Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)
Vol. 8, No. 4, December 2022, pp. 621-633
ISSN: 2338-3070, DOI: 10.26555/jiteki.v8i4.24643

Raspberry Based Hand Gesture Recognition Using Haar Cascade and Local Binary Pattern Histogram

Helfy Susilawati [1], Fitri Nuraeni [2]
[1] Department of Engineering, Universitas Garut, Jl. Terusan I No.13, Kec. Tarogong Kaler, Kab. Garut. Jawa Barat 44151, Indonesia
[2] Department of Engineering, Institut Teknologi Garut, Jl. Mayor Syamsu No.1,Kec. Tarogong Kidul, Kab.Garut, Jawa Barat 44151, Indonesia

ARTICLE INFO

Article history:

Received August 19, 2022
Revised November 28, 2022
Published December 28, 2022

Keywords:

Face recognition;
Hand gestures;
LBPH

ABSTRACT

One of the methods that can be used to detect the attendance is hand gesture detection. This research aims to detect hand gestures made by the employees to ensure whether they really come to work or not. This research make the chance for manipulation using photo or fake GPS is quite small. For the purpose of hand gesture recognition, this study utilized Local Binary Pattern Histogram algorithm. The hand gesture image was first taken using a raspberry pi camera and then processed by the device to examine whether it matches the registered ID or not. The results showed that ID recognition by using hand gestures is detectable. The number recognition in hand gestures includes numbers 1 to 10. The test results showed that for 5 trials, the average time required for reading hand gestures using a laptop was 9.2 seconds, while that of using raspberry was 14.2 seconds. The results of this research show that the system has not been able to distinguish which hand is read first, so numbers that have the same number are considered the same, such as 81 and 18. So, the motion reading using a raspberry takes longer than that of using a laptop because the laptop's performance is higher than that of a raspberry and system cannot distinguish between numbers consisting of the same number.

Corresponding Author:

Helfy Susilawati, Department of Engineering, Jl. Terusan I No.13, Kec. Tarogong Kaler, Kab. Garut. Jawa Barat 44151, Indonesia
Email: helfy.susilawati@uniga.ac.id

1. INTRODUCTION

Recording attendance is an activity carried out daily by people who work either in private companies or state companies (i.e. civil servants). Currently, collecting the attendance of state civil servants is done by using employee selfie photos and GPS detection in their cellphone. However, attendance detection using these two methods still can be easily manipulated by leaving a special cellphone for attendance at the work location and using a previously printed photo for the selfie one. Therefore, to ensure that the employee comes to work and stands by on location, motion or gesture recognition feature is highly required. This gesture recognition will help distinguish whether the employee is actually onsite or just using a photo. Gestures that can be used are hand gestures. Hand gesture is the most popular gesture which detects hand movement [1]. Due to the advancement of technology, hand gesture recognition do not require special devices [2]. Hand gestures can help overcome difficulties and make the human's tasks become easier [3]. Hand gestures are widely used for non-verbal communication such as medical applications, robot control, and human-computer interaction (HCI) [4]. In human-computer interaction (HCI), hand gestures are a field that can be explored more [5]. The hand gestures used are representative of the ID of each employee. Each finger represents its own ID; thus, 10 fingers in humans indicate 10 IDs.

Several studies related to hand gestures have been carried out, including research that investigate hand gestures for sign language using 3DCNN for its recognition [6]. The hand gestures for human-robot interaction (HRI) was also investigated [7]. In this research, there were 15 variations of hand gestures, of which 9 variations were used to indicate numbers 1 to 9 and the other variants were punch, span, horizontal, collab, xSign, and TimeOut. Hand gestures were also used in smart dolls as assistants when telling stories to children [8]. This study utilized 3 types of gesture sets, namely the basic single-hand gesture which has 7 gesture patterns, the second type, the advanced sing-hand gesture set, which has 7 gesture patterns, and the navigation gesture set which has 8 gesture patterns. The use of hand gestures was also used for elderly care [9].This study involved a kinect sensor and used 5 types of gestures where the average accuracy of recognizing the five gestures was 95.53%, and the average error in recognizing the type of gesture was 4.47%. Hand gestures was also used for video annotation [10]. The method used was the KNN method, using 20 types of gestures. The results showed that by using the KNN method the average recognition rate was 97%. Hand gesture recognition was also employed to control household appliances remotely [11]. This study used an accelerometer and gyroscope found on mobile phones to control home appliances remotely.

Based on the review on the previous works explained above, hand gestures for recording attendance still require further explorations. Hand gesture used to complement the face detection system for attendance has been done in research [12] and research [13]. In previous research, no one has combined face detection with hand gesture using the Haar Cascade and Local Binary Pattern Histogram. The algorithms used in this research are Haar Cascade and Local Binary Pattern Histogram. Hand gestures are expected to ensure that employees are really present at the workplace. The purpose of this study is to build a system that can detect faces followed by hand gesture detection to determine the attedance of employees who are actually present at work and to minimize fraud by employees

2. METHODS

The methods used in this research are face detection and hand gesture recognition. The input, process, and output of face recognition can be seen in Fig. 1.

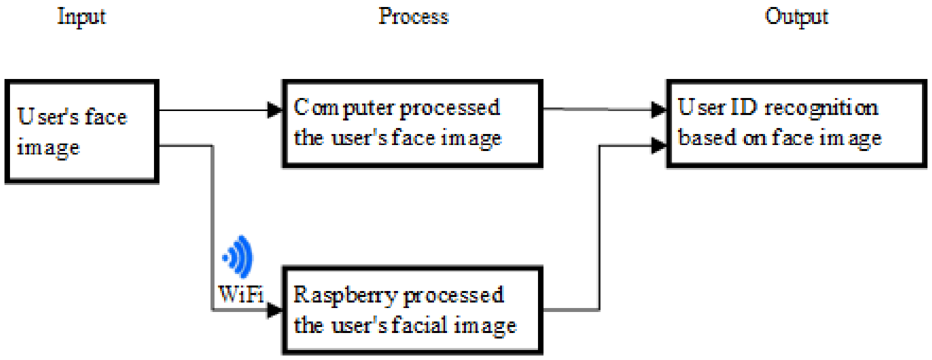


Fig. 1. Diagram of the Input, Process, and Output of Face Recognition

From Fig. 1 shows the input, process and output diagram of face recognition detection. The input in this research was the face image to be tested, while the process involved a computer and a raspberry. On its use, the computer requires no additional internet connection to process the face detection, while a raspberry needs a wifi or internet connection. The output was facial recognition data according to the previous training data.

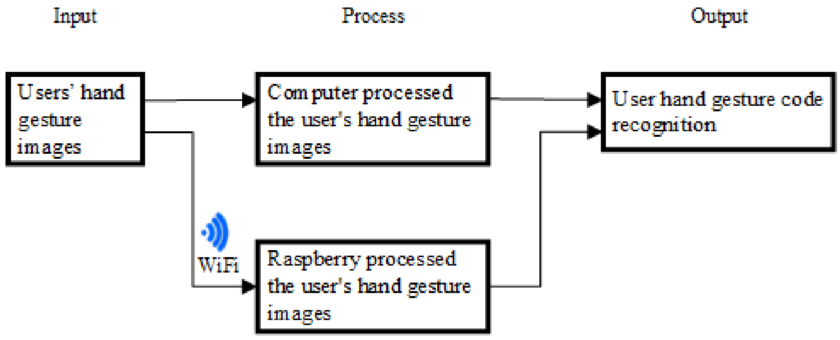


Fig. 2. Diagram of Input, Process, and Output Hand Gesture Detection

Form Fig. 2 shows a diagram of the input, process and output of hand motion detection. The input in this study was the images of the user's hand gestures, and the process utilized a computer and a raspberry. The output was hand gesture data that has been previously set. In this study, to detect face and hand movements, the Haar Cascade algorithm and Local Binary Pattern Histogram (LBPH) were used.

LBPH has been used in several studies to detect faces in real time as was done in studies of [14][15]. Haar Cascade for face detection has also been utilized in several studies including research conducted by [16][17][18]. LBPH is an algorithm that is often used to detect faces (face recognition). A study comparing several algorithms in detecting faces showed that the use of the LBPH algorithm has advantages over the other 2 algorithms compared, in which LBPH produced an accuracy of 40% and 95% of the existing data set [19]. LBPH can also serve function as a classifier of high-speed patterns, and the results indicated that the system built can recognize the 1.12µs class, and is 33% faster than the other pattern classification system [20]. LBPH was also employed to recognize facial expressions such as angry, happy, surprised, disgusted, sad, neutral and afraid in the Multimedia Understanding Group (MUG) and the Japanese Female Facial Expression Database (JAFFE) [21].

Moreover, Haar Cascade in digital image processing has also been widely used, including in the research that seeks to detect traffic violations that occur at the zebra cross traffic light [22]. Haar cascade was used to in image indexing research whose results showed that the algorithm was able to retrieve and display images with high retrieval accuracy and moderate relevance [23]. Haar Cascade was also used to detect faces that used masks. To be specific, Haar Cacade was used to detect faces, while YOLO 3 was utilized to detect masks. The results indicated that the system obtained 90.1% accuracy [24]. Haar cascade was also applied to control the smart traffic light prototype. In this study, the image was divided into 2, namely positive and negative images. The positive images contained images with the object of a 4-wheeled vehicle, while the negative images consisted of images of the vehicle sign. The large number of vehicles will affect the duration of the traffic signs [25].

2.1. Research Procedures

This research utilized two hardware devices, namely Computer and Raspberry. Raspberry Pi was used to implement artificial vision [26]. Data that were previously processed and trained in the computer were then uploaded to Raspberry. Raspberry was involved in order to save space since it can function as a computer replacement. The use of Raspberry can help reduce costs, simplify the process, and is easy and fun to learn and use [27]. Moreover, this present study applied two algorithms to perform face recognition and hand motion detection. In regard to the images processing, the images were divided into two, namely training data and test data. The training data were obtained by taking pictures several times so that the computer recognized the person. Meanwhile, the test data were gathered at the time of testing in which a person's image was given directly to check whether the computer recognized that person was the same person or not. In analyzing the data, there were 3 levels of stages; low level processing, intermediate level processing, and high level processing. The stages of this research were explained as follow:

1) Low level processing
   a. Image acquisition: the process of acquiring the objects in the form of faces and finger movements in real time
   b. Pre-processing :
      1. Grayscale: changes the color of the original image to gray. This process was carried out when taking the image for the registration process. The images that have been trained were then converted into images with gray or grayscale color. Grayscale images allow having their pixel gray levels decomposed into bit fields which are then compressed [28].
      2. Thresholding: a process of changing the value of a color or gray image into a binary image. Thresholding is one of the simplest segmentation methods, in which it has fewer iterations and the calculation is direct, compared to other techniques [29]. The purpose of the Thresholding process is to separate the image into two parts based on a predetermined threshold value. The thresholding process will change the pixel value of the gray image to a value of 0 (black) or a value of 255 (white) following the specified threshold value. The most widely used method that is able to extract changes and information that do not change is probably thresholding [30].
      3. Active Contour: edge detection process on hand motion objects. This process will give a green line on each edge of the hand; thus, it will distinguish objects that are detected and those that are not. At this time active contour is a segmentation method that has been widely used in object extraction from remote sensing images [31].

4. Convex Hull: the outermost points on the contour. Convex Hull is a process that is required to help computers analyze those related to human skin color. In other words, it observes interaction in human hands [32]. Convex hull lines if connected will surround the contour, making all contour points be in the convex hull area. The use of the convex hull facilitates the implementation of algorithms with the aim of visualizing the tangent area, and reducing dimensions [33]. The convex hull is the smallest convez that contains the sample [34].
5. Convexity Defects: the areas between the convex-hull line and the contour line. Convexity defects give the value of each defect in vector form. This vector contains the starting and ending points of the defects line in the convex hull. These points represent the indices of the contour point coordinates. Convexity defects enables the program to recognize the movement and shape of the hands easily. Convexity is a feature of an image which at a time this image may only contain the contours of the hand. In other words, convexity contains the minimal features needed to describe the hand [35].

2) Intermediate level processing
   a. Segmentation: This stage aims to divide the image into several main parts that contain important information about the object. The segmentation method was carried out by first preprocessing the image and then selecting the pixel with the lowest gray value as a seed that will grow certain elements [36]. At this stage, the haar cascade classifier method was used to detect objects when the camera was taking real-time images. The input images in the form of faces, mouths, eyes were extracted using Harr Cascade, and then Sobel was used to obtain characteristic values [37].
   b. Image Representation and Description: In this case, feature extraction and selection will be carried out on an area that represents an equation that can distinguish classes of image objects. At this stage, the LBPH method was used through which the image matching process between the image that has been learned and the image taken by the camera in real time were carried out. LBPH is a very simple and efficient texture operator [38].
3) High level processing
   Image recognition and interpretation: an image classification process in which the captured image matches with the trained image. Image classification is used to classify images into different categories [39]. Image classification has the aim of identifying the features contained in the image and then labeling it based on the identified features [40].

2.2. Flowchart

The flowchart of the system is show in Fig. 3 which is an image of the entire system. The system begins by creating training data for faces detection and then training data for hand gesture. After the training data is created, a test is carried out using a computer. After the test using a computer, the training data and program are uploaded to the Raspberry pi. The final stage is to evaluate the system.

Fig. 4 is a flowchart of a facial and hand gesture recognition system using a computer and raspberry. In Fig. 4 is a flowchart of face and hand gesture recognition. The first input to the system is the face image of the user. If the face image is not recognized, then the face can be registered first. However, if the face image is recognized, a face detection process will be carried out to identify whose face is in the image. Once the latter is done, the next step is to input hand gestures following the hand movement code that has been previously set. If the hand gesture code can be recognized, the user's ID and the hand movement code will appear. However, if the code cannot be detected and read, then the data will be blank.

2.3. Testing

The testing in this study was divided into 2 stages. The first stage was facial recognition testing which has been undergoing the training data. The training data used 20 photos of each person.

Can seen in Table 1 showed 5 training data. For each training data, 5 samples were selected from the minimum of 20 samples that has previously been taken for each training data. Taking some samples for each training data was conducted so that the computer can recognize a person based on several sides of the face. By this way, if at the time of testing there is a different side, the computer can still recognize that person.

Meanwhile, the hand gesture recognition was carried out by using the pre-existing libraries. The fingers used were the fingers of both right and left hands. The fingers on the right hand show the numbers of 0 to 5, while those of on the left hand indicate the numbers of 6-10. The following are the codes of hand gesture recognition as shown by Table 2.

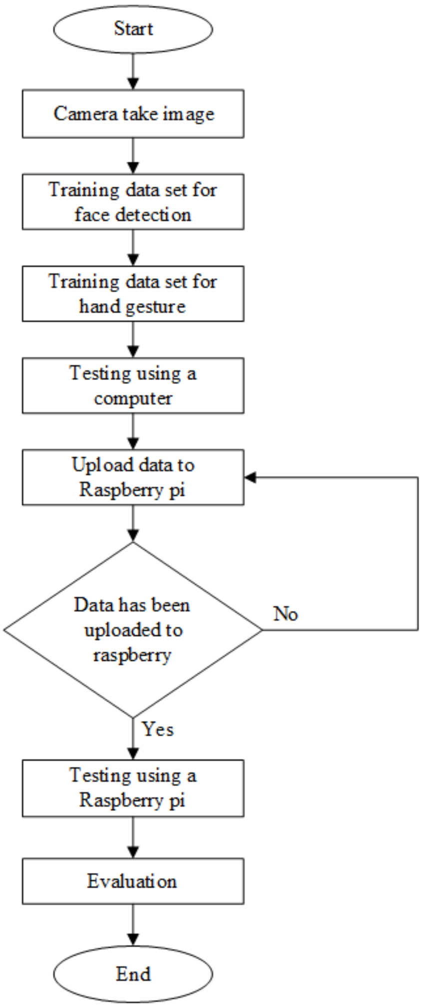


Fig. 3. Flowchart system

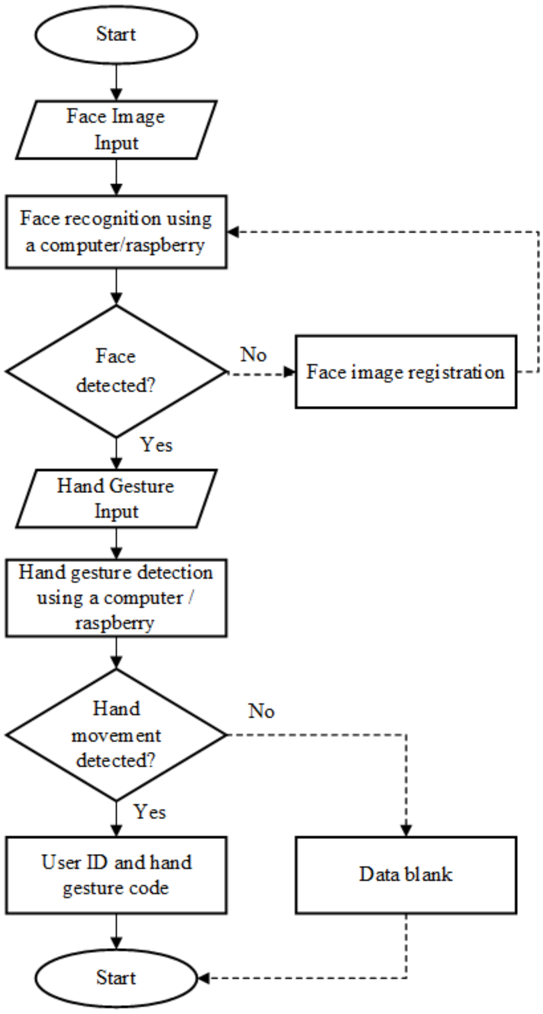
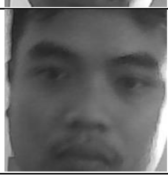
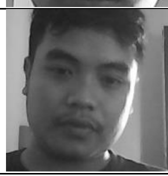


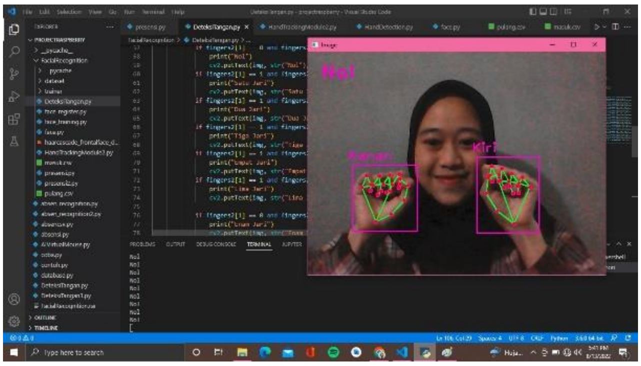
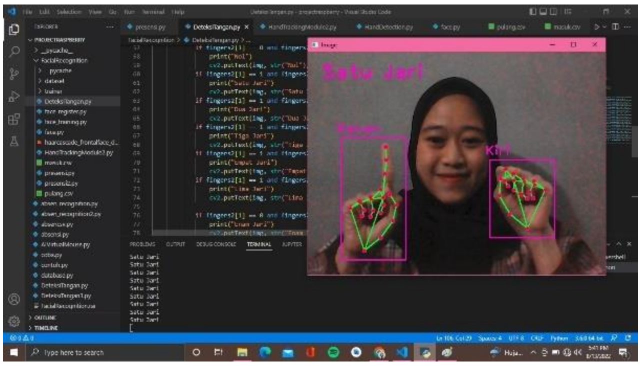
Fig. 4. Flowchart of Face and Hand Motion Detection

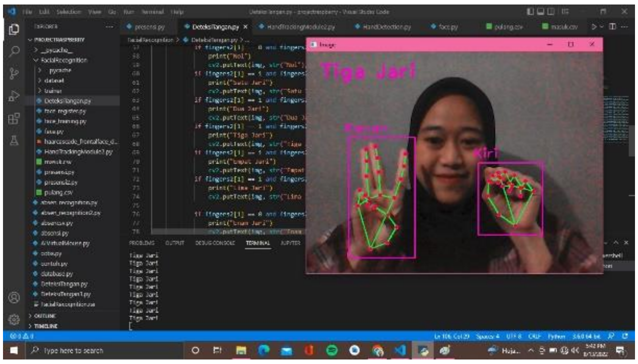
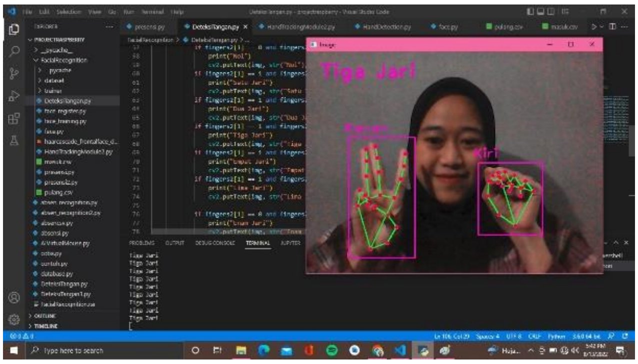
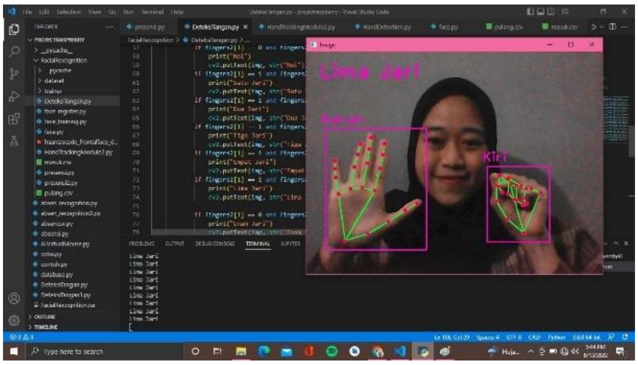
Table 1. Training data sample on 5 training data

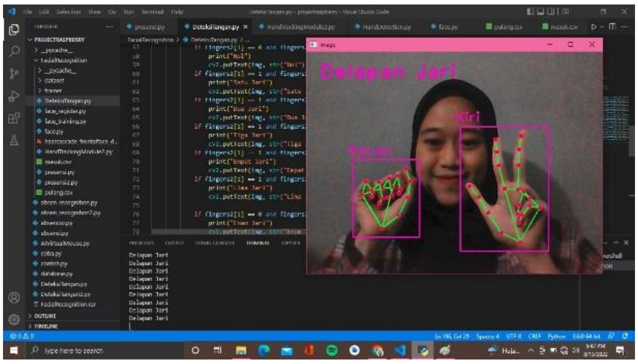
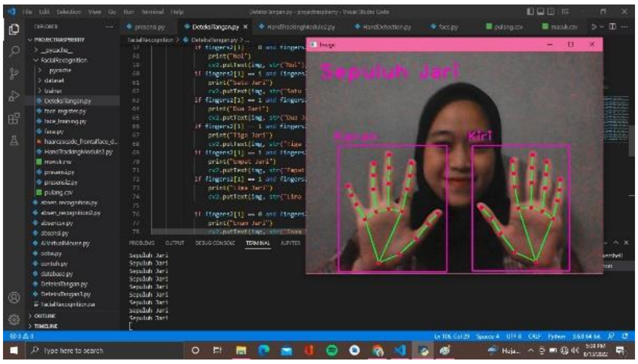
No.	User	Samples of Training Data				
1	1.					
2	2.					
3	3.					

No.	User	Samples of Training Data				
4	4.					
5	5.					

Table 2. Hand gesture codes

No.	Hand Gesture Codes
0	
1	
2	

No.	Hand Gesture Codes
3	
4	
5	
6	

No.	Hand Gesture Codes
7	
8	
9	
10	

3. RESULTS AND DISCUSSION

3.1. Results

The previous study of Abha Thakral's; Akshat Vohra's, studies which showed that lighting when taking training data and test data became one of the determining factors in obtaining accurate analysis in decision making process [41]. In this study, the testing was carried out by using 5 test parameters, namely facial expressions, eye contact, lighting, face angles, and user photos to determine the effect of these parameters on the readibility of face detection. The results showed that lighting is one of the factors that influences whether the system can recognize faces and hand gesture symbols.

It can be seen in the Table 3 showed that the four parameters have effects on the readability of the user ID that has been previously trained. If the facial expression, eye contact, lighting, and face angle are already present in the training data or at least close to the training data, then the face in the test data can be recognized. However, if these four parameters are not present in the training data, then the face will not be recognized. To test the recognition speed, a test using a stopwatch was carried out. It was done to determine the time required for registration, and faces and hand movement recognition.

Table 3. Test parameters

User ID	Facial expression	Eye Contact	Lighting	Face angles	User picture
1.	Have an effect	Have an effect	Have an effect	Have an effect	Unrecognized
2.	Have an effect	Have an effect	Have an effect	Have an effect	Unrecognized
3.	Have an effect	Have an effect	Have an effect	Have an effect	Unrecognized
4.	Have an effect	Have an effect	Have an effect	Have an effect	Unrecognized
5.	Have an effect	Have an effect	Have an effect	Have an effect	Recognized on the distance <= 10 cm

From Table 4, it can be seen that the fastest time of the 3 phases is the time for face recognition, while the longest time is in the hand gesture recognition phase. It is shown in Table 5 that by using raspberry, the fastest time of the 3 phases is the time for face recognition, while the longest time is in the hand gesture recognition phase. If the average time required for registration using a raspberry and computer was compared, then raspberry outscored the computer in which it has the fastest time. However, for the hand gesture recognition, the raspberry took a longer time than the computer.

Table 4. Time required to register and detect the faces and hand movements using Laptop

User ID	Register	Face Recognition	Hand gesture detection
1.	5 second	0.37 second	15 second
2.	6 second	0.31 second	10 second
3.	4 second	0.44 second	6 second
4.	4 second	0.15 second	5 second
5.	7 second	0.54 second	10 second
Average	5.2 second	0.362 second	9.2 second

Table 5. Time required to register and detect face and hand movements using Raspberry Pi

User ID	Register	Face Recognition	Hand Gesture Detection
1.	4 second	0.17 second	22 second
2.	4 second	0.15 second	10 second
3.	4 second	0.44 second	6 second
4.	3 second	0.18 second	10 second
5.	4 second	0.12 second	20 second
Average	3.8 second	0.212 second	13.6 second

Hand gestures with the same hand that cannot be read. For example, it happened in number 11 which consists of 2 numbers that are both on the right hand finger. Likewise, if both numbers are on the left, for example the number 69, then the system cannot read the number. The system also cannot recognize the difference in numbers, for example 18 and 81. The system will read hand movements after the right and left fingers are in the frame; thus the system will only read the same pattern, not for the opposite number.

It can be seen in the Fig. 5 and Fig. 6 showed that the readability results of the system are the same, namely 18, although the hand movement in Fig. 5 was done by raising the hand with code 1 on the right finger, and raising the hand movement with code 8 on the left finger. Despite different method in Fig. 6 in which the process was done by raising the hand with code 8 on the left finger and then raising code 1 using the right hand, the same legibility, which is 18, was achieved. Previous research can only represent 1 digit numbers [8], whereas in this research can represent 2 digit numbers.

Fig. 5. Hand gestures for number 18

Fig. 6. Hand gestures for number 81

3.2. Discussion

This research has several shortcomings that can be further developed, including how to get the hand gesture code where the system can read the code sequentially, not simultaneously. In this study, the hand gesture code can be read if the code is on two different sides and not sequential.

4. CONCLUSION

The results of the study indicate that face can be recognized properly if the previously measured parameters are already included in the training data. If these parameters are not contained in the training data, the recognition during the testing are difficult to be done. Additional parameters tested in this study were facial expression, eye contact, lighting, and face angle. The addition of training data can help overcome this issue. It is not possible to determine whether the system can detect a person from just a photo or not. Of the 5 users who experimented with the photos, 4 user photos could not help identify whether the photo was a user or not.

The results of the hand gesture test show that the hand gesture code can be recognized if both sides of the hand gesture are within the specified frame. In this study, the code for hand gestures with consecutive hand movements from the same finger side, either the left or the right side, cannot be read. Likewise, the same thing cannot happen for consecutive hand movements that start with the right hand first and then the left hand, or vice versa. The hand gestures are still considered the same as two numbers because the reading of the code for the hand gestures is the same.

In addition, for registration and face recognition, raspberry actually recognizes faster than the computer. However, for reading the hand gesture codes, the computer is actually faster than raspberry. The data uploaded to the raspberry are the ones that were previously processed on the computer; therefore, when they are on the raspberry, they are read based on how they have previously been tested on the computer making the raspberry take a quicker time to read. However, for hand gesture where higher image processing is required, the computer is faster than raspberries. This is natural because the raspberry, which is a small computer, has to perform two tasks, namely reading based on the existing data and recognizing which hand movement images are used. Raspberry will certainly experience delays than the actual computer.

Further research can be developed by adding the number of digits that can be read in one frame (such as 3 digits), then the system can also read digit numbers which are done sequentially. Further research can also be carried out by looking for better image processing methods for hand motion detection. In addition, a technique is needed so that the Raspberry Pi can detect faces and hand gesture faster

Acknowledgments

A sincere gratitude goes to the Faculty of Engineering of Universitas Garut which has facilitated this present research and to the Research Institute of Universitas Garut which has provided direction in conducting the research. Immense gratitude also goes to LPDP which has provided funding for the research.

REFERENCES

[1] R. E. Nogales and M. E. Benalcázar, "Hand gesture recognition using machine learning and infrared information: a systematic literature review," *International Journal of Machine Learning and Cybernetics*, vol. 12, no. 10, pp. 2859-2886, 2021, https://doi.org/10.1007/s13042-021-01372-y.

[2] A. N. Aziz and A. Kurniawardhani, "The Development of Hand Gestures Recognition Research: A Review," *International Journal of ArtificialIntelegence Research*, vol. 6, no. 1, 2022, https://doi.org/10.29099/ijair.v6i1.236.

[3] M. J. Cheok, Z. Omar, and M. H. Jaward, "A review of hand gesture and sign language recognition techniques," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 1, pp. 131-153, 2019, https://doi.org/10.1007/s13042-017-0705-5.

[4] M. Oudah, A. Al-Naji, and J. Chahl, "Hand Gesture Recognition Based on Computer Vision: A Review of Techniques," *Journal of Imaging*, vol. 6, no. 8, p. 73, 2020, https://doi.org/10.3390/jimaging6080073.

[5] D. Sarma and M. K. Bhuyan, "Methods, Databases and Recent Advancement of Vision-Based Hand Gesture Recognition for HCI Systems: A Review," *SN Computer Science*, vol. 2, no. 6, p. 436, 2021, https://doi.org/10.1007/s42979-021-00827-x.

[6] M. Al-Hammadi *et al*.,"Hand Gesture Recognition for Sign Language Using 3DCNN," *IEEE Access*, vol. 8, pp. 79491-79509, 2020, https://doi.org/10.1109/ACCESS.2020.2990434.

[7] C. Nuzzi, S. Pasinetti, R. Pagani, G. Coffetti, and G. Sansoni, "HANDS: an RGB-D dataset of static hand-gestures for human-robot interaction," *Data in Brief*, vol. 35, p. 106791, 2021, https://doi.org/10.1016/j.dib.2021.106791.

[8] H. Liang, J. Chang, I. K. Kazmi, J. J. Zhang, and P. Jiao, "Hand gesture-based interactive puppetry system to assist storytelling for children," *The Visual Computer*, vol. 33, no. 4, pp. 517-531, 2017, https://doi.org/10.1007/s00371-016-1272-6.

[9] M. Oudah, A. Al-Naji, and J. Chahl, "Elderly care based on hand gestures using kinect sensor," *Computers*, vol. 10, no. 1, p. 5, 2021, https://doi.org/10.3390/computers10010005.

[10] M. Z. Alksasbeh *et al.*, "Smart hand gestures recognition using K-NN based algorithm for video annotation purposes," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 21, no. 1, pp. 242-252, 2021, https://doi.org/10.11591/ijeecs.v21.i1.pp242-252.

[11] Y. C. Chu, Y. J. Jhang, T. M. Tai, and W. J. Hwang, "Recognition of hand gesture sequences by accelerometers and gyroscopes," *Appl. Sci.*, vol. 10, no. 18, 2020, https://doi.org/10.3390/app10186507.

[12] M. AL-Rawi, "Automated attendance machine using face detection and recognition system," *Radioelektronika, Nanosistemy, Inf. Tehnol.*, vol. 12, no. 2, pp. 297-304, 2020, https://doi.org/10.17725/rensit.2020.12.297.

[13] M. S. M. Suhaimin, M. H. A. Hijazi, C. S. Kheau, and C. K. On, "Real-time mask detection and face recognition using eigenfaces and local binary pattern histogram for attendance system," *Bull. Electr. Eng. Informatics.*, vol. 10, no. 2, 2021, https://doi.org/10.11591/eei.v10i2.2859.

[14] A. Anand, V. Jha, and L. Sharma, "An improved local binary patterns histograms technique for face recognition for real time applications," *International Journal of Recent Technology and Engineering*, vol. 8, no. 2 Special Issue 7, pp. 524-529, 2019, https://doi.org/10.35940/ijrte.B1098.0782S719.

[15] F. Deeba, A. Ahmed, H. Memon, F. A. Dharejo, and A. Ghaffar, "LBPH-based enhanced real-time face recognition," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 5, 2019, https://doi.org/10.14569/IJACSA.2019.0100535.

[16] N. Kamarudin *et al.*, "Implementation of haar cascade classifier and eye aspect ratio for driver drowsiness detection

using raspberry Pi," *Univers. J. Electr. Electron. Eng.*, vol. 6, no. 5, pp. 67-75, 2019, https://doi.org/10.13189/ujeee.2019.061609.

[17] G. Jenifa, N. Yuvaraj and K. R. Sripreethaa, "Visitor/Intruder Monitoring System using Haar-cascade Classifier Algorithm," *Innovations in Information and Communication Technology*, pp. 229-231, 2020, https://doi.org/10.46532/978-81-950008-1-4_050.

[18] A. Karishma, K. V. Anand Krishnan, A. Kiran, E. M. Dalin, and S. Shivaji, "Smart office surveillance robot using face recognition," *International Journal of Mechanical and Production Engineering Research and Development*, vol. 8, no. 3, pp. 725-734, 2018, https://doi.org/10.24247/ijmperdjun201877.

[19] M. M. Ahsan, Y. Li, J. Zhang, M. T. Ahad, and K. D. Gupta, "Evaluating the Performance of Eigenface, Fisherface, and Local Binary Pattern Histogram-Based Facial Recognition Methods under Various Weather Conditions," *Technologies*, vol. 9, no. 2, 2021, https://doi.org/10.3390/technologies9020031.

[20] O. Mujahid and Z. Ullah, "High Speed Partial Pattern Classification System Using a CAM-Based LBP Histogram on FPGA," *IEEE Embed. Syst. Lett.*, vol. 12, no. 3, pp. 87–90, 2020, https://doi.org/10.1109/LES.2019.2956154.

[21] M. S. Kaushik and A. B. Kandali, "Recognition of facial expressions extracting salient features using local binary patterns and histogram of oriented gradients," *2017 Int. Conf. Energy, Commun. Data Anal. Soft Comput. ICECDS 2017*, pp. 1201–1205, 2018, https://doi.org/10.1109/ICECDS.2017.8389632.

[22] M. P. Anggadhita and Y. Widiastiwi, "Breaches Detection in Zebra Cross Traffic Light Using Haar Cascade Classifier," *Proc. - 2nd Int. Conf. Informatics, Multimedia, Cyber, Inf. Syst. ICIMCIS 2020*, pp. 272–277, 2020, https://doi.org/10.1109/ICIMCIS51567.2020.9354275.

[23] S. Dawn, S. Tulsyan, S. Bhattarai, S. Gopal, and V. Saxena, "An Efficient Approach to Image Indexing and Retrieval Using Haar Cascade and Perceptual Similarity Index," *2020 6th Int. Conf. Signal Process. Commun. ICSC 2020*, pp. 108–113, 2020, https://doi.org/10.1109/ICSC48311.2020.9182741.

[24] T. Q. Vinh and N. T. N. Anh, "Real-Time Face Mask Detector Using YOLOv3 Algorithm and Haar Cascade Classifier," *Proc. - 2020 Int. Conf. Adv. Comput. Appl. ACOMP 2020*, pp. 146–149, 2020, https://doi.org/10.1109/ACOMP50827.2020.00029.

[25] V. Aarthi and B. Harini Gauri, "Prototype Design of Intelligent Traffic Signal Control using Haar Cascade Classifier," *2021 Int. Conf. Wirel. Commun. Signal Process. Networking, WiSPNET 2021*, pp. 260–264, 2021, https://doi.org/10.1109/WiSPNET51692.2021.9419366.

[26] T. Adep, R. Nikam, S. Wanewe, and D. K. B. Naik, "Visual Assistant for Blind People using Raspberry Pi," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 3307, pp. 671–675, 2021, doi: 10.32628/cseit2173142.

[27] F. Salih and M. S. A. Omer, "Raspberry pi as a Video Server," *2018 Int. Conf. Comput. Control. Electr. Electron. Eng. ICCCEEE 2018*, pp. 1–4, 2018, https://doi.org/10.32628/CSEIT2173142.

[28] D. Dube, "Lossless compression of grayscale and colour images using multidimensional CSE," *International Symposium on Image and Signal Processing and Analysis (ISPA)*, pp. 222–227, 2019, https://doi.org/10.1109/ISPA.2019.8868744

[29] S. K. Chari, A. Gupta, P. Gupta, and J. Mohan, "Threshold selection in image segmentation using parametric entropy measures," *Fourth International Conference on Image Information Processing (ICIIP)*, pp. 1-5, 2018, https://doi.org/10.1109/ICIIP.2017.8313724.

[30] R. Kheddam, A. Tahraoui, and A. Belhadj-Aissa, "Land Change Detection in Bitemporal and Multispectral Images Using Multivariate Alteration Detection and Statistical Thresholding Methods," *2018 9th International Symposium on Signal, Image, Video and Communications (ISIVC)*, pp. 300–305, 2018, https://doi.org/10.1109/ISIVC.2018.8709197.

[31] W. Wang, H. Sheng, S. Liu, Y. Chen, J. Wan, and J. Mao, "An Edge-Preserving Active Contour Model with Bilateral Filter based on Hyperspectral Image Spectral Information for Oil Spill Segmentation," *2019 10th Workshop on Hyperspectral Imaging and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pp. 1–5, 2019, https://doi.org/10.1109/WHISPERS.2019.8921042.

[32] S. R. Chowdhury, S. Pathak, and M. D. A. Praveena, "Gesture Recognition Based Virtual Mouse and Keyboard," *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, pp. 585–589, 2020, https://doi.org/10.1109/ICOEI48184.2020.9143016.

[33] A. P. Nemirko and L. A. Manilo, "Machine learning methods based on convex hulls analysis in medical diagnostics tasks," *2019 International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON)*, pp. 0596-0600, 2019, https://doi.org/10.1109/SIBIRCON48586.2019.8958383.

[34] H. Cevikalp, H. S. Yavuz, and B. Triggs, "Face Recognition Based on Videos by Using Convex Hulls," In *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 12, pp. 4481–4495, 2020, https://doi.org/10.1109/TCSVT.2019.2926165.

[35] S. C. Mesbahi, M. A. Mahraz, J. Riffi, and H. Tairi, "Hand gesture recognition based on convexity approach and background subtraction," *2018 International Conference on Intelligent Systems and Computer Vision (ISCV)*, pp. 1–5, 2018, https://doi.org/10.1109/ISACV.2018.8354074.

[36] T. Zhang, J. Zong, and Q. Liu, "A segmentation processing method for effectively solving the problem of partial covering of angiography images.," *2020 International Conference on Computer Vision, Image and Deep Learning (CVIDL)*, pp. 197–200, 2020, https://doi.org/10.1109/CVIDL51233.2020.0-102.

[37] D. Yang, A. Alsadoon, P. W. C. Prasad, A. K. Singh, and A. Elchouemi, "An Emotion Recognition Model Based on Facial Recognition in Virtual Learning Environment," *Procedia Computer Science*, vol. 125, pp. 2–10, 2018, https://doi.org/10.1016/j.procs.2017.12.003.

[38] J. Rajevenceltha, V. H. Gaidhane, and V. Anjana, "A novel approach for Drowsiness Detection using Local Binary Patterns and Histogram of Gradients," *2019 International Conference on Electrical and Computing Technologies and Applications (ICECTA)*, pp. 1–6, 2019, https://doi.org/10.1109/ICECTA48151.2019.8959669.
[39] Y. A. L. Alsabahi, L. Fan, and X. Feng, "Image classification method in DR image based on transfer learning," *2018 Eighth International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pp. 1–4, 2019, https://doi.org/10.1109/IPTA.2018.8608157.
[40] G. Ramesh, W. Srihari, A. M. Bharadwaj, and H. N. Champa, "Kannada ImageNet: A Dataset for Image Classification in Kannada," *2021 International Conference on Computer Communication and Informatics (ICCCI)*, pp. 27–30, 2021, https://doi.org/10.1109/ICCCI50826.2021.9402356.
[41] A. Thakral and A. Vohra, "Comparison between local binary pattern histograms and principal component analysis algorithm in face recognition system," *2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon)*, pp. 973–978, 2018, https://doi.org/10.1109/SmartTechCon.2017.8358516.

BIOGRAPHY OF AUTHORS

Helfy Susilawati, received the Bachelor's Degree in physical education from Universitas Sunan Gunung Djati Bandung, Bandung, Indonesia, in 2011, and Master's Degree in Electrical Engineering from Institut Teknologi Bandung, Bandung, Indonesia, in 2014. She is currently a lecturer in Universitas Garut, Indonesia. Her research interests include image processing, computer vision, and artificial intelligence.
Email: helfy.susilawati@uniga.ac.id

Fitri Nuraeni received a Bachelor's degree in Informatics Engineering from STMIK Tasikmalaya, Indonesia, in 2010, and a Master's Degree Informatics Engineering from Dian Nuswantoro University, Semarang, Indonesia, in 2012. She is currently a lecturer at the Institut Teknologi Garut, Indonesia. Her research interests include data mining, expert systems, cryptography and decision support systems.
Email: fitri.nuraeni@itg.ac.id