique the image on an object applied to be able to distinguish and separate the objects needed color white (foreground) and black background objects that must be removed (background) [21]; [22] so that it produces a binary image [21]. For basic operations thresholding OpenCV function is used *cv:in Range* and *u* to reduce noise the morphological operation is done so that the contour of the object detected in binary imagery is obtained. Next, detect the bounding box i.e. the system will draw a bounding box that shows the width and length of the object to be measured using the *find Contours ()* function that is available in the OpenCV library. This rectangular bounding box is the area found in the contour stage image. This process is done by using functions *find Contour ()* in the OpenCV library, the biggest contour in binary imagery will give a value of length and width in pixel units. The final stage of image preprocessing is performed to the pixel value conversion unit centimeter, Inch and Kilogram. So that the weight of the detected object can be known. The flow of each image processing level is summarized in figure 5.

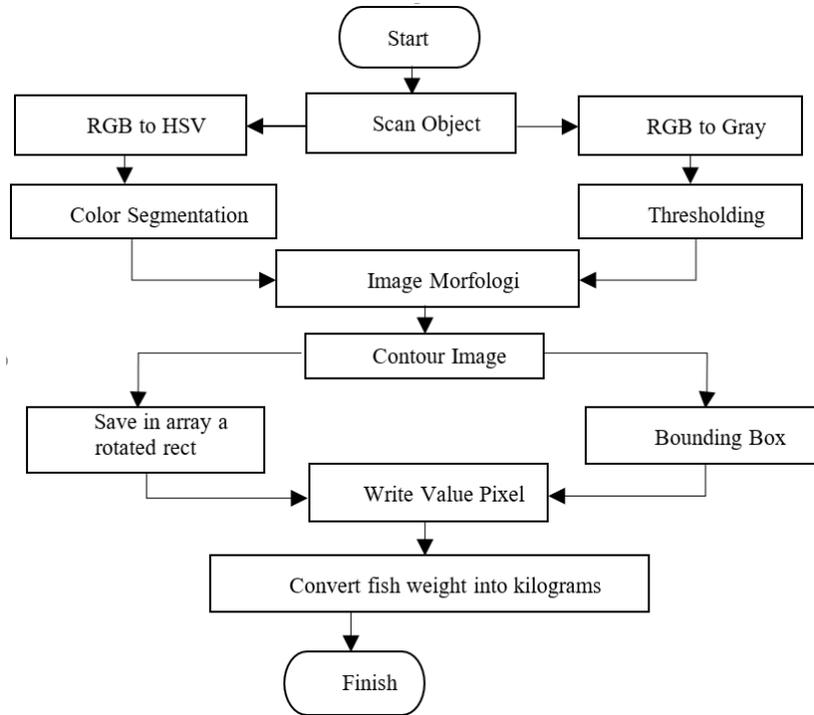


Fig. 5. The Flow of Each Image Processing Level

Camera connection to the internet allows applications and services to be used when users are everywhere [6] Before android smartphone application is really used, we do camera calibration to locate and track detection patterns to be performed by the system, calibration carried out on a large light intensity affects the image capture. Calibration is done with the following equation

$$f = \frac{P * D}{W} \tag{3}$$

Where :

- F = Camera Focal Length (pixel)
- P = Object Length in camera (pixel)
- D = Distance object from camera (cm)
- W = Actual object length (cm).

### 3 Result

Image processing application built on Android The smartphone successfully identifies the object as an area of digital image processing. The object contained in the image is a pixel location which is considered to have a difference in image intensity

and is marked as a boundary box (Bounding Box). Bounding box will give the value of the length and width of the fish object and then calculate the weight estimate.

### 3.1 Pseudocode programming models on device

Finding contours is one of the most important stages in image processing. Contours are the result of a threshold process that forms an area segment and appear as a bounding box that is, a rectangular box follows the edge of the object detected. Some of the relevant OpenCV functions are as follows:

```
/// Find contours
findContours(threshold_output, contours, hierarchy,
CV_RETR_TREE, CV_CHAIN_APPROX_SIMPLE, Point(0, 0) );
```

To convert RGB to Grayscale color, the following functions are used:

```
/// Load source image and convert it to gray
src = imread( argv[1], 1 );
/// Convert image to gray and blur it
cvtColor( src, src_gray, CV_BGR2GRAY );
blur( src_gray, src_gray, Size(3,3) );
```

As for the color conversion from RGB to HSV used function as follows:

```
//Convert Image to HSV
cvtColor(frame,hsv,CV_BGR2HSV); in
Range(hsv,Scalar(h_min,s_min,v_min),Scalar(
h_max,s_max,v_max),hsv);
imshow("hsv",hsv);
```

The detection results will be stored in the SQL database system to become calibration data. The program method for storing new detection parameters.

```
if(!db.cekDataDetector()){
db.addParameterAutoDetect("0","0","56","255","255","2
55","time");
}
```

The detection results will show the length and width of the object in the centimeter unit, the value will be converted to Inch then weighted using the following function

```
// conversion cm to inch
double inch = cm / 2.54;
// Formula to find fish weight.
B = (L * (G * G) ) / 800;
// return the contents of the result variable
```

The Android smartphone application built has a control panel for selecting HSV colors by determining the minimum and maximum HSV color values. The program code used is as follows:

```
//Create a trackbar on the HSV frame
inRange(hsv,Scalar(h_min,s_min,v_min),Scalar(h_max,s_max,v_max),hsv);
createTrackbar("H_MIN: ", "hsv", &h_min,255);
createTrackbar("H_MAX: ", "hsv", &h_max,255);
createTrackbar("S_MIN: ", "hsv", &s_min,255);
createTrackbar("S_MAX: ", "hsv", &s_max,255);
createTrackbar("V_MIN: ", "hsv", &v_min,255);
createTrackbar("V_MAX: ", "hsv", &v_max,255);
```

The implementation of the program code that has been made will be the rule in the device used. To see how the device works, the system framework is presented in Figure 6. The system framework shows the object detection process and then converts the color from RGB to HSV and Grayscale to get a line that produces values of length and width in centimeters. This value is then in conversion to Inch unit and kilogram (kg) to get a weight value. So that the end result of processing this image is a value of the weight of an object, in this case, is milkfish.

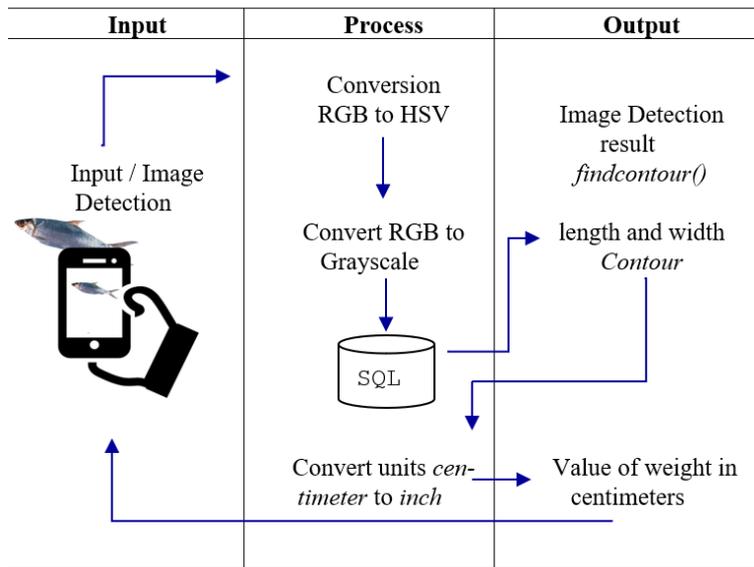


Fig. 6. System Framework of Android Smartphone

### 3.2 Implementation

Android-based detection system modeling smartphone has an interface for displaying images of milkfish captured by cameras and pieces setting minimum and maximum HSV values and Grayscale. The home interface image is presented in Figure 7.

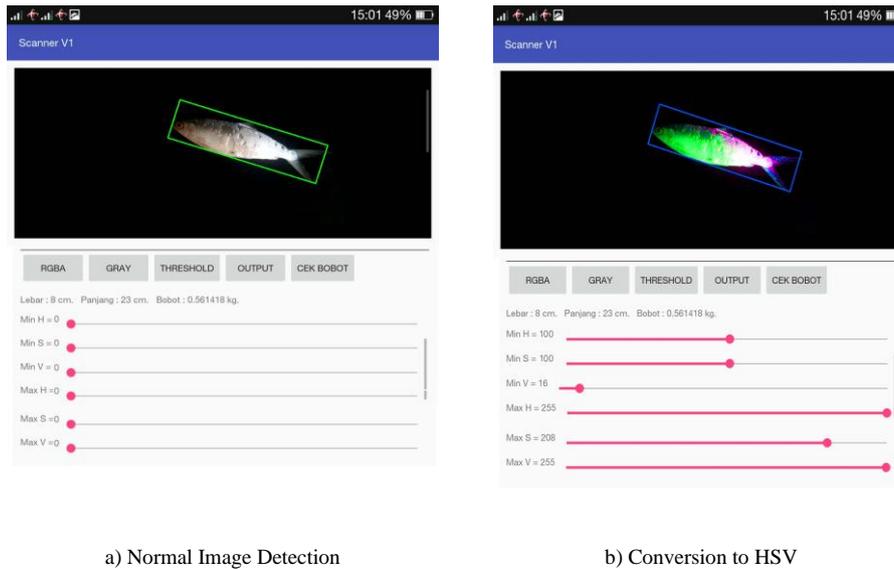
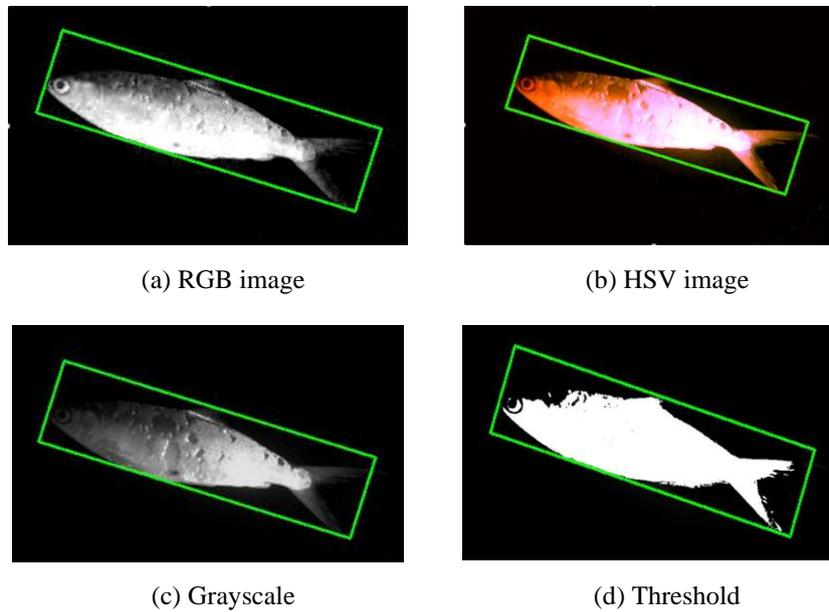


Fig. 7. HSV value conversion settings

Figure 7 is the original image captured by the camera in RGB format with HSV = 0, while figure 7 b shows image conversion from RGB to HSV. The Android smartphone application has a control panel that uses the *Trackbar()* function on OpenCV, this serves to select or adjust the color of HSV both minimum and maximum. The value of  $h_{min} = 100$ ,  $h_{max} = 255$ ,  $s_{min} = 100$ ,  $s_{max} = 208$ ,  $v_{min} = 15$ ,  $v_{max} = 255$ . The value of  $H$  (Hue) represents the color,  $S$  (Saturation) represents the dominance of colors and  $V$  (Value) represents the level of brightness [19]. This value gives a reddish-green color. If the saturation value is 0 then it is not colored at all or between black and white. In the RGB color type, the HSV value can be seen in the component with the lowest value (min) and the highest value (max) or in other cases it can be called the lower and upper limit values. The color obtained from the HSV conversion will not affect the length and width and the weight produced.



**Fig. 8.** Result of the RGB / HSV / Gray / Threshold Image Type Conversion

Figure 8 shows the results converting the color of the object from the original the green line around the object is the *bounding box* created with the OpenCV *findContours ()* library, this line will determine the length and width of the fish object detected. To obtain the contour value, you must consider two contours, one inside and one outside. Because the contour line with pixel width is greater than one, then the contour value in using the *Inner* function will be used in the OpenCV library.

### 3.3 Advantage

Android smartphones with image processing capabilities to get the weight and price of milkfish have provided new and practical ways for fishermen or pond farmers to see the weight of fish without having to use scales or traditional methods. In addition, the application provides a list of products or types of milkfish along with the weights and prices that have been detected previously. Android smartphone application.

### 3.4 Limitation

The application cannot be used for a large number because it has not considered the efficiency and required power consumption. Autofocus on the camera does not work well so that the distance shooting still affect the results of the detection, in addition, design user interface application design has not been based on the concept of user experience so that it becomes familiar.

## 4 Conclusion

Image processing methods that exist in the OpenCV library have worked well on Android devices. The capacity of performance and many functions of the algorithm the one in the makes it easier for a war on android apps. In addition, the image processing method that has been entered into the smartphone android device has succeeded in detecting objects and digging up information that exists on these objects as the purpose of this study. Even so the system that was built and the Android smartphone device as a detection medium has not considered the power consumption needed so it is not possible to control all objects in large numbers other than that the thresholding technique used on an image, is still strongly influenced by uneven lighting or image factors blurry which causes the histogram can not be partitioned properly so that it does not give satisfactory results. To anticipate some of the deficiencies that exist, further research can be done by increasing the flexibility and sensitivity of the detection camera and better calibration.

## 5 Acknowledgement

This research was funded by a grant from the Ministry of Research and Technology's DRPM in 2019. Thank you to the partner team of the University of Hasanuddin Makassar who has provided direction in research to write articles and also thanks to the research of the Al Asyariah Mandar University Computer Science Faculty which provided the facilities without completing this research

## 6 References

- [1] S. S. Martinez, J. G. Ortega, J. G. Garcia, A. S. Garcia, and E. Estevez, "An industrial vision system for surface quality inspection of transparent parts," Springer-Verlag. Springer-Verlag Berlin Heidelberg, London, 2013.
- [2] R. A. Ciora and C. Mi. Simion, "Industrial Applications of Image Processing." Acta Universitatis Cibiniensis – Technical Series, Romania, pp. 17–21, 2014. <https://doi.org/10.2478/aucts-2014-0004>
- [3] A. Hussain, E. O. C. Mkpojiogu, N. Ishak, and N. Mokhtar, "A Study on the Perceived Mobile Experience of Myeg Users," Int. J. Interact. Mob. Technol., vol. 13, no. 11, pp. 4–23, 2019. <https://doi.org/10.3991/ijim.v13i11.11306>
- [4] A. Anuar, K. M. Saipullah, N. A. Ismail, and Y. Soo, "OpenCV Based Real-Time Video Processing Using Android Smartphone," Int. J. Comput. Technol. Electron. Eng., vol. 1, no. 3, pp. 58–63, 2011.
- [5] R. Bernsteiner, B. Ebersberger, and D. Kilian, "Mobile Cloud Computing for Enterprise Systems: A Conceptual Framework for Research," Int. J. Interact. Mob. Technol., vol. 10, no. 2, pp. 72–76, 2016. <https://doi.org/10.3991/ijim.v10i2.5511>
- [6] N. Katuk, N. H. Zakaria, and K. R. Ku-Mahamud, "Mobile Phone Sensing using the Built-in Camera," Int. J. Interact. Mob. Technol., vol. 13, no. 2, pp. 102–114, 2019. <https://doi.org/10.3991/ijim.v13i02.10166>

- [7] F. Wei, C. Alias, and B. Noche, “Applications of Digital Technologies in Sustainable Logistics and Supply Chain Management,” in Springer Nature Switzerland AG, Switzerland: Springer Nature Switzerland, 2019, pp. 235–263. [https://doi.org/10.1007/978-3-319-98467-4\\_11](https://doi.org/10.1007/978-3-319-98467-4_11)
- [8] G. Xie and W. Lu, “Image Edge Detection Based on Opencv,” *Int. J. Electron. Electr. Eng.*, vol. 1, no. 2, pp. 104–106, 2013.
- [9] M. Boudhane and B. Nsiri, “Underwater Image Processing Method for Fish Localization and Detection in Submarine Environment,” *Image Vis. Comput.*, vol. 00, pp. 1–15, 2016. <https://doi.org/10.1016/j.jvcir.2016.05.017>
- [10] L. A. Elrefaei, “Smartphone Based Image Color Correction for Color Blindness,” *Int. J. Interact. Mob. Technol.*, vol. 12, no. 3, pp. 104–119, 2018. <https://doi.org/10.3991/ijim.v12i3.8160>
- [11] I. Sharmin, N. F. Islam, I. Jahan, T. A. Joye, M. R. Rahman, and M. T. Habib, “Machine vision based local fish recognition.” Springer Nature Switzerland, Bangladesh, pp. 1–12, 2019. <https://doi.org/10.1007/s42452-019-1568-z>
- [12] Suprijanto, F. I. Muchtadi, and I. Setiawan, “Segmentasi Citra Secara Semi-Otomatis Untuk Visualisasi Volumetrik Citra Ct-Scan Pelvis,” *MAKARA of Technology Series*, vol. 13, no. 2. Makara, Teknologi, Bandung, Indonesia, pp. 59–66, 2009. <https://doi.org/10.7454/mst.v13i2.469>
- [13] R.J. Hemalatha, T. R. Thamizhvani, A. J. A. Dhivya, J. E. Joseph, B. Babu, and C. R., “Active Contour Based Segmentation Techniques for Medical Image Analysis.” *Licensee IntechOpen*, pp. 17–33, 2018. <https://doi.org/10.5772/intechopen.74576>
- [14] B. Li and P. Xu, “AMachine Vision-Based Multifunctional Image Processing Platform.” Springer Nature Singapore, Shanghai, China, pp. 121–130, 2019.
- [15] A. Lazaro, J. L. Buliali, and B. Amaliah, “Deteksi Jenis Kendaraan di Jalan Menggunakan OpenCV.” *Jurnal Teknik ITS, Surabaya*, pp. 293–298, 2017. <https://doi.org/10.12962/j23373539.v6i2.23175>
- [16] J. Howse, *Android Application Programming with Ppencv*. Birmingham: Packt Publishing, 2013.
- [17] J. F. Fauzi, H. Tolle, and R. K. Dewi, “Implementasi Metode RGB To HSV pada Aplikasi Pengenalan Mata Uang Kertas Berbasis Android untuk Tuna Netra,” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 2, no. 6. pp. 2319–2325, 2018. <https://doi.org/10.14710/jtsiskom.4.2.2016.305-314>
- [18] M. R. Kumaseh, L. Luther, and N. Nainggolan, “Segmentasi Citra Digital Ikan Menggunakan Metode Thresholding.” *Manado*, pp. 74–79, 2013. <https://doi.org/10.35799/jis.13.1.2013.2057>
- [19] B. Y. Budi Putranto, W. Hapsari, and K. Wijana, “Segmentasi Warna Citra Dengan Deteksi Warna Hsv Untuk Mendeteksi Objek,” *J. Inform.*, vol. 6, no. 2, pp. 1–14, 2011. <https://doi.org/10.21460/inf.2010.62.81>
- [20] E. H. Yossya, J. Pranata, T. Wijaya, H. Hermawan, and W. Budiharto, “Mango Fruit Sortation System using Neural Network and Computer Vision,” *Procedia Comput. Sci.* 116, vol. 116, pp. 569–603, 2017. <https://doi.org/10.1016/j.procs.2017.10.013>
- [21] M. H. Fauzi and H. Tjandrasa, “Implementasi Thresholding Citra Menggunakan Algoritma Hybrid Optimal Estimation,” *Teknologi Informasi. Institut Teknologi Sepuluh Nopember, Surabaya*, 2010. <https://doi.org/10.24089/j.sisfo.2014.03.011>
- [22] U. H. Dewi, B. Hidayat, and E. Yuni, “Estimasi Bobot Sapi Berdasarkan Registrasi Citra Digital Dengan Metode Fraktal Dan Klasifikasi K-Nearest Neighbor (KNN),” *e-Proceeding Eng.*, vol. 6, no. 1, pp. 697–704, 2019.

## 7 Authors

**Akhmad Qashlim** is the Chairperson of the Indonesian Association of Higher Education in Informatics and Computing, West Sulawesi Province. He received an M.Kom degree in Information Systems from Diponegoro of University, Semarang, Central Java, Indonesia. Currently, he works as a lecturer at Al Asyariah Mandar University, Polewali Mandar, West Sulawesi, Indonesia. His research interests are in the areas of Image Processing, System Information, and Human and Computer Interaction. Email: [qashlim@unasman.ac.id](mailto:qashlim@unasman.ac.id)

**Basri** was born on July 05, 1987 His received a Master (MT) in 2015 in the Electrical Engineering Department, Faculty of Engineering, Hasanuddin University, Indonesia. Currently, he works as a lecturer at Al Asyariah Mandar University, Polewali Mandar, West Sulawesi, Indonesia. His research interests are in the areas of Image Processing and Artificial Intelligence, Data mining.

**Haeruddin** was born on January 19, 1990. He is a lecturer of Agribusiness Department, Faculty of Agriculture, Al Asyariah Mandar University, West Sulawesi, Indonesia. His research interests are in the areas of supply chain management, industry technology, fishing technology.

**Ardan** is with the Electrical Engineering Department, Faculty of Engineering, Hasanuddin University, Makassar, Indonesia. Currently, her works as a lecturer at the Electrical Engineering Department, Faculty of Engineering, Hasanuddin University, Makassar, South Sulawesi, Indonesia. Her research interests are in the areas of Image Processing, bio information and computer vision.

**Ingrid Nurtanio** with the Electrical Engineering Department, Faculty of Engineering, Hasanuddin University, Makassar, Indonesia. He works as a lecturer at the Electrical Engineering Department, Faculty of Engineering, Hasanuddin University, South Sulawesi, Indonesia. His research interests are in the areas of Cloud Computing, Computer System, and Information System.

**Amil Ahmad Ilham** received an S. Kom degree in Information Systems from Al Asyariah Mandar University, Polewali Mandar, Indonesia. Currently Assistant Lecturer at the Faculty of Computer Science for the field of website programming and artificial intelligence in addition to being active in the Artificial Intelligence research group and information systems.

Article submitted 2019-11-18. Resubmitted 2019-12-23. Final acceptance 2019-12-25. Final version published as submitted by the authors.