Face Recognition Using Local Binary Patterns Histogram Method Using Raspberry PI

Budi Cahyo Wibowo
Program Studi Teknik Elektro,
Fakultas Teknik,
Universitas Muria Kudus
Kudus, Indonesia
budi.cahyo@umk.ac.id

Imam Abdul Rozaq
Program Studi Teknik Elektro,
Fakultas Teknik,
Universitas Muria Kudus
Kudus, Indonesia
imam.rozaq@umk.ac.id

Andre Maulana Yusva
Program Studi Teknik Elektro,
Fakultas Teknik,
Universitas Muria Kudus
Kudus, Indonesia
andremyusva@gmail.com

Abstract : **Throughout his life, humans have the ability to recognize tens to hundreds of faces. One of the biometric techniques that relate body measurements and calculations that are directly related to human characteristics is a system that can detect and identify faces. To be able to overcome various current problems, facial recognition is required through computer applications, including identification of criminals, development of security systems, image and film processing, and human-computer interaction. So the author makes a face processing system based on Raspberry Pi with the Local Binary Patterns Histogram (LBPH) method. In running a facial recognition system using a face, at the initial stage the process of sampling the face of the person who is the owner of the room access is carried out. Then from the face samples that have been obtained, the learning process is carried out by converting the image into digital values through the Local Binary Patterns Histogram method. This method reduces image data into simpler data, to speed up the face recognition process. The results of the test show that face recognition works as expected, even being able to detect at low light brightness values (≥6 lux) and even face recognition accuracy of 79.15%. For face data that has been through the learning process, the face can be recognized, then the recognized face data is stored in a directory.**

Keywords: ***Raspberry, Face Recognition*, LBPH, Local Binary**

I. INTRODUCTION

One of the main areas in the biometric verification process, such as matching signature patterns, fingerprint patterns, and the retina of the eye, is Human Face Recognition. Its application has been used in various cases, such as network security systems, door control systems, to the use of attendance recording systems[1]. The development of image processing technology (Image Processing) at this time can produce a person's facial recognition (Face Recognition) directly, where the room has a camera that can directly access data through software[1].

Face recognition is a facial recognition technique that is the same as fingerprints and eye retinas, where camera captures are matched with photos and facial curves that are already in the database[2]. Many facial recognition methods have been researched and introduced through articles that have been published in journals. Among them is research with the title Identification of Face Recognition for Presence Systems Using the KNN (K-Nearest Neighbor) Method, this research is implemented for identification of presence systems with a focus on research results on presence validation tests conducted[3].

Indra Darma Wijaya also conducted research related to face recognition, where his research was applied to server room security systems using the triangle face method. This research was conducted to meet the needs of a server room security system that is easy to apply and inexpensive in terms of cost and maintenance. The results of this research detection system has an accuracy of 75%[4]. Face Recognition using the Haar-Cascade method which is a face detection system will be applied to the Raspberry Pi which is a Single-Board Circuit. Smarthome occupant identity will be stored in a dataset. The room can be opened if the facial data is in accordance with the facial sample data that has been entered by system learning in recognizing faces. The accuracy rate in this study using the Haar Cascade has a percentage of 65%. Researchers connected face recognition to the process of opening and closing room doors. All stages carried out are in the form of Self Process on RasPi[5].

To process facial recognition data, Raspberry Pi is used, or often abbreviated as RasPi, which is a single-board circuit (SBC) computer that is almost the size of a credit card[6]. The need for the use of Internet of Things technology. One of the advantages of IoT is that it can carry out home monitoring from a long distance even by utilizing existing applications. Telegram Messenger is an instant messenger application. This application is open source, so users can see the source code, protocol and Application Program Interface (API) contained in it[7].

Based on the above background, it is necessary to introduce other facial recognition methods for learning biometric techniques, which are body measurement and calculation relationships that are directly related to

human characteristics. So, the authors created a Raspberry Pi-based facial processing system with a method which is a development of Local Binary Patterns, namely Local Binary Patterns Histogram (LBPH).

II. BASIC THEORY

In this research, the LBPH (Local Binary Patterns Histogram) method was used in the face recognition process because it is fast and easy to apply. This method performs image matrix data reduction, converts it into binary data, then performs histogram formation, and to create a larger grid, it is necessary to combine the histograms into one to become a digital image according to the original.

A. Face Detection

Biometrics is a technology that utilizes the characteristics of the human body as a differentiator between one person and another, for example fingerprints as fingerprint technology, eye prints, voice as voice detection technology and faces as face detection technology[4].

Face detection is the initial process regarding the system's ability to recognize an input image whether there is a face or not. The face detection method used in this study by the authors is the Haar Cascade Classifier algorithm. In general, haar-like features are used to detect objects contained in digital images. The term Haar is taken from a mathematical function (Hhaar Wavelet) which is in the form of a box, similar in principle to the Fourier function. However, this method is not effective, because initially image processing is limited to the RGB (Red, Green, Blue) value of each pixel. Then the previous method was developed into a Haar-Like feature by scientists Viola and Jones. The Haar-like feature reduces the image Per box resulting in several pixels being processed and there is a difference in value between the dark and light areas, the difference in value is used as the basis for image processing.

Calculation of the value of the Haar-like feature method is to subtract the pixel value between the white area and the black area. A media, namely Integral Image, is used in the Haar Algorithm to speed up the value calculation process. Integral Image is an image in which the value of each pixel is taken from the sum of the pixel values from the bottom right to the top left. As an example, a pixel (a, b) has an accumulative value for all pixels (x, y). Where $x \leq a$ and $y \leq b$. In using the haar cascade method, there are several types of images that can be processed, one of which is grayscale.

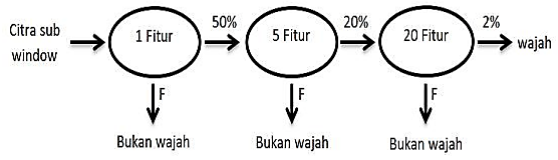


Figure 1. Flow Of *Cascade Classifier*[8]

To get more accurate results, the Cascade Classifier step is carried out in the form of calculating multiple and repeated Haar Feature values. Figure 1 shows the workflow of the Cascade Classifier until there is a reduction in the past sub-images until they are close to the existing sample images[9]. Facial recognition is needed by various parties, including civilians, the police and the military for identification verification and physical access control. Face detection is a technology that is often used and is always being developed in line with developments in computer technology[10].

B. Face Recognition

Face recognition is a computer technology for determining face location, face size, facial feature detection and background image ignoring, then facial image identification is performed[8].

Face recognition is a recognition method that focuses on faces. This recognition can be divided into two parts, namely when it is recognized or not recognized, after carrying out a comparison with the previous pattern stored in the database. This method is required to recognize objects other than faces. The facial recognition model calculation has several problems. Difficulties arise when faces are displayed in a pattern that contains unique information that distinguishes them from other faces[6][11].

The face is the part of the human body, which is the focus of attention in social interactions, the face plays a vital role by showing identity and emotion. The human ability to know someone by their face is extraordinary. We can recognize thousands of faces because of the very frequent or fleeting frequency of interactions even over a very long period of time[12].

In the work process of face recognition, an algorithm is needed that is able to detect objects to make it easier to retrieve facial image datasets and methods that can study received facial image data so that they can recognize the desired facial object[13].

C. Local Binary Patterns (LBP)

LBP is a texture analysis method that utilizes structural models and statistics. The LBP operator uses a comparison of the gray values of neighboring pixels. The basic LBP operator is 3 x 3 using 8 neighboring pixels in from a center pixel ic[14].

The nth neighbor pixel is thresholder using the gray value of the middle pixel and the thresholding function s(x). The binary code resulting from the neighboring pixel LBP operator will be used to represent the features of the middle pixel ic. The LBP procedure can be extended using different radii and neighbors, called Circular LBP.

The first process is subtracting the side pixels from the center pixels. Then the result of the reduction is thresholder, if the result is ≥ 0 then it is given a value of 1 and if the result is <0 then it is given a value of 0. After that, the side pixel binary values are arranged anticlockwise, and the 8 binary bits are converted back to a decimal value to replace the value mid pixel bit. The advantages of LBP are that it is easy to use and

the learning is lower, so it does not require a long time for feature extraction.

D. Local Binary Patterns Histogram (LBPH)

Image Matching process with Local Binary Patterns Histogram algorithm. With this method, the photos that have been studied will be compared with several images in the database and then adjusted using the histogram values extracted from the images using the Local Binary Patterns Histogram equation. The main characteristic of face recognition using this method is the composition of the micro-texture-pattern, which is a non-parametric operator that describes the local spatial layout of the image. LBPH is defined as the ratio of the binary pixel value at the center of the image to the 8-pixel values around it. Just like the LBP process as an initial stage, then the results of the process are formed into a histogram to represent the original image value[15].

E. Raspberry PI

The Raspberry Pi is an SBC (single-board computer) the size of a credit card. Raspberry pi is equipped with all the functions of a complete computer, using ARM's SOC (System-on-a-Chip) packaged and integrated on a PCB (circuit board). This Raspberry Pi is able to work like a computer in general with the ability to run the Linux operating system and applications such as LibreOffice, multimedia (audio and video), web browsers, or programming[16].

Figure 2. Raspberry PI

F. OpenCV (Computer Vision)

OpenCV is a free library developed by Intel Corporation developers. This library consists of computer vision functions and API (Application Programming Interface) for high-level and low-level image processing and as real time application optimization. OpenCV is highly recommended for programmers who will focus on computer vision, because the library is capable of making reliable applications in the field of digital vision and has features similar to human visual processing[16].

G. Webcam

Webcam (abbreviation of web camera) is a designation for a real time camera (meaning current conditions) whose images can be accessed or viewed via the World Wide Web, instant messaging programs, or video call applications.

Figure 3. Web Camera

A simple web camera consists of a standard lens mounted on a circuit board to capture image signals; casing (cover), including the front casing and side casing to cover the standard lens and has a lens hole in the front casing that is useful for inserting images; support cable, which is made of flexible material, one end is connected to the circuit board and the other end has a connector, this cable is controlled to adjust the height, direction and angle of view of the web camera[4].

III. METHODS AND DESIGN

This research methodology can be divided into four stages of the process, namely the facial data collection stage, the face learning stage, the face recognition stage, and the database storage stage.

A. Process of Data Retrieval and Learning

In Figure 4, you can see the process of taking facial data as input which functions as a sample photo which will later be used in the facial recognition system. To do this, it is necessary to use the face of the person you want to recognize to sample data on the face of the person you want to recognize. In addition, an ID code (can be a number or a person's name) is required for each person who wants to be recognized. So, the algorithm can use the information to identify between the images entered and the output images. Output images of people must match the same ID. After the data collection steps are in accordance with the conditions given, the next step is to carry out face learning by converting facial image samples into matrix-shaped digital values for each sample. The digital values are stored together in the TrainningData.yml file. With the training set that has been made, then the LBPH computation steps are carried out.

B. Implementing LBP Operations

LBPH's first computational step is to create an intermediate image that describes the original image with facial characteristics. To do so, the algorithm uses the sliding window concept, based on radius and neighbor parameters. For example, a part of an image measuring 3x3 pixels forms a 3x3 matrix which contains the intensity of each pixel (0 ~ 255). Then, the middle value of the matrix is taken to be used as a threshold. This value will be used to determine the new value of the 8 matrix values. For each neighbor of the threshold center value, a new binary value is assigned.

by assigning 1 to values equal to or higher than the threshold and 0 to values lower than the threshold. The matrix will contain only binary values (ignoring central values).

Next, it is necessary to combine each binary value from each position, even the matrix from row to row is changed to a new binary value (such as 10001101). Then, it converts this binary value to a decimal value and sets it to the matrix center value, which is actually the pixels of the original image[17].

At the end of this procedure (LBP procedure), a new image is obtained that better represents the characteristics of the original image. Note: The LBP procedure is extended using a different radius and neighbor, called Circular LBP.

C. Extracting Histograms

Figure 3 is the last step and the result of the LBP process, using the Grid X and Grid Y parameters to divide the image into several grids, as can be seen as follows.

Figure 5. LBP results[17]

Based on Figure 5, a histogram of each region is then made. Then concatenate each histogram to create a new and bigger histogram. For example, with an 8x8 grid, you will get 8x8x256 = 16.384 in the final histogram. The final histogram represents the characteristics of the original image.

D. Do Face Recognition

At this stage, the algorithm has been trained. Each image used is represented by a histogram created from the training dataset. So, given an input image, pre-tagspan the new image and create a histogram that represents the image. So, to find a match with the input image, you only need to compare the two histograms and return the image with the closest histogram. Various approaches can be used to compare histograms (calculating the distance between two histograms. One of them is using the Euclidean distance (which is well known) based on the following formula,

$$D = \sqrt{\sum_{i=1}^{n}(h1_i - h2_i)^2} \quad (1)$$

So, the output of the algorithm is the ID of the image with the closest histogram. The algorithm must also return a calculated distance, which can be used as a measure of confidence.

After the username appears, the door access results are stored in the database. In this study, data is stored in csv (comma separated value) format which is compatible with all spreadsheet applications including LibreOffice and Microsoft Excel. Besides that, the data is also stored in the SQLite portable database. This research also sends notifications that are directed to telegrams by sending pictures of faces that access the room in real-time.

IV. ANALYSIS AND DISCUSSION

The tests carried out in this study consisted of several stages including the results of the facial recognition program, webcam component testing, and facial recognition system testing.

A. Results of System Design

The result of designing the entire system, from software and hardware, creates a face recognition system using the LBPH method, which is applied for various purposes, including room security systems, presence systems, etc. The results of this system can be seen in Figure 4.

Figure 6. Hardware Design Results

From the picture the camera is connected via USB raspberry pi, for the application after the raspberry is turned on the application is ready to use. For this research programming is done using OpenCV through the Python programming language.

B. Program Design Results

Face recognition from the design of the research method can describe the results of several stages of this study,

1. Stages of Data Recording

At this stage the Python language programming code to capture facial images via a webcam camera connected to the Raspberry Pi. After that the image sequence is stored in the dataset directory. Next, the facial images that have been saved, like checking first, if there are images that are not good and clear, some of these images are deleted.

2. Face Learning

After the image is adjusted to the data recording, the face training (learning) stage is then carried out, the learning process uses the LBPH method and then the learning outcome data is stored in "trainingData.yml" which is in YAML format (yet anathor markup language) in the recognizer folder. This data is useful later for the facial recognition process. Programming takes libraries including the OS (Operating system) which is used to be able to open directories, cv2 (OpenCV), numpy, and PIL (Python Image Library). For the introduction of using the LBPHFaceRecognizer library, it is useful to run the LBPH library.

From the program above defines taking pictures, taking pictures above by accessing the dataset folder is carried out by the training process and the results are stored in the recognizer directory with the file name "trainningData.yml".

3. *Face Recognition*

After the face learning process has been carried out, to try the results of facial recognition it can be done with the face recognition stages, after the face is recognized the data (id, name, date, time) is stored in the face.db which is SQLite database.

In the recognition process using the while function so that it repeats until it stops until an image is found that matches the data that has been recognized, in the programming code above after the face is predictable the program will retrieve id and confidence data. Furthermore, the data is stored in the image and for naming the file with the date and time, this aims to find out the face that accessed the room at that time.

4. *Interface Design*

Screenshots of the process of recording facial images through the Raspberry Pi camera which is demonstrated via GuiZero can be seen in Figure 7. The use of the Graphical User Interface (GUI) functions to simplify running programs based on the interface of some of the programs above.

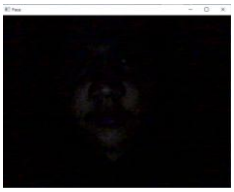
Figure 7. Demo Using GuiZero

In Figure 7, there are two command buttons, namely Entry and learning, and detection. Where when the Entry and learning buttons are pressed it will call the face data sampling program after which it calls the face learning program, with a process like the description in the sub-chapters above by giving a new id and name then the photo data samples that have been taken are carried out by the learning process by changing the photos into digital value. Meanwhile, if the detection button is pressed, it will invoke the face detection program. The process described will detect a face when it does not pass the recognized face threshold, which will be stored in the login directory.

5. *Webcam Testing*

The results of this test are carried out to determine whether the webcam camera as a sensor can function properly or not. The webcam used is already in the form of a module, so it's easy to assemble or use. Simply connect the USB that is on the Raspberry Pi. Schematic of a series of cameras with specifications of 5 MP with a resolution of 640x480 connected directly via the Raspberry Pi USB Port. In this test to try the Webcam as a face detection sensor in a face capture system that can detect faces in room conditions with lighting levels expressed in lux, this test uses a YF-1065 digital lux tester to determine the light intensity in the room. The following tests have been carried out summarized in table 1.

Table 1. Webcam testing with facial recognition using the LBPH method.

No	Light Intensity (Lux)	Face Detection	Information
1	0		Invisible face
2	4		Invisible Face
3	6		Visible Face
4	8		Visible Face
5	10		Visible face
6	20		Visible Face

From the results of the experiments that have been carried out, it can be concluded that the webcam can work even when the room is dim, even with lighting values ≥ 6 lux, but it is more optimal in bright room conditions with lux values above 10 lux.

6. *Face Recognition Testing*

Testing at this stage measures the accuracy of the facial recognition system being implemented to find out the percentage of the system in face recognition

accuracy according to the ID and name given to the user of the room and also to find out errors when the system recognizes a face incorrectly. Knowing the level of facial recognition accuracy is very important, because whether the performance of the system fails will depend on the results of the tests that have been carried out. The testing stage in using the facial recognition system is to prepare 4 people as users who will later access the room. The user is then carried out the stages of the face recognition process in the form of facial data sampling and face learning. Testing the level of accuracy is carried out by carrying out the facial recognition stages for each person, so the system is declared to be running correctly when the face is recognized. However, on the contrary or being recognized as someone else, an error occurs in the system. From the test results with each of the 75 tests on everyone, data is obtained as in Table 2 below:

Table 2. Face Recognition

Users	Recognized Face (%)
Andre MY	80
Mike	74,6
Kepin	77
Cuwil	85

From the results of the table test, it shows that the number of correct recognition and incorrect recognition has a relationship in the calculation. The total test data for everyone is 75 tests whose results will be attached in the attached sheet. The attached test results are given status 0 if they are not recognized or exceed the given threshold value and are given status 1 if the face is recognized. It can be concluded from the system experiment, the level of accuracy of face recognition using the LBPH method functions well. The LBPH method is very easy to recognize when the light conditions are sufficient. System errors that occur are registered individuals detected as IDs and names that do not match the detected faces. It can be risky if used as a security system. The average level of accuracy regarding registered faces based on the percentage value of everyone's face recognition in the table above is 79.15%, this indicates that the facial recognition method is working well enough for faces that the system has recognized.

V. CONCLUSION

In this study the introduction of the LBPH method can identify users with an average match of 79.15%. The webcam device as an input sensor in face detection and processing can work in dim conditions even with lighting values of more than 4 lux. Photo data collection and learning are needed to introduce faces to the security system using the LBPH method. RasPi as a single-board computer can perform a facial processing system which is manifested in the form of room security.

REFERENCES

[1] C. Lesmana, R. Lim, and L. W. Santoso, "Implementasi Face Recognition menggunakan Raspberry pi untuk akses Ruangan Pribadi," *J. Infra Petra*, vol. 7, no. 1, pp. 2–5, 2019, [Online]. Available: https://publication.petra.ac.id/index.php/teknik-informatika/article/view/8046.

[2] Munawir, L. Fitria, and M. Hermansyah, "Implementasi Face Recognition pada Absensi Kehadiran Mahasiswa Menggunakan Metode Haar Cascade Classifier," *InfoTekJar J. Nas. Inform. dan Teknol. Jar.*, vol. 5, no. 1, pp. 314–320, 2020, [Online]. Available: https://jurnal.uisu.ac.id/index.php/infotekjar/article/view/2333/pdf.

[3] D. Yulianti, I. Triastomoro, and S. Sa'idah, "Identifikasi Pengenalan Wajah Untuk Sistem Presensi Menggunakan Metode Knn (K-Nearest Neighbor)," *J. Tek. Inf. dan Komput.*, vol. 5, no. 1, pp. 1–10, 2022, doi: 10.37600/tekinkom.v5i1.477.

[4] I. D. Wijaya, U. Nurhasan, and M. A. Barata, "Implementasi Raspberry PI Untuk Rancang bangun Sistem Keamanan Pintu Ruang Server Dengan Pengenalan Wajah," *J. Inform. Polinema*, vol. 4, pp. 9–16, 2017.

[5] M. I. Maulana, M. Nishom, and D. I. Af'idah, "Pengolahan Citra untuk Identifikasi Pelat Nomor Kendaraan Mobil Menggunakan Metode Haar Cascade dan Optical Character Recognition," *J. Bumigora Inf. Technol.*, vol. 4, no. 1, pp. 1–16, 2022, doi: 10.30812/bite.v4i1.1952.

[6] Ikhwan Alfath Nurul Fathony, "Sistem Kamera Pengawas Dengan Menggunakan Raspberry Pi Disertai Motion Detection Dan Auto Backup Cloud (Google Drive)," Universitas Islam Indonesia, 2018.

[7] M. I. KURNIAWAN, U. SUNARYA, and R. TULLOH, "Internet of Things : Sistem Keamanan Rumah berbasis Raspberry Pi dan Telegram Messenger," *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 6, no. 1, p. 1, 2018, doi: 10.26760/elkomika.v6i1.1.

[8] D. Suprianto, R. N. Hasanah, and P. B. Santosa, "Sistem Pengenalan Wajah Secara Real-Time dengan Adaboost, Eigenface PCA & MySQL," *J. EECCIS*, vol. 7, no. 2, pp. 179–184, 2013.

[9] T. F. Abidin, A. A. AzZuhri, and F. Arnia, "Pengenalan Karakter Plat Nomor Kendaraan Bermotor Menggunakan Zoning dan Fitur Freeman Chain Code," *J. Rekayasa Elektr.*, vol. 14, no. 1, pp. 19–25, 2018, doi: 10.17529/jre.v14i1.8932.

[10] A. R. Syafira, "Sistem Deteksi Wajah Dengan Modifikasi Metode Viola Jones," *J. Tek. Elektro*, vol. 17, no. 01, pp. 1–8, 2020.

[11] N. Anggraini, F. Martunus, I. Marzuki Shofi, and L. K. Wardhani, "Implementasi Face Recognition Dengan Opencv Pada 'Smart Cctv' Untuk Keamanan Brankas Berbasis Iot," *J. Ilm. FIFO*, vol. 13, no. 1, p. 41, 2021, doi: 10.22441/fifo.2021.v13i1.005.

[12] Afrizal Zein, "Pendeteksian Multi Wajah Dan Recognition Secara Real Time Menggunakan Metoda Principal Component Analysis (Pca) Dan Eigenface," *J. Teknol. Inf. ESIT*, vol. XII, no. 01, pp. 1–6, 2018.

[13] B. Hartika, "Face Recognition Menggunakan Algoritma Haar Cascade Classifier Dan Convolutional Neural Network," *J. Math. UNP*, vol. 6, no. 3, pp. 12–19, 2021, [Online]. Available: http://ejournal.unp.ac.id/students/index.php/mat/articl

e/view/11954.

[14] A. L. Shiyam, "Presensi Mahasiswa Dengan Ekstraksi Fitur Wajah," pp. 1–11, 2019.

[15] S. Al-Aidid and D. S. Pamungkas, "Sistem Pengenalan Wajah dengan Algoritma Haar Cascade dan Local Binary," *J. Rekayasa Elektr.*, pp. 1–8, 2018.

[16] R. S. Prihantono, A. Mahzaruddin, Shiddiqi, Hudan, and Studiawan, "Rancang Bangun Sistem Keamanan dan Pengenalan Objek dalam Ruangan Sebagai Pengganti CCTV dengan Menggunakan Raspberry Pi," *J. Tek. POMITS*, vol. 2, pp. 2301–9271, 2013.

[17] K. S. d. Prado, "Face Recognition: Understanding LBPH Algorithm," 2022. .