Room Security System with Face Recognition using Local Binary Pattern Histogram Algorithm based on the Internet of Things

Turkhamun Adi Kurniawan[1*], Istiqomah Sumadikarta[2], Sukarno Bahat Nauli[3], Faizal Zuli[4], Teguh Budi Santoso[5], Muhammad Roufiqi Desma[6]

Faculty of Engineering, Satya Negara Indonesia University, Jakarta, Indonesia.

Emails: t.adikurniawan@usni.ac.id (Corresponding author), istiqomah.sumadikarta@usni.ac.id, Sukarnobahat@gmail.com, 4faizal.zuli@yahoo.com, , teguh.bs@usni.ac.id, roufiqi33@gmail.com

Received: 25 February 2023 Revised: 26 March 2023 Accepted: 11 May 2023

ABSTRACT:

An Internet of Things-based security system and OpenCV technology have been developed to improve the efficiency and ease of monitoring video footage from CCTV. The face detection process is carried out using the Haar Cascade method, while facial recognition is carried out using the Local Binary Pattern Histogram algorithm. The test results show that light intensity has a significant influence on system accuracy, but this system provides convenience in monitoring CCTV video in real-time through a webserver and improves security, especially in rooms by utilizing Internet of Things technology. The current facial recognition success rate is 72%. Therefore, for the subsequent development of the system, it is recommended to increase the success rate of facial recognition and also implement the File Transfer Protocol to ensure better and better system performance.

KEYWORDS: Security system, Internet of Things, OpenCV, Face Detection, Facial Recognition, Light Intensity, Webserver, File Transfer Protocol.

1. INTRODUCTION

Along with the times and the development of technology, the need for fast information is needed in various sectors of life, especially in the aspect of security. Personal safety and the surrounding environment are everyone's desire. One type of crime that often occurs is the theft of goods, especially in rooms that are easily accessible by many people such as the campus library, Dean's Room, PUSTIKOM, and off campus such as PT. Handa Teknik. These rooms have the same characteristics, which are equipped with CCTV cameras to monitor and maintain room security. However, monitoring through CCTV camera video recordings often creates great difficulties and effort, because you have to watch videos manually and update non-essential parts to find the information needed.

The Mini Office Lab Room contains valuable items such as electronic goods and others that require security protection. Therefore, we need a security system that can overcome this problem. This becomes the basis for research that will discuss the development of security systems. Even though there are many existing room security systems, a system capable of providing real-time data information and sending it to a database server through the support of an internet connection to record the faces of people who enter the room and play CCTV camera video recordings without having to watch videos manually, is still very difficult.

The purpose of doing this research is to build a security system in a room by developing how a mini computer operating system works using a Raspberry Pi 3B on the hardware used and WebCam as a camera function in identifying every face that enters the room. Researchers will use a webcam camera that will be placed in front of the entrance. Closed Circuit Television (CCTV) IP cameras will be used as a video recording device which will record video for 24 hours.

The method to be used in this study is the Haar Cascade classification method in the face detection process and template matching using the Local Binary Pattern Histogram (LBPH) algorithm in the human face recognition process. Where this research will support the concept of the Internet of Things (IOT) so that data read by the Raspberry Pi can be connected to a web server

Paper type: Research paper
DOI: 10.30486/mjee.2023.1984928.1120
How to cite this paper: T. Adi Kurniawan, I. Sumadikarta, S. Bahat Nauli, F. Zuli, T. Budi Santoso and M. Roufiqi Desma, **"Room Security System with Face Recognition using Local Binary Pattern Histogram Algorithm based on the Internet of Things"**, *Majlesi Journal of Electrical Engineering*, Vol. 17, No. 2, pp. 159-163, 2023.

and stored in a MySQL database to read monitoring data in real time (short time).

Based on the background described above, the authors chose the title "Room Security System with Facial Recognition Using the Internet of Things-Based Local Binary Pattern Histogram Algorithm (Case Study: Mini Office Lab Room)".

2. RELATED WORKS

The system is a collection of elements in the form of data, a network of interconnected procedures, human resources, technology, both hardware and software that interact with each other as one unit to achieve the same specific goals/objectives[1]. Etymologically, security comes from the Latin, namely securus (se and cura) which means freedom from danger and freedom from fear [2]. Face Recognition is a method in technology with the process of recognizing faces that are applied to existing technology [3].

The Local Binary Pattern (LBP) algorithm was introduced by Ojala in 1996, where the basic concept of LBP is to infer the local structure of an image by comparing each pixel with the surrounding pixels [4]. The histogram is a method of adjusting the contrast of facial images. The histogram value is obtained by averaging the spread of pixel values in each image so as to improve the overall contrast [5]. Internet of Things is a structure that provides a limited identity object with the ability to move data or information through a network without requiring a two-way relationship between human to human (source to destination) or human to computer interaction[6].

3. SYSTEM ARCHITECTURE DESIGN

At the system architecture design stage, it is proposed to create a security system that initially only is relied on CCTV cameras to become the "Internet of Things" by using a webserver as a data storage and receiving data sent according to data requests

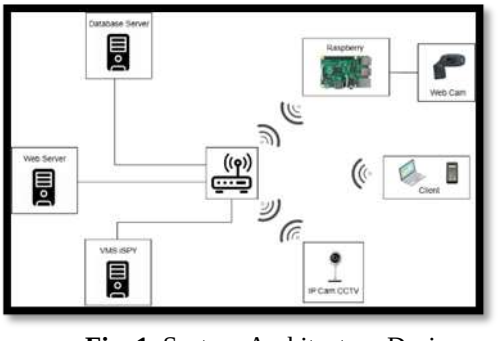

Fig. 1. System Architecture Design.

Someone accesses a room and approaches the WebCam camera that has been installed in the room. This WebCam camera will capture the appearance of the person's face which has or has not been previously registered on the Raspberry PI. Raspberry will perform face detection using the Python programming language and the Haar Cascade method. This method will look for unique features such as eyes, nose, and other areas of the human face. After the face is detected, Raspberry will perform facial recognition using a template matching method called Local Binary Patterns (LBPH). LBPH will compare registered or unregistered face data with existing training data. Recognized faces will be recorded and sent by Raspberry PI using the Message Queue Telemetry Transport protocol called publisher.

Publisher sends messages containing data consisting of id, name, NIM, and time through a Transmission Control Protocol or Internet Protocol network using a router. Messages received by the webserver database are referred to as data collections. Data processing is carried out using the PHP programming language, because the data is still raw and requires processing. Data processing performs filtering of data that is considered the same based on name and time. The webserver acts as a broker and processes the data received from the publisher. Processed data is stored in the webserver database, by separating data collection and data processing.

The admin requests data that has been processed on the webserver via a laptop or smartphone to be displayed in a browser such as Google Chrome. Data requests made by the admin are called subscribers. A laptop or smartphone displays the requested data, which is called data visualization. Data visualization can be in the form of graphics and text containing information on the time the face was detected and the identity of the name of the face being detected. Video recordings from CCTV are received using the iSpy Video Management System software according to the time requested by the admin. Information containing data that has been processed is compared with the results of video recordings from IP CCTV cameras.

The admin takes action by clicking on the time and name, so the video playback application will play the video according to the time chosen by the admin. This allows the admin not requiring to watch the CCTV footage as a whole, but only to choose to see at a certain time which is processed from the data.

3.1. Face Recognition Architecture Design

In the early stages of the facial recognition process on the system, he program will run the webcam to capture the user's facial image, then save it as an image in a dataset with face ID based on the ID entered by the user. The next stage is to carry out the training process so that the system can recognize the user's face based on that ID. The training output from the dataset is saved as a .yml file. Then, the system performs real-time facial recognition using the Haar feature. The system will compare the Haar feature in the program with the facial

object detected by the camera. After that, the system will make predictions by reading the training results from the .yml file and determining which user faces are detected. If the detected face is the user's face, the name, age, and gender will be written on the image display according to the recognized face

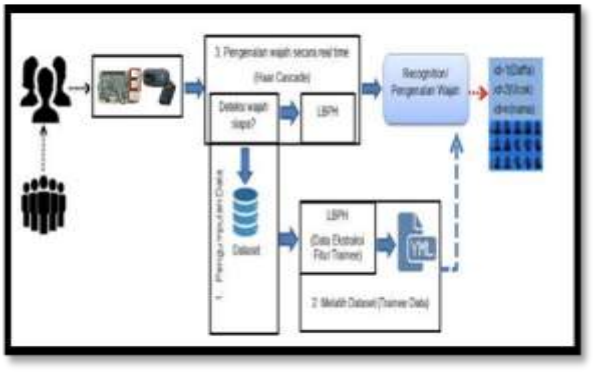

Fig. 2. Face Recognition Architecture Design.

3.2. Test Parameters

Parameter testing will be carried out to be able to display the system with the intention of finding errors in a system, before the system is given to users. Testing this system is necessary to determine the level of accuracy of the system being designed.

Table 1. Test Parameters

NO	System Testing
1	Face Detection with four faces simultaneously
2	Face recognition with one person
3	Face recognition using physical photos
4	Face recognition at a distance of one meter
5	Face recognition using images in the confusion

4. SIMULATION AND RESULTS

4.1. Implementation of the Haar Cascade Method in Registration

At this stage, the user must register by entering ID, name, age, and gender for the image label. After registering in text form, the system will then open the webcam camera to perform face detection using the Haar Cascade method. This process involves taking different facial positions, such as facing forward, right, left, up and down. like in the picture below.

Fig. 3. Retrieval of Datasets using the Haar Cascade Method

After inputting data in the form of text and retrieving data in the form of images, the results of inputting text data will be stored in the SQLite database and the resulting data will be stored in the "dataset" folder and training will be carried out on text data and image data.

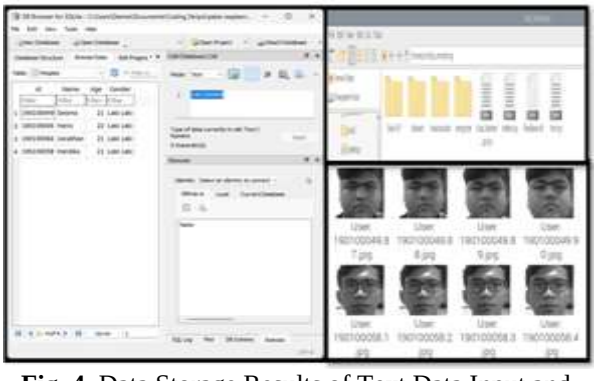

Fig. 4. Data Storage Results of Text Data Input and Image Data Retrieval.

4.2. Implementation of the Local Binary Pattern Histogram (LBPH) Algorithm

After carrying out the registration stage and the data training stage, facial recognition is then carried out using the local binary pattern histogram algorithm. LBP breaks the face into cells and measures the intensity of the pixels in each cell. This intensity is then converted to a binary value and compared with the neighboring pixel intensities. The results of this comparison are then collected into a histogram that describes the unique features of the face. LBPH also measures the euclidean distance between the histograms of the faces to be recognized and the histograms of the faces in the database. If the distance between the two histograms is small enough, then the faces are said to be the same and the face recognition process is complete

Fig. 5. Implementation of the Local Binary Pattern Histogram Algorithm.

4.3. Testing the Facial Recognition System Using Photos

Testing using photos is done using an external WebCam. The aim is to determine the success of the system in detecting faces in photos.

Fig. 6. Testing the Facial Recognition System using Photos.

4.4. Recording Face Recognition Results to MySql Database

After testing the facial recognition system, data recording of facial recognition results will be carried out using the Internet of Things concept. Faces that are detected using the Haar cascade method and recognized using a local binary pattern algorithm that uses the python programming language will save the facial recognition data to the MySql database by requesting PHP code in the python programming. The PHP code will store facial recognition data into the person table.

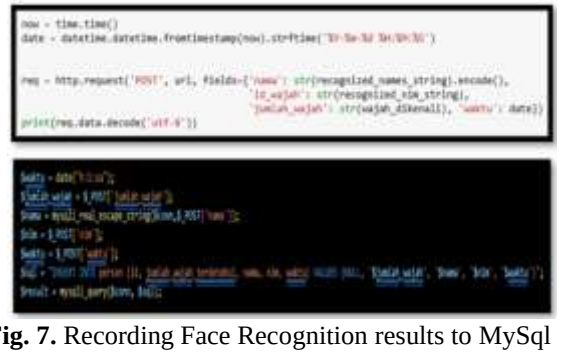

Fig. 7. Recording Face Recognition results to MySql Database.

4.5. Testing in the Mini Office Lab

Testing in the mini office lab is carried out using an external WebCam which aims to determine the success of the system in detecting faces in real conditions. The trials are carried out by entering one, two, or three faces into the room with various movements and are limited by lighting conditions that affect the success rate of the system

Table 2. Test Results in the Mini Office Lab.

No	Test	Expected results	Test results	Status
1	One person: Farrisa	Unknown	Unknown	Succeed
2	One person: Hardika	Hardika	Hardika	Succeed
3	One person: Jonathan	Jonathan	Jonathan	Succeed
4	One person: Desma	Desma	Desma	Succeed
5	Two Perssen: Farrisa, Hardika	Unknown, Hardika	Jonathan, Hardika	Failed
6	Two Perssen: Farrisa, Jonathan	Unknown, Jonathan	Unknown, Jonathan	Succeed
7	Two Perssen: Farrisa, Desma	Unknown, Desma	Unknown, Desma	Succeed
8	Two Perssen: Hardika, Jonathan	Hardika, Jonathan	Hardika, Jonathan	Succeed
9	Two Perssen: Hardika, Desma	Hardika, Desma	Unknown, Not Detected	Failed
10	Two Perssen: Jonathan, Desma	Desma, Jonathan	Unknown, Jonathan	Failed
11	Three Perssen: Desma, Jonathan, Hardika	Desma, Jonathan, Hardika	Desma, Jonathan, Hardika	Succeed
12	Three Perssen: Desma, Hardika, Farrisa	Hardika, Unknown, Desma	Jonathan, Hardika, Desma	Failed
13	Three Perssen: Desma, Jonathan, Farrisa	Desma, Jonathan, Unknown	Desma, Jonathan, Unknown	Succeed
14	Three Perssen: Hardika, Jonathan, Farrisa	Hardika, Jonathan, Unknown	Hardika, Jonathan, Unknown	Succeed

In the table above, the system has been tested four times, consisting of one face test, two face tests, and three face tests. The results obtained from the above test are 9 faces that are detected correctly and 5 faces that are detected with incorrect recognition. Then the percentage of system success can be calculated as follows:

$$\text{Success} = \frac{10}{14} \times 100\% = 72\%$$

$$\text{Error} = \frac{4}{14} \times 100\% = 28\%$$

4.6. Webserver Monitoring Results

The results of web monitoring show the process of data sent to the web, where the data is the result of the user's face detection. Web monitoring is made using php programming commands.

Fig. 7. Webserver Dashboard Page Display and CCTV Video Playback.

The first picture above is a dashboard page web display that displays data in the form of a line chart visualization data. The line chart on the dashboard page displays the percentage or total number of faces that are detected and recognized from the results of processing data. Line charts consist of daily, monthly, weekly, to yearly data which previously have been set by default to display daily, weekly, monthly to yearly data. Then the second picture above shows the appearance of the CCTV video playback web page and matching data contained in the webserver database. The admin takes action by clicking on the time or name in the arrival prediction data table. After clicking, the web will start playing the CCTV video recording which goes directly to the time previously chosen by the admin.

5. CONCLUSION

The room monitoring system using IoT and OpenCV functions properly after testing. Using the Python programming language and the OpenCV library, as well as the face recognition method, the system has a success rate of 72% and an error rate of 28%. This system is connected to a webserver and makes it easy for users to monitor real-time CCTV video and monitor room security. This system provides better and more efficient security by utilizing IoT technology.

REFERENCES

[1] D. Sitinjak, Maman, and J. Suwita, **"Analisa Dan Perancangan Sistem Informasi Administrasi Kursus Bahasa Inggris Pada Intensive English Course Di Ciledug Tangerang,"** *JURNAL IPSIKOM*, Vol. 8, No. 1, 2020.

[2] Z. H. Azizah, **"Mendefinisikan Kembali Konsep Keamanan dalam Agenda Kebijakan Negara-Bangsa,"** *Jurnal Diplomasi Pertahanan*, Vol. 6, No. 3, pp. 94–104, 2020.

[3] M. Arsal, B. Agus Wardijono, and D. Anggraini, "**Face Recognition Untuk Akses Pegawai Bank Menggunakan Deep Learning Dengan Metode CNN,"** *Jurnal Nasional Teknologi dan Sistem Informasi*, Vol. 6, No. 1, pp. 55–63, Jun. 2020, doi: 10.25077/teknosi.v6i1.2020.55-63.

[4] L. W. Alexander, S. R. Sentinuwo, and A. M. Sambul, **"Implementasi Algoritma Pengenalan Wajah Untuk Mendeteksi Visual Hacking,"** *Jurnal Teknik Informatika*, Vol. 11, No. 1, pp. 1–8, 2017.

[5] R. Purwati and G. Ariyanto, **"Pengenalan Wajah Manusia berbasis Algoritma Local Binary Pattern,"** *Emitor: Jurnal Teknik Elektro*, Vol. 17, No. 2, pp. 29–38, 2017, doi: 10.23917/emitor.v17i2.6232.

[6] Y. Efendi, **"Internet of Things (Iot) Sistem Pengendalian Lampu Menggunakan Raspberry Pi Berbasis Mobile,"** *Jurnal Ilmiah Ilmu Komputer*, Vol. 4, No. 1, 2018, [Online]. Available: http://ejournal.fikom-unasman.ac.id.