Object Detection System to Help Navigating Visual Impairments

Cahya Rahmad[1], Rawansyah[3], Tanggon Kalbu[4]
Department of Information Technology
State Polytechnic of Malang, Malang, Indonesia

Kohei Arai[2]
Department of Information Technology
Saga University, Saga City, Japan

Abstract—The number of people with severe visual impairments and blind people in the world is 216.6 million and 38.5 million, respectively in 2018 and that number will increase every year. While the development of Computer Vision technology became popular after that method is used in automatic driving system using an object detection system to detect surrounding object, this technology can be a solution to help blind people too. This can be done by implementing Harris Corner Detection method. Harris Corner Detection method is used to detect the corner of the object in the image taken. The number of corner and location of corner in detection result can be used for predicting position and distance. To predict the distance, a triangle rule will be used in finding the distance. Furthermore, it can predict the location and distance of the object in the picture taken. From the results of the implementation above it was found that the accuracy of object detection using Harris Corner Detector's angle detection method is 88%. Therefore, this application can help detecting objects based on the number of corner and location detected using a Smartphone.

Keywords—Object detection; corner detection; computer vision; visual impairments; blind people

I. Introduction

Computer Vision is a technology that aims to mimic the function of human vision electronically [1]. Computer Vision relies a lot in feature extraction [2]. Feature extraction is the initial stage of object detection [3]. This technology works by taking features in the image. Images have many features such as lines, contours, angles and colors. These features are very important in the field of image processing and computer vision [4]. The example of feature detection methods are Harris Detector, Canny, Laplacian of Gaussian, and Sobel. In the past few years, this system is used in a cars as a tool to navigate when driving [5]. Using edge detector to detect road markings as guide line for driving navigation and corner detector to detect traffic signs [6].

This system also could be useful especially for people who have visual impairments. This system could replace human eye because its function that resemble human vision. Statistics says that there are 216.6 million people who have visual impairments and as many as 38.5 million people who suffer from total blindness are predicted to increase each year [7]. That is the challenge that we must face. To maintain the safety of the community and the people that have visual impairments, access to visual information about surrounding object is necessary.

Apart from existing methods to help such as personal assistants who help people with visual impairments, an intelligent system needs to be developed that can automatically help people with visual impairments carry out their daily activities to help navigate and detect objects around them [8]. So from that it is necessary to create a system that detects objects around it on the android platform. By using the Harris Corner Detection method that will be implemented on a smartphone, users can retrieve information about object around them even if the user has a visual impairment.

Quite recently, E. Rosten and T. Drummond proposed "Machine learning for high-speed corner detection," in the European Conference on Computer Vision [9]. Also, Leonardo Trujillo and Gustavo Olague reported: "Automated design of image operators that detect interest points" [10]. Meanwhile, Ivan Laptev and Tony Lindeberg proposed "Space-time interest points" [11]. On the other hand, Geert Willems, Tinne Tuytelaars and Luc van Gool proposed "An efficient dense and scale-invariant spatiotemporal-temporal interest point detector" [12]. Meantime, Tony Lindeberg proposed "Spatio-temporal scale selection in video data" [13]. Also, I. Everts, J. van Gemert and T. Gevers proposed "Evaluation of color spatio-temporal interest points for human action recognition" [14]. These are alternatives of the proposed Harris Cornet Detection Algorithm.

By utilizing a smartphone camera, the user can press volume button on the headset or earphone that has been programmed to take pictures with back camera. The captured image will be processed using the Harris Corner Detection method to get information in the form of text about what is caught on the camera. The information is then converted into audio so that users can listen to the location and distance of the object in front of it. So that it is expected to help to detect objects around it and help navigate its users.

II. Harris Corner Detector

Harris Corner Detection is a method that aims to take and detect corner on an image. This corner extraction feature is the initial stage of detection of this object. Corner detection is used to detect the angle of an object in the image. So that it functions to detect objects that have angles such as doors, the lower end of the wall, etc.

This method works by considering 8 pixels next to a pixel that will detect the angle. For example pixel p is a pixel whose corner will be detected. If the intensity of 8 pixels next to pixel p is almost the same as pixel p, then it is a flat region. If there is a significant change but there is no change in the surrounding pixels in the same direction, it is called the edge region. If there is a significant change in each direction, the pixel is a corner. Fig. 1 is the following illustration of flat, edge, and corner region.

To achieve Harris Value in every pixel in an image, this following steps must be taken [2]:

- Grayscaling the image. Give a grayscale effect on the original image. This aims to reduce the number of matrix dimensions from 3 x 3 to 2 x 2 in order to simplify calculations [15].
- Smoothing the image. Using blur block to convolute the grayscale image. This also reduce noise and unwanted corner.
- Computing gradient of the image using convolution between grayscaled and blurred image with gradient masks.
- Computing autocorrelation matrix.
- Computing eigen value.
- Computing corner strength.

A. Image Gradient

In order to get the gradient difference between 8 neighbor pixel, Harris and Stephens [16], using central difference for gradient masks. Central difference will be convoluted with grayscaled and blurred image. Central difference kernel can be seen in Fig. 2. While the horizontal kernel is Ix and vertical kernel is Iy.

B. Autocorrelation Matrix

The product of derivatives (image gradient) Ix and Iy will be inserted to A, B, and C variable. This variable is used for matrix M. In which A is Ix^2, B is IxIy, and C is Iy^2. Therefore matrix M will be written like in Equation (1).

$$M = \begin{bmatrix} I_x I_x & I_x I_y \\ I_x I_y & I_y I_y \end{bmatrix} = \begin{bmatrix} A & B \\ B & C \end{bmatrix} \quad (1)$$

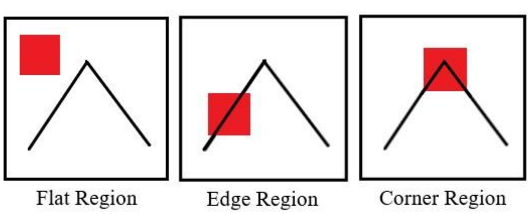

Fig. 1. Illustration of Flat, Edge, and Corner Region.

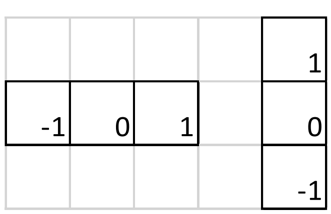

Fig. 2. Central difference Gradient Masks.

C. Eigen Value

The eigen value is symbolized by the lambda symbol (λ) which is important step when looking for corner. Eigen Value in Harris Corner Detector method is used to find corner from intersection edges that are not horizontal or vertical.

$$\det M = \lambda_1 \lambda_2 = AC - BB \quad (2)$$

$$trace\ M = \lambda_1 + \lambda_2 = A + C \quad (3)$$

$$\lambda = \frac{traceM \pm \sqrt{traceM^2 - 4detM}}{2} \quad (4)$$

D. Harris Value

Harris value can be symbolized by R. Which R equal in equation (5). With k value is 0.04. What define a pixel is a corner, edge, or flat region is based on the product of R. If R <0 means $\lambda_1 > \lambda_2$ or vice versa then the area is classified as the edge region. If R> 0 and the score is high then λ_1 and λ_2 are high, then the area is classified as a corner region. If R is low, then it is a flat region. There is no fixed value of how low is R classified as flat region, so it is a personal preference about how low r can be classified as flat region. In Fig. 3 is a classification how to define flat, edge, or corner by how much R is.

$$R = detM - k(traceM)^2 \quad (5)$$

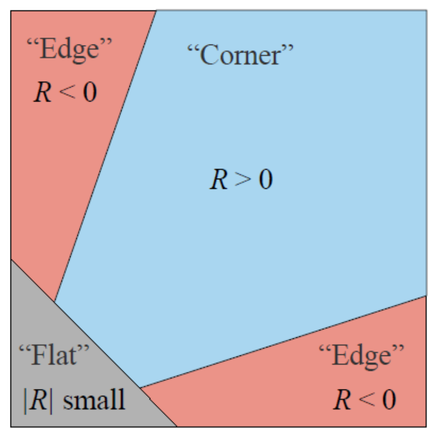

Fig. 3. R Classification Illustration.

III. Triangle Rule

It is a simple method on triangles that can be used as a guide to find distance. Fig. 4 is an illustration of a person who uses a smartphone. Variable a is the distance between the land and the smartphone. This distance can be obtained from 83% of the user's height. Variable b is the prediction of the distance between the user and the object that was detected. Variable c is the direction of the camera's angle of view. While the angle θ is the angle formed by the point of view of the camera with the distance from the ground to the smartphone. Variables Distance 1, Distance 2, and Distance 3 are three groups of distance predictions. Object A is in the distance group 1, Object B is in the distance group 2, and Object C is in the distance group 3.

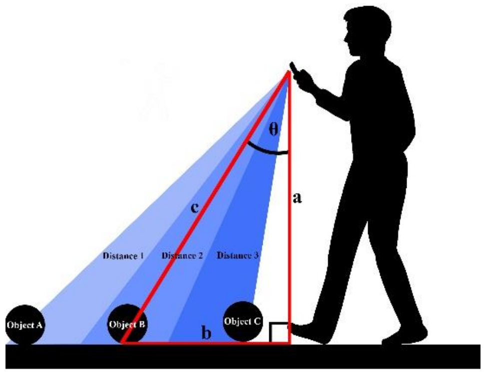

Fig. 4. Triangle Illustration

If the user tilts his smartphone to the point of view of the camera forming an angle θ then the value of b can be calculated which is the prediction of the distance between the user and Object B using Equation (6). If distance of Object B has been found, the distance of Object A is further from Object B. The distance of Object C is closer than the distance of Object B.

$$b = a * \tan\theta \qquad (6)$$

IV. Measuring Distance and Position

In Fig. 5, the object distance determination is done by dividing the image into nine equal parts.

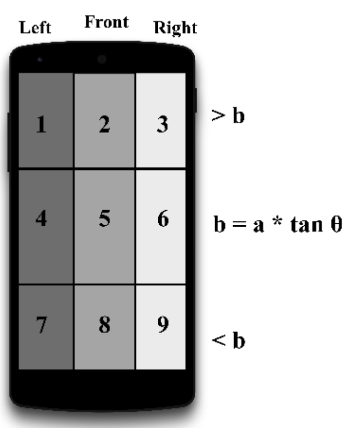

Fig. 5. Corner Position and Distance by Image.

The division is done based on the length and width of the image. Each width and height divides by 3 equally. The distance is determined using the triangle rule as in equation (6). The angle taken has a range between 40-60 degrees. Determination of the angle is done using an accelerometer.

V. Implementation

In Fig. 6, it is explained that the application starts by declaration the required variables. The y variable is used to store the y value which is related to the slope of the smartphone. The ACL variable is a variable to store the type of sensor that will be used and the variable v stores the vibrating function that is on the smartphone.

After that the application waits for variable input from OnKeyDown. If you press the volume down button, the application will continue. If the volume is not down, then it is volume up. Volume up button is used for entering user heights. It can be done by saying the default height value, which is 150 centimeters. In height settings, up and down volume button is for adjusting the user's height by 10 centimeters. When height is fixed, user shall press volume up button for 3 seconds and out from height settings. When out of the height settings, the volume down button have function to retrieve the y value from the accelerometer sensor. This value is a consideration for determining the camera's angle of view based on the tilt of the smartphone. If it does not meet the angle range of 40 to 50, the smartphone will vibrate.

Angles 40 to 50 if scaled with a value of y then the angle of 40 is equivalent to 4.4 and the angle of 50 is equivalent to 5.5. If the angle is within that range, the application will stop the vibration using v.cancel(). After the vibration stops, the application will automatically take pictures, calculate the HCD value, determine the position and distance based on the HCD value, and voice it. The time needed from the four processes is around 1 to 2 seconds. The faster the information variables that have to be voiced are few and the longer the more information that has to be voiced.

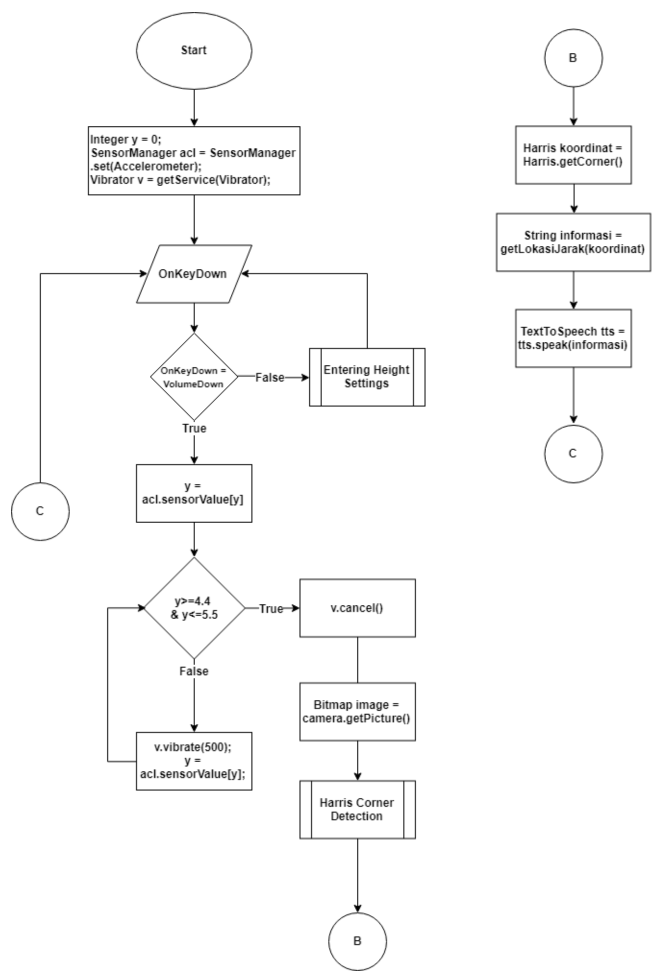

Fig. 6. Full Application Flowchart.

VI. Testing

Testing is done by taking pictures 50 times in each area. After that, checking the truth using human vision with normal vision to find out whether the location and distance variables of the object in the application are true. Accuracy can be calculated by adding up the number of correct predictions then divided by the total number of predictions. Image testing can be accessed in this references [17]. Example of testing images is shown in Fig. 7.

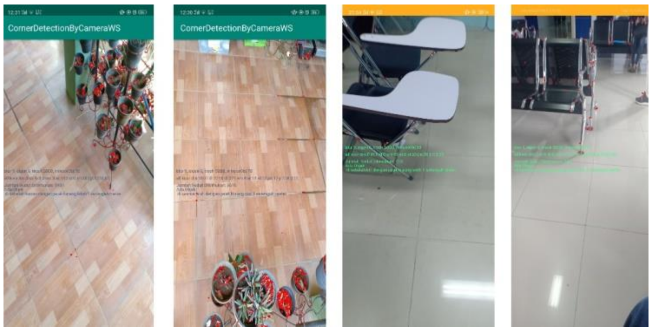

Fig. 7. Example of Testing Images.

After testing, obtained an accuracy of 88%. This number is taken from 44 times true and 6 times false. The prediction will be wrong if there is a lot of noise and unnecessary patterns on the floor. This results in misinformation. The application detects the pattern of the floor as an object. Floors or environments that have many holes or are uneven or have a certain pattern can cause the object to be detected. Even though there are no objects in front that gets in the way.

VII. Conclusion

- Detecting objects using the HCD method on a smartphone can be done with the processing time from taking pictures until finished talking for 1 to 2 seconds.
- Predicting the distance of the object and providing information about the position of the object can be done using the rules of the triangle. By utilizing the hegiht of the user as b and tan the angle of 45 to 60 to get 3 distance categories b meters, more than b meters and less than b meters.
- Based on testing with real objects, obtained an 88% of accuracy with the correct number of predictions 44 and 6 wrong.

References

[1] S. Milan and H. Vaclav, Image Processing, Analysis, and Machine Vision. Thomson, 2008.

[2] Sánchez, N. Monzón, and A. Salgado, “An Analysis and Implementation of the Harris Corner Detector,” Image Process. Line, 2018.

[3] K. Jeong and H. Moon, “Object detection using FAST corner detector based on smartphone platforms,” in Proceedings - 1st ACIS/JNU International Conference on Computers, Networks, Systems, and Industrial Engineering, CNSI 2011, 2011.

[4] Y. Li, S. Wang, Q. Tian, and X. Ding, “A survey of recent advances in visual feature detection,” Neurocomputing, 2015.

[5] R. Gandhi, “R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms,” Towards Data Science, 2018. [Online]. Available: https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e. [Accessed: 04-Dec-2018].

[6] T. Wenzel, T. W. Chou, S. Brueggert, and J. Denzler, “From corners to rectangles-Directional road sign detection using learned corner representations,” in IEEE Intelligent Vehicles Symposium, Proceedings, 2017.

[7] R. R. A. Bourne et al., “Magnitude, temporal trends, and projections of the global prevalence of blindness and distance and near vision impairment: a systematic review and meta-analysis.,” Lancet. Glob. Heal., vol. 5, no. 9, pp. e888–e897, 2017.

[8] F. S. Bashiri, E. LaRose, J. C. Badger, R. M. D’Souza, Z. Yu, and P. Peissig, “Object detection to assist visually impaired people: A deep neural network adventure,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2018.

[9] E. Rosten and T. Drummond (May 2006). "Machine learning for high-speed corner detection,". European Conference on Computer Vision.

[10] Leonardo Trujillo and Gustavo Olague (2008). "Automated design of image operators that detect interest points" (PDF). Evolutionary Computation. 16 (4): 483–507. doi:10.1162/evco.2008.16.4.483. PMID 19053496. Archived from the original (PDF) on 2011-07-17.

[11] Ivan Laptev and Tony Lindeberg (2003). "Space-time interest points". International Conference on Computer Vision. IEEE. pp. 432–439.

[12] Geert Willems, Tinne Tuytelaars and Luc van Gool (2008). "An efficient dense and scale-invariant spatiotemporal-temporal interest point detector". European Conference on Computer Vision. Springer Lecture Notes in Computer Science. 5303. pp. 650–663. doi:10.1007/978-3-540-88688-4_48.

[13] Tony Lindeberg (2018). "Spatio-temporal scale selection in video data". Journal of Mathematical Imaging and Vision. 60 (4). pp. 525–562. doi:10.1007/s10851-017-0766-9.

[14] I. Everts, J. van Gemert and T. Gevers (2014). "Evaluation of color spatio-temporal interest points for human action recognition". IEEE Transactions on Image Processing. 23 (4). pp. 1569–1589.

[15] P. Ram and S. Padmavathi, “Analysis of Harris corner detection for color images,” in International Conference on Signal Processing, Communication, Power and Embedded System, SCOPES 2016 - Proceedings, 2017.

[16] C. Harris and M. Stephens, “A Combined Corner and Edge Detector,” 2013.

[17] Cahya Rahmad Image Testing. [Online]. Available: http://bit.ly/CahyaRahmadImageTesting.

Authors’ Profile

Cahya Rahmad: He received BS degrees from Brawijaya University Indonesia in 1998 and MS degrees from Informatics engineering at Sepuluh Nopember Institute of Technology Surabaya Indonesia in 2005. He is a lecturer in The State Polytechnic of Malang Since 2005. He received Doctoral degrees at Saga University japan in 2013, His interest researches are image processing, data mining and patterns recognition.

Kohei Arai, He received BS, MS and PhD degrees in 1972, 1974 and 1982,respectively.He was with The Institute for Industrial Science and Technology of the University of Tokyo from April 1974 to December 1978 also was with National Space Development Agency of Japan from January, 1979 to March, 1990. During from 1985 to 1987, he was with Canada Centre for Remote Sensing as a Post-Doctoral Fellow of National Science and Engineering Research Council of Canada. He moved to Saga University as a Professor in Department of Information Science on April 1990. He was a counselor for the Aeronautics and Space related to the Technology Committee of the Ministry of Science and Technology during from 1998 to 2000. He was a councilor of Saga University for 2002 and 2003. He also was an executive councilor for the Remote Sensing Society of Japan for 2003 to 2005. He is an Adjunct Professor of University of Arizona, USA since 1998. He also is Vice Chairman of the Science Commission “A” of ICSU/COSPAR since 2008 then he is now award committee member of ICSU/COSPAR. He wrote 37 books and published 570 journal papers. He received 30 of awards including ICSU/COSPAR Vikram Sarabhai Medal in 2016, and Science award of Ministry of Mister of Education of Japan in 2015. He is now Editor-in-Chief of IJACSA and IJISA. http://teagis.ip.is.saga-u.ac.jp/index.html