Face Recognition Using Local Binary Pattern Histogram for Visually Impaired People

Vestiana Aza
Postgraduate Student of Electrical Engineering Department
Hasanuddin University
Makassar, Indonesia
vestiana.aza18@gmail.com

Indrabayu
Informatics Department
Hasanuddin University
Makassar, Indonesia
indrabayu@unhas.ac.id

Intan Sari Areni
Electrical Engineering Department
Hasanuddin University
Makassar, Indonesia
intan@unhas.ac.id

***Abstract*— This study aims to build a real-time system that can help visually impaired to recognize people present in their surroundings. Visually impaired people experience many difficulties in daily activities, especially to recognize people. The proposed system applies the concept of a mobile personal assistant that runs in Android smartphone. The input data is a real-time video taken using 16-megapixel smartphone camera with a resolution of 1280x720. The system built uses Gamma Correction and Difference of Gaussian methods to improve the accuracy of the Local Binary Pattern Histogram (LBPH) algorithm. Based on the experiment with 148 cm and 172 cm user heights and 8 respondents, the system was able to increase the accuracy by 8.32% for 148 cm height, while the accuracy for 172 cm height increased by 6.22%. The output of this system is the sound of the identification result.**

Keywords—visually impaired people, Haar-Cascade Classifier, LBPH, Gamma Correction, Difference of Gaussian, face recognition

I. Introduction

Face recognition is considered as the most important issue because face plays an important role in recognizing people in social life [1]. Because of the visual impairment, visually impaired people experience many difficulties in daily activities. One of them is recognizing people in their surroundings. The face has many features that can be used to identify the expression of a person, for example, to detect drowsiness using eye feature [2]. Therefore, it is possible to recognize face using its features. Face recognition technology can help visually impaired people to identify people and start the conversation by greeting other people first like normal people. Therefore, visually impaired people will feel more confident.

Research conducted by Kramer et al. in 2010 proposed a face recognition prototype based on remote servers by utilizing smartphone technology for visually impaired people using Verilook technology. The proposed face recognition technology can tolerate up to a 40 degrees angle between the direction a person is looking and the camera's axis. However, the system still used wireless networks to send images from a smartphone and performed face recognition in a server [3].

Chilaron & Dunai in 2015 offered a real-time face detection and recognition system for visually impaired people using boosted cascade for face detection and eigenface for face recognition. The system was developed on Raspberry [4]. Jin et al. in the same year provided a face recognition system using smart cane and glasses to help visually impaired people in recognizing human faces. Recognized results from the mobile computer are sent to the microcontroller on the cane via Bluetooth communication. Next, the microcontroller generates a vibration pattern using a vibration motor according to the result of face recognition and identify people through those vibration patterns. However, the user has to recall the vibration pattern for each person. Consequently, it's hard to remember the patterns of many people [5].

Furthermore, research conducted by Neto et al. built a real-time face recognition system using smartwatch [6]. Two years later, in 2017, Neto et al. developed a real-time face recognition system using a Microsoft Kinect sensor as a wearable device to help visually impaired and low-vision people [7].

The related research provided solutions by creating a system that can detect and recognize faces for visually impaired people. The system still uses several devices and that's not convenient for the users. Therefore, this research proposes a system that can help visually impaired people to recognize a person in daily activities by applying a mobile personal assistant concept in android smartphone. This personal assistant is applied to android smartphones so they are more practical in helping visually impaired people to recognize people around them so that visually impaired people can socialize comfortably.

II. Literature Review

A. Haar-Cascade Classifier

OpenCV face detector uses the Viola-Jones method with Haar-Cascade Classifier. Some of the parameters used by OpenCV are image, objects, scaleFactor, minNeighbors, flags, minSize, and maxSize. Image is a matrix of the type CV_8U containing an image where objects are detected. Objects are rectangles vector where each rectangle contains the detected object. ScaleFactor is a parameter that determines how much the image size is reduced at each image scale. MinNeighbors, a parameter is specifying how many neighbors each candidate rectangle should have to retain it. The old cascade format only uses parameter Flags. MinSize is the minimum possible size of the object. Objects that are smaller than that are ignored. MaxSize is the maximum possible size of the object. Objects that are larger than that are ignored [8].

The Haar-Cascade Classifier algorithm uses a rectangle-shaped sliding window that slides across an image from pixel to pixel. From this rectangle window, Haar-like features will be detected. After the sliding window ends, the size of the rectangle-shaped sliding window will be reduced (resize) and it will begin to scan the image pixel by pixel as before. This process will be repeated until the size of the sliding window cannot be reduced anymore. Hence, many features will be detected many times. Haar uses a grayscale

978-1-7281-3832-9/19/$31.00 ©2019 IEEE

image. Therefore, the RGB frame is converted into grayscale.

B. Local Binary Pattern Histogram

The Local Binary Pattern Histogram (LBPH) is a face recognition algorithm based on the local binary operators proposed in 2006. LBPH is a popular algorithm and widely used because of its discriminative power and simple computation. NxM sized image is used and split into several regions. The same size is recommended in both width and height to obtain MxM regions. In each region, the local binary operator is used by comparing a center pixel of the image with its eight closest neighbor pixels. If the value of the neighbor pixel is higher or equal than the center pixel, then the value is "1", On the contrary, returning a value of '0'. By applying this process to all eight neighbors, eight binary values are derived. Then the 8-bit binary number is formed by merging those values and converted it into a decimal value called the pixel LBP value in the range between 0-255 [9]. The conversion process of a grayscale image to LBP is illustrated in Fig. 1.

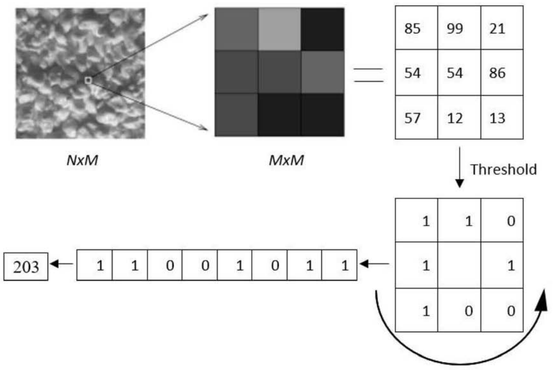


Fig. 1. The basic LBP operator [10]

There are four parameters in the LBPH algorithm, which are Radius, Neighbors, Grid X, and Grid Y. Radius is used to build a Circular Local Binary Pattern and represent the radius around the central pixel. The value of the radius is set to 1. "Neighbors" is the number of sample points to form a Circular Local Binary Pattern. An appropriate value to use is 8 sample points. Grid X is the number of cells in the horizontal direction. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector. It is normally set to 8. And Grid Y is the number of cells in the vertical direction, 8 is a common value used in publications. The more cells, the finer the grid, the higher the dimensionality of the resulting feature vector [11].

Based on Fig. 1, the histogram of each region is extracted using Grid X and Grid Y parameters to divide the image into several grids. The grayscale histogram image in each grid will only contain 256 positions (0-255) which represent the occurrences of each pixel intensity. Then, each histogram is concatenated into a new bigger histogram. The last histogram represents the characteristics of the original image.

C. Gamma Correction

Gamma Correction is a nonlinear gray level transformation that replaces gray level I with $I^{1/\gamma}$ (for $\gamma > 0$) or log (I) (for $\gamma = 0$), where $\in [0, 1]$ is a parameter specified by the user. This process enhances the local dynamic range of image in dark or shadowed areas and compresses it in bright areas [12].

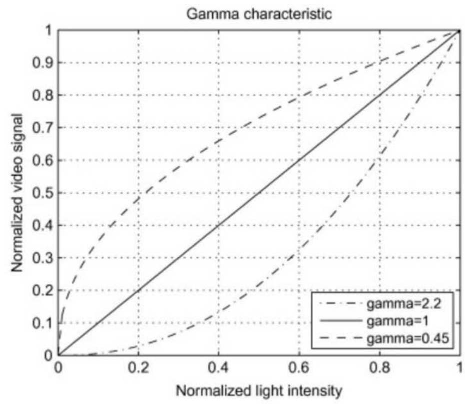


Fig. 2. Curve gamma γ= {0.45,1,2.2} [13]

In Fig. 2, the range of pixel from (0,0) represents black and (1,0) represents white. The curve shows that the black part of the image remains black and the white part of the white image remains white, while the area between the two are seamlessly adjusted. Hence, the overall color of an image depends on the gamma value, whether the image would be brighter or darker, by maintaining the dynamic range of the image [14]. When the value of γ is less than one, the converted image would become brighter than the original image; and when it is greater than one, the converted image would become darker than the original image [15].

D. Difference of Gaussian

Difference of Gaussian is obtained by subtracting two Gaussian kernels, where a kernel has a standard deviation smaller than the previous one [16].

DoG (Difference of Gaussian) is used to simplify the process of finding keypoints by subtracting images in one octave. Fig. 3 shows an illustration of the subtraction process between each image in one octave.

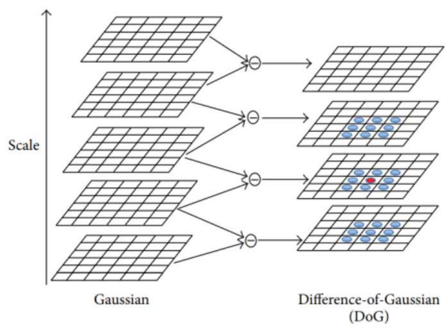


Fig. 3. *Difference of Gaussian* process

The next process is to find the local extrema values by comparing a sample pixel with each of its eight neighbors to find out whether the sample pixel is minima or maxima to the neighbor pixels. In order to achieve good accuracy, the comparison can be done by comparing sample pixels with neighbors above and below them. A point will be chosen as the keypoint candidate if the point value is smaller or larger than all of its 26 neighbors. Fig. 4 shows the process to find the local extrema value.

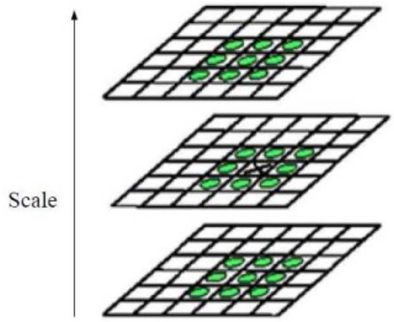


Fig. 4. Process to find the local extrema value

After obtaining keypoint candidates from the previous process, the next step is to eliminate unstable keypoint candidates. Usually, an unstable keypoint is a keypoint that has a different value from the surrounding value on the matrix which might be caused by noise. An unstable candidate can also be a candidate that has low contrast or is at the edge of the image [17].

III. Proposed Method

The steps of the research process are illustrated in Fig. 5.

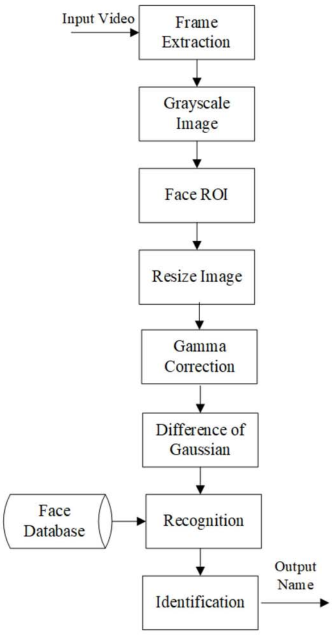


Fig. 5. Block diagram of the research

The input data is a real-time video taken using 16-megapixel smartphone camera with a resolution of 1280x720. Input data are face images from 8 respondents and stored in the database. Samples of face images from 8 respondents can be seen in Fig. 6.

Fig. 6. Samples of face images

Point of view camera or povie is used as the smartphone holder and placed on the neck of the blind with predetermined angles, 25 degrees for user with 148 cm height and 15 degrees for user with 172 cm height. The illustration can be seen in Fig. 7.

Data collection process using povie is based on the range of people's height. The selection of 148 cm height is to represent the short people and 172 cm is to represent the tall people.

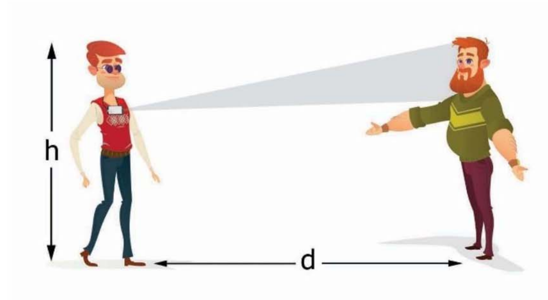


Fig. 7. Illustration of system design

The input data is extracted per frame and converted to a grayscale image, as shown in Fig. 8 dan Fig. 9.

Fig. 8. The example of RGB frame

Fig. 9. The example of Grayscale frame

Face detection uses a grayscale image parameter because Haar cascade detector detects faces based on grayscale images. According to the analysis in [18] to detect faster and real-time video, the ScaleFactor parameter is set to 1.2. Haar-Cascade Classifier's parameters used in this study for detecting faces can be seen in Table I.

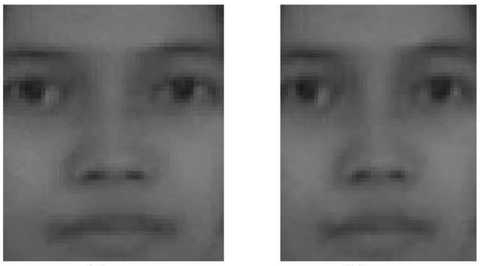
TABLE I. PARAMETERS IN HAAR-CASCADE CLASSIFIER

Parameter	Value
ScaleFactor	1.2
minNeighbors	5
Flags	2

When a face is detected, a bounding box will be generated around the face, as shown in Fig. 10.

Fig. 10. Result of face detection

The next process is to determine the ROI (Region of Interest) of the face. By determining the ROI, certain parts can be chosen and the next process can be more accurate. Determination of face ROI based on the result of cropped face detection can be seen in Fig. 11.



(a) (b)

Fig. 11. (a) Face ROI; (b) Resize image

The image of face ROI is resized to uniform the image size, as shown in Fig. 11. In this stage, the size of the face image is set to 110x128 pixel to reduce the execution time of the recognition process without losing relevant information of the image and the face recognition process is more convenient.

After that, the illumination normalization process is performed on the resized images to uniform the lighting conditions that vary from the test image and database image. This is done to get better recognition result. The illumination normalization process is carried out using 0.2 gamma value based on research conducted by Tan and Triggs [12] and difference of gaussian with a radius of 1 and 23.

After illumination normalization process, the next step is face recognition process using the LBPH algorithm with the default values of Radius, Neighbors, Grid X, and Grid Y parameters which are 1, 8, 8, 8. Then threshold is used to estimate the face recognition results of the algorithm. The result of face recognition is shown in Fig.12. The output of this system is the sound of the identification result.

Fig. 12. Face recognition result

IV. RESULT AND DISCUSSION

Experimentation was carried out to show the effect of the normalization of illumination on improving the level of accuracy. It was performed in the afternoon and in a room without lights. As for each scenario, 8 respondents moved toward the user from a distance of 2.5 meters to 1 meter. The testing was carried out in 10 trials with 8 respondents and 5 different threshold values, i.e., 70, 75, 80, 85 dan 90.

The determination of the threshold values based on pre-trial results shows lower accuracy using threshold below 70, meanwhile threshold above 90 does not indicate a change in accuracy. Therefore, to obtain optimal results, a test using threshold values between 70 and 90 is performed. The average test resulted from 10 trials without normalization of illumination (*Wn*) and with normalization of illumination (*n*) is shown in Table II.

TABLE II. TEST RESULT

Height (cm)	Illumination Normalization	Accuracy (%)				
		70	*75*	*80*	*85*	*90*
148	*Wn*	85.57	85.57	86.09	87.2	87.2
	n	94..65	94.65	94.65	94.65	94.65
172	*Wn*	86.56	87.74	88.46	88.46	88.46
	n	94.01	94.1	94.1	94.1	94.48

The test result in Table II shows that the highest accuracy for 148 cm height is 87.2% using threshold values of 85 and 90, while the highest accuracy for 172 cm height is 88.46%, obtained using threshold values of 80, 85 and 90. The result obtained due to variations in lighting in the test images and the database images. Therefore, illumination normalization process was carried out before the recognition stage using gamma correction and difference of gaussian filtering to improve recognition accuracy.

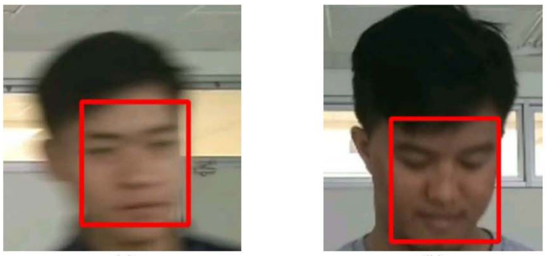
The result after illumination normalization using gamma correction and difference of gaussian shows an increase in the average accuracy of 8.32% from 5 thresholds for 148 cm height and 6.22% for 172 cm height. These are caused by the illumination normalization process that uniforms lighting conditions, so that face images with large illumination variations become relatively equal with the illumination of other facial images. Gamma correction is used to adjust image contrast. The gamma value of 0.2 is chosen to lower the contrast, so the resulting image does not have a significant difference between light and dark areas and between the test image and the database image.

Due to low contrast produced by gamma correction, the difference of gaussian is needed to determine the feature (keypoint) on the object by marking the part that becomes the focus of the image. Sample images before and after the normalization of illumination with gamma correction and difference of gaussian can be seen in Fig. 13.

(a) (b)

Fig. 13. (a) Image before illumination normalization; (b) Image after illumination normalization

Several conditions cause the recognition error in this system, i.e., the rapid movement of respondents causing the camera tends to capture blur images and the position of the respondent's face that is not facing the camera. The examples of the recognition error are shown in Fig. 14.



(a) (b)

Fig. 14. Examples of recognition error

V. Conclusions

This real-time face recognition system for visually impaired people runs in Android smartphone and gives the highest accuracy of 87.2% out of 10 trials, experimented using threshold value of 85 and 90. While, the best accuracy for 172 cm height scenario is 88.46%, obtained with threshold values of 80, 85 and 90. The proposed system with the illumination normalization process using gamma correction and difference of gaussian can increase accuracy by 8.32% in 148 cm height scenario, while the accuracy in 172 cm height scenario increases by 6.22%. However, this system can only recognize one face. For future work, the system can be developed to recognize more than one face at one time.

References

[1] S. Sultoni and R. Hariyanto, "Real Time Face Recognition Using the Camshift Method and Erosion Operator Based on Face Image," *JICTE J. Inf. Comput. Technol. Educ.*, vol. 1, p. 8, Nov. 2017.

[2] I. Amirullah, R. Anriani Tacok, and I. Areni, "Modification on Brightness Enhancement for Simple Thresholding in Eyelid Area Measurement," 2017, pp. 101–104.

[3] K. M. Kramer, D. S. Hedin, and D. J. Rolkosky, "Smartphone based face recognition tool for the blind," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, 2010, pp. 4538–4541.

[4] M. Chilaron and L. Dunai, "Novel face recognition system for blind people," 2015.

[5] Y. Jin, J. Kim, B. Kim, R. Mallipeddi, and M. Lee, "Smart Cane: Face Recognition System for Blind," 2015.

[6] L. Britto Neto, V. Maike, F. Koch, M. Cecília Calani Baranauskas, A. De Rezende Rocha, and S. Klein Goldenstein, "A Wearable Face Recognition System Built into a Smartwatch and the Visually Impaired User," in *ICEIS 2015 - 17th International Conference on Enterprise Information Systems, Proceedings*, 2015, vol. 3.

[7] L. B. Neto *et al.*, "A Kinect-Based Wearable Face Recognition System to Aid Visually Impaired Users," *IEEE Trans. Hum.-Mach. Syst.*, vol. 47, no. 1, pp. 52–64, Feb. 2017.

[8] "Cascade Classification — OpenCV 2.4.13.7 documentation." [Online]. Available: https://docs.opencv.org/2.4/modules/objdetect/doc/cascade_classification.html#cascadeclassifier-detectmultiscale. [Accessed: 21-Apr-2019].

[9] N. Stekas and D. v d Heuvel, "Face Recognition Using Local Binary Patterns Histograms (LBPH) on an FPGA-Based System on Chip (SoC)," in *2016 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, 2016, pp. 300–304.

[10] S. Singh, G. Mohiuddin, and S. Pandit, "Face Detection and Recognition by Haar Cascade Classifier, Eigen Face and LBP Histogram," p. 4, Feb. 2019.

[11] T. Ahonen, A. Hadid, and M. Pietikainen, "Face Description with Local Binary Patterns: Application to Face Recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.

[12] X. Tan and B. Triggs, "Enhanced Local Texture Feature Sets for Face Recognition Under Difficult Lighting Conditions," *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1635–1650, Jun. 2010.

[13] "Gamma Correction - an overview | ScienceDirect Topics." [Online]. Available: https://www.sciencedirect.com/topics/engineering/gamma-correction. [Accessed: 13-Jul-2019].

[14] A. Satish and N. Devarajan, "Preprocessing technique for face recognition applications under varying illumination conditions," *Glob. J. Comput. Sci. Technol. Graph. Vis.*, vol. 12, pp. 13–18, Jan. 2012.

[15] Sumathi, "Automatic Text Extraction from Complex Colored Images Using Gamma Correction Method," *J. Comput. Sci.*, vol. 10, no. 4, pp. 705–715, Apr. 2014.

[16] L. Assirati, N. R. Silva, L. Berton, A. A. Lopes, and O. M. Bruno, "Performing edge detection by Difference of Gaussians using q-Gaussian kernels," *J. Phys. Conf. Ser.*, vol. 490, p. 012020, Mar. 2014.

[17] A. Wijayana, "Analysis and Implementation of Objects Tracking Using ASIFT and Mean Shift Methods," vol. 2, p. 11, Apr. 2015.

[18] D. Chen, J. Wang, and Y. Zhou, "Face Detection Method Research and Implementation Based on Adaboost," in *2010 International Symposium on Intelligence Information Processing and Trusted Computing*, 2010, pp. 643–646.