SPAM (Smart Patient Monitoring System) using Structural Similarity Index Measurement

A Karima[*1], A N A Thohari[2], F Abdollah[3], S Fahriah[4], P Rahardjo[5], W Sulistiyo[6], Sukamto[7]

[1,2,4,5,6,7]Politeknik Negeri Semarang, Indonesia

[3]Universiti Teknikal Malaysia Melaka, Malaysia

E-mail: aisya.karima@polines.ac.id[1]

Submitted: 5 November 2022, revised: 24 January 2023, accepted: 14 February 2023

Abstract. The number of patients in hospitals during the pandemic covid-19 has increased significantly which causes patients to not get the optimal service because of the limitation of human resources. Furthermore, they need tools to detect humans in patients' rooms and monitor the movement of people. IoT is capable to control the room properly. Regarding these problems, the aim of this research is to develop SPAM *(Smart Patient Monitoring System)* which implements the *Internet of Things (IoT)* to control the patient in the hospital using Raspberry Pi. Those data are real-time and completed by notification via telegram. Consequently, if there are emergencies they can observe easily. This study applied *Structural Similarity Index Measurement* (SSIM) technique by comparing different images on several consecutive frames of video by Raspberry Pi. The research methodology was instrument preparation, system designing, data processing, testing, and evaluation. The experiment proved that the system effectively spotted human things accurately captured on camera in more than 15 trials. Although there was a delay of between 5 and 40 seconds, notifications were also correctly transmitted. The system correctly recognized the object when the light was bright with lux > 100, and able evaluating the level of light intensity at a distance of 50 cm to 300 cm. However, when the light level was below 20 lux, the system was unable to identify the object. The system could still detect in low-light situations at a distance of less than 200 cm but beyond that, it was unable to identify human objects.

Keywords: Security, Internet of Things, Hospital, SSIM, Raspberry Pi

1. Introduction

A monitoring system is one of the necessities in life that increase rapidly because there are several implemented tools used to improve security and productivity. Public services which use this service are banking, warehousing, offices, and various public service facilities such as stations, terminals, hospitals, and other public facilities. One of them is a security system to control patients in hospitals such as regular patient monitoring to record all activities in the patient care room. In the future, if there is something critical or emergency occurs, they can be immediately identified.

CCTV technology is frequently employed as a security measure to monitor public locations. However, it has limitations. It is used for surveillance [1], one of which involves identifying a person in a certain location. One of the drawbacks of CCTV is that the operator must continuously monitor the footage and manually recognize any familiar or unfamiliar faces. Another drawback is that since surveillance is ongoing, CCTV cameras will produce and store a lot of data [2]. Building an intelligent system that can recognize moving objects as an alternative to CCTV work could lead to an automatic data control and surveillance system.

For instance, during the pandemic, physical distancing is required in whole places, especially in hospitals. This is to protect suspect patients not to transmit the virus to other people in the same place. The number of nurses in hospitals is not equal to the number of patients. Consequently, a nurse handles several patients in different rooms in different buildings. Therefore, a patient monitoring system is needed to make sure the patient is safe.

Regarding those problems, this study proposed SPAM (Smart Patient Monitoring System) which implements the Internet of Things (IoT) to control the patients in hospitals using Raspberry Pi. Those data are real-time and completed by notification via telegram. Consequently, if there are emergencies the patients can be monitored easily. This study implemented Structural Similarity Index Measurement (SSIM) technique by comparing different images on several consecutive frames of video by Raspberry Pi.

2. Related Work

Several studies are using indoor security systems, including research [3] which states that a home security system is built using Arduino Uno, magnetic door switch sensor, and sim808. A house security system is constructed utilizing an Arduino Uno, a magnetic door switch sensor, and a SIM808. This is stated in one study that uses indoor security systems [3]. When the door is opened, the system will beep and send an SMS notification. A study discussed other implementations of home security to protect swallow nests that are developed with motion detection and notification via SMS Gateway [4]. In addition, research [5] utilizes a PIR sensor as a human motion detector and a camera to record movement with a Raspberry Pi-based main device. The research provides a solution to the main issue of power consumption. Another home security system [6] utilizes the parallel port of the computer and the light sensor. In an online-based home security system with the use of mobile devices [7] it is stated that security monitoring is carried out through mobile devices. There is no image processing technique for recognizing human things, according to the numerous studies that have been discussed. Additionally, the accuracy of the test is invalid because it solely takes into account the sensor's sensitivity.

Cameras are used in place of sensors to detect things in research on intelligent surveillance systems for the Internet of Things applications [8]. The camera's job is to record footage, which is then sent to the Raspberry Pi for background subtraction processing. The benefit of this monitoring system is that when a moving object is discovered, notifications may be sent via the Telegram program. The telegram notification includes a video clip of the object that is photographed. In another case [9] the vehicle unit receives the data from the IoT-based Raspberry Pi micro-controller through the RF transmitter, which immediately prevents the vehicle from using the driving circuit to control the relay. Numerous open-source communities, where the Raspberry Pi is primarily employed in Multi-Utility/Multi-functionality Robotic applications, would benefit from this effort [10].

A study of [11] suggests using the SSIM (Structural Similarity) approach to identify the existence of items. The idea of SSIM with Frame Difference and Image Subtraction is similar in that both count the number of pixel differences. In addition, [12] presents a method for obtaining symmetric similarity matrices from regional histograms of grey matter volumes calculated from T1-weighted MRI scans. According to [13], semantic and structural similarity analyses make up the suggested method. Semantic similarity is determined using lexical information in the class diagram, whereas structural similarity is calculated using the diagram's structure, ignoring its lexical information. A study of [14] has developed a set of advice on how to use SSIM most successfully based on investigations and trials, including ways to reduce its computational burden.

The Internet of Things (IoT), a distributed platform, can be used to link medical devices via the internet to get around these restrictions. With this method, data from many sources can be combined to more accurately diagnose the health of the patient and propose potential preventive measures [15].

The Internet of Things (IoT) has been instrumental in the healthcare sector's transition to the next generation of patient care in the area of interconnected medical equipment and sensors [16]. Additionally, since the population of the aged and crippled rises daily, there is a pressing need for a healthcare infrastructure to treat them and prevent needless and avoidable deaths.

Based on this background, this study proposes an intelligent system called SPAM (Smart Patient Monitoring System) with Raspberry Pi for monitoring patients in hospitals. This is carried out as one of the hospital's security procedures in the patient care area. This technology for tracking human mobility is crucial for keeping track of people entering and exiting a room. As for how this method differs from earlier investigations, it uses the ESO32Cam gadget to monitor in real-time and provides telegraph notifications if a human item is found. Additionally, human items are more accurately detected. Therefore, the hospital can quickly monitor it in the event of an emergency involving the patient's room.

3. Method

The research method in this research consists of several stages given in Figure 1 below.

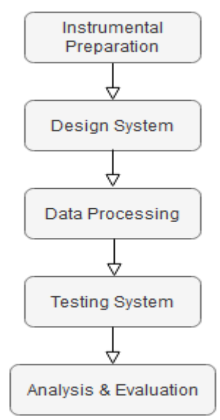

Figure 1. Research Methodology

Following is the process of this research methodology.

3.1. Instrumental Preparation

The tools and instruments used in this research consisted of hardware and software. Those were personal computers or laptops with a minimum specification of RAM of 12 GB, Processor Intel Core i7, and a hard disk of 60 GB space. The other hardware was Single Board Computer (Raspberry Pi Versi 4 Type B), Raspberry Pi Camera Modul V2, 22-inch monitors, MicroSD 128 GB, LAN cables, Power Adapter 5V 2.5A, Power Bank, Raspberry Pi Case, and USB Wifi. For the software/library, this study used Python 3.8, OpenCV 3.5, and Android Arduino Uno.

3.2. Design System

Video data captured by the camera was used as input data for the system developed in this study. The single-board computer processed the video data (SBC). At SBC, the structural similarity method was applied to process images using OpenCV and Python. The output of video processing took the form of object identification, which was determined by the variation in pixel values between the first and subsequent frames. The user then received a notification when the system recognized an object.

3.3. Data Processing

Videos captured by the camera in real-time were the type of data handled in this study. Then, the system would identify the video data to detect moving objects. Video data from the hospital patient room and image data of human objects were the two types of data that were processed for this investigation. Raspberry Pi, an adapter, a camera, and other gear were used in this study. The video data were used to monitor the safety of people going in and out of the room's patient and control their health condition. This was done during the pandemic when only a few people could access that room to protect and sterilize the place. The camera was located on the entrance door near the bed. Therefore, there was a guarantee for the privacy of the patient.

The image consisted of pixels (picture elements), while the grayscale image consisted of 2-dimensional elements. There were three-dimensional components in the color image. A 3x3 matrix, for instance, represented an image with 3x3 pixels. Therefore, equation (1) was employed to determine the total pixel value.

$$A = \begin{bmatrix} 0 & 100 & 50 \\ 0 & 60 & 60 \\ 0 & 200 & 50 \end{bmatrix}$$

The value of all pixels could be calculated as follows:
S = (0 + 100 + 50) + (0 + 60 + 60) + (0 + 200 + 50)
Consequently, the following equation could be used to get the total number of pixels in an image.

$$S = \sum_{i,j} A_{i\,j} \tag{1}$$

By comparing the values of every pixel from the first image to the following image, the value of image similarity could be calculated. Sum Square Error (SSE) and structural similarity were two techniques for comparing image similarity (SSIM).

3.3.1. Sum Square Error (SSE)

The total difference between the achieved value and the actual value was measured using the statistical method known as SSE. The Summed Square of Residuals was another name for SSE [17]. The equation of SSE is shown in a two-dimensional picture made up of the matrices I and J as follows.

$$SSE = \sum_{i,j} (A_{ij} - B_{ij})^2 \tag{2}$$

Where,
A = Value of the reference image and B = Value of new image

In this study, the value of A was the initial value saved as a reference to be compared with the value of B as a part of the test data. When forecasting an observed model, an SSE value that was near 0 implied that the model had the smallest random error component. Keep in mind that SSE was previously described in terms of the minimum squares' feasibility approach.

3.3.2. Structural Similarity Index Metrics (SSIM)

SSIM is a technique for calculating how similar two photos are to one another. SSIM measurement using the Bicubic Interpolation method produces a higher SSIM value compared to other methods [17]. SSE is excellent at identifying differences in inaccuracy, but the flaw with SSE is that it examines each pixel separately. In this study, SSIM analyzed two images as a set of pixels, which was not the case with SSE. It was more efficient to compare photos based on pixel groups because there were fewer errors when there was noise.

$$SSIM\ (A,B) = \pounds(A,B)\ .\ c(A,B)\ .\ s(A,B) \tag{3}$$

Where
A = Reference image
B = New image
£ = Luminance
c = Contrast
s = Structure

It was preferable to portray an image as a group of pixels as opposed to viewing each pixel separately. It was clear that the photographs were distinct after examining ones with various contrasts, lighting, and structural elements. For the SSIM index value, there were upper and lower bound values.

$$-1 \leq SSIM \leq 1 \tag{4}$$

The resemblance of the image was fairly strong if the SSIM index value was close to 1. However, the image had a rather big difference if the SSIM index value was close to -1.

3.4. Testing System

The effectiveness of the hardware and software design process was evaluated through tests. If the results acquired were not satisfactory or do not run in accordance with what has been decided, design changes are made.
On the other hand testing is then repeated until the results can be deemed satisfactory or run in accordance with what has been decided.

The outcomes of video data processing to identify objects recorded by the camera were the parameters evaluated in this study. Due to changes in similarity metrics between frames, objects could be recognized. When the object was located, the system would alert the customer by sending a telegram message to their cell phone. The system was fixed by changing the program code if it still could not recognize a moving object or display a notification.

3.5. Analysis and Evaluation

This research and evaluation compared video data retrieval depending on the amount of light the camera was exposed to. Otherwise, the delay between detected objects and sending notifications via telegram was analysed.

4. Result and Discussion

4.1. Developing System

The first step in conducting this research was to gather the materials and tools that were discussed in the previous chapter. The necessary hardware was built into the laptop for additional testing. The documentation for connecting the ESP32 camera to a laptop is shown in Figure 2. The next process was software installations including Python and Open CV.

Figure 2. ESP32 Camera connection

This system received input in the form of video data of human movement that was recorded by the camera. This instance used the Camera ESP32. On the Raspberry Pi device, the video data was handled using a Single Board Computer (SBC), which was followed by image processing using OpenCV and Python using the Structural Similarity approach.

The output of video processing took the form of motion object identification, which was determined by the difference in pixel values between the first and second frames. The user would receive a telegraph notification when the system recognized a captured object. Figure 3 displays the system design used in this study.

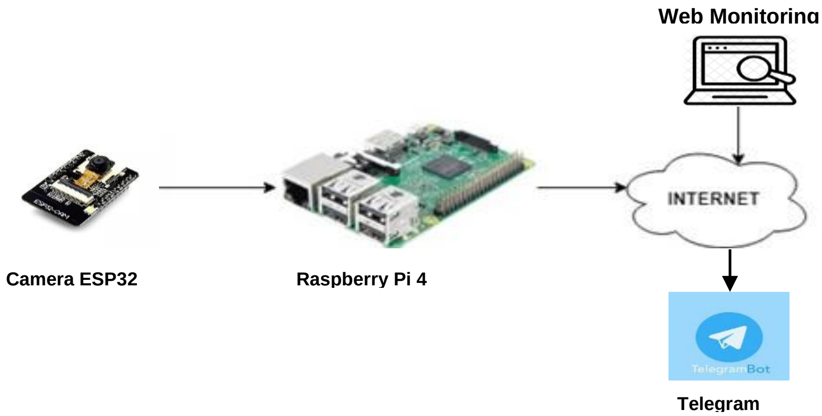

Figure 3. Design System

The first step in the system design process was making sure the port was correct by opening the Arduino IDE app and checking it under Tools > Port > COM 4. Then, the Arduino files were uploaded into the Camera ESP32. The next process was opening PyCharm and clicking RUN. Lastly, a verification of the Local IP on the Camera ESP32 and the IP on PyCharm was conducted as shown in Figure 4.

```
import cv2
from cvzone.PoseModule import PoseDetector
import pyglet.media
import os
import requests

cap = cv2.VideoCapture('http://192.168.43.44/mjpeg/1') #esp32cam
#cap = cv2.VideoCapture(0) #webcam

if not cap.isOpened():
    print("Camera can't open!!!")
    exit()
```

Figure 4. IP Configuration

Here is the test result after launching the program. The system was able to detect human objects captured by ESP32 Camera. After detecting a human object, the alarm rang and a notification appeared via telegram as shown in Figure 5.

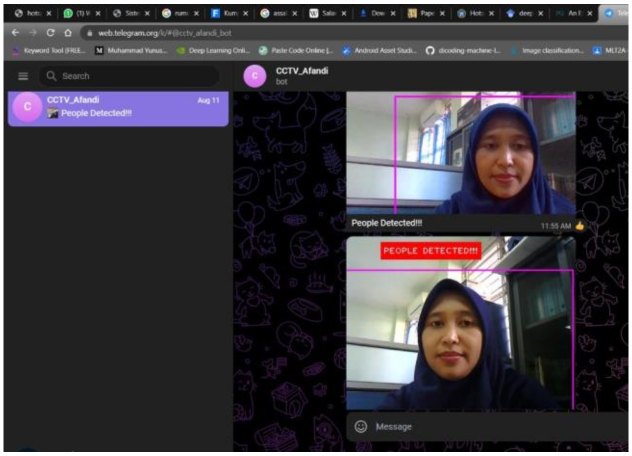

Figure 5. Bot Telegram Notification

In terms of setup and configuration, the Raspberry Pi device was used to process video files obtained from the ESP32 camera. After breaking the video down into frames, the video file was converted into a jpg file format. All captured videos were kept on the Raspberry Pi according to the file's output.

4.2. Data Processing

The following processes were conducted to detect objects.

- Firstly, the system took the first frame to be used as a reference.
  The image set's frame was made into a grayscale version. Since the gray image was a 2-dimensional matrix, the objective was to make calculating pixel values simpler.
- The camera connected to the SBC performed video acquisition and generated frames of video sequentially within a certain time.
- Several frames obtained from image acquisition were converted into grayscale images.
- The new grayscale frame was compared with the first frame that served as a reference.
- The SSIM method was used to compare the frames to calculate the similarity index value.
- The system classified an image as different if the similarity index value was less than the 0.9 threshold value. It means that when the camera moved, more objects emerged.
- When an item was declared, the system transmitted a movement signal, notifying the client's smartphone via telegram.

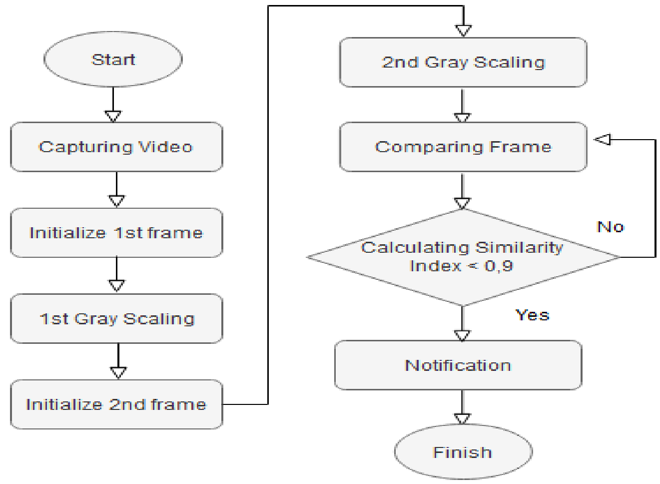

Figure 6. Flowchart

Figure 6 shows the procedure for identifying an object by comparing two frames. Figure 7 shows the experiment's findings.

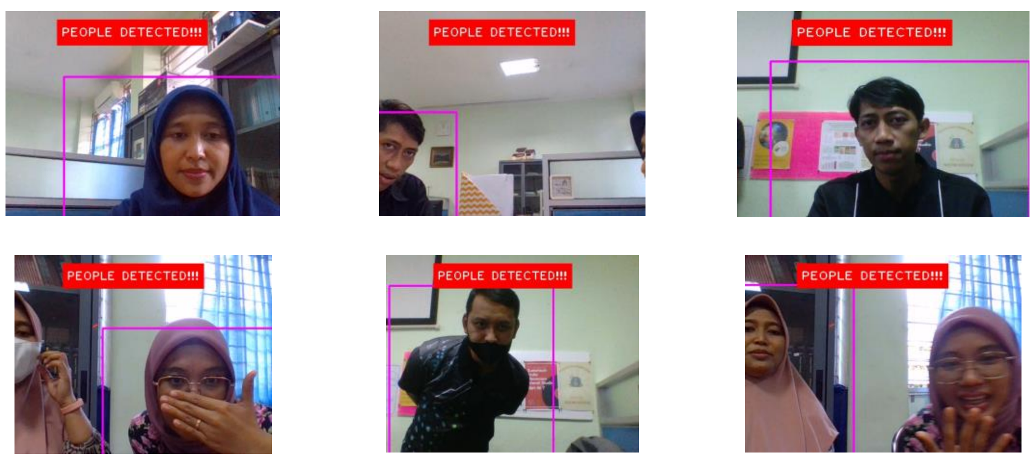

Figure 7. Experimental Result

4.3. Result and Analysis

A notification was provided via the telegram bot in the form of a photo message when the system detected a moving object. The examination of the experimental findings indicated that the SPAM system has been successful in identifying human things captured on camera. Based on the testing process, this system was still unable to distinguish between several objects. For instance, if there were two people in the same place, only one was successfully detected, because the bounding box only capture one object. During the testing procedure, a trial of detecting the presence of human objects was conducted using two different types of tests, including an examination of the delays when delivering Telegram bot alerts and testing of distance and light intensity.

4.3.1. Testing the telegram notification delay

Table 1 shows the findings from testing the Telegram notification delay. The results of 15 experiments showed that SPAM was able to detect objects of people entering the room. Every time an object was spotted, telegram notifications were also delivered correctly, although there was a delay in sending them. The delay period was between 20 and 40 seconds. This was because the provider that was chosen had an impact. The choice of internet service provider causes the speed quality of the data processing capabilities of the Raspberry Pi. The time it took the Raspberry Pi to process the recorded video clip was another important consideration. On the other hand, after trying with another ISP connection, the average delay was reduced to under 30 seconds.

Table 1. The Telegram bot notification delivery delay test results

Testing	Notification Status	Delay
1	Submitted	25 second
2	Submitted	20 second
3	Submitted	27 second
4	Submitted	30 second
5	Submitted	25 second
6	Submitted	25 second
7	Submitted	38 second
8	Submitted	29 second
9	Submitted	33 second
10	Submitted	25 second
11	Submitted	40 second
12	Submitted	36 second
13	Submitted	22 second
14	Submitted	25 second
15	Submitted	20 second

4.3.2. The distance and light intensity testing

Distance and light intensity were two factors that could be used to evaluate an object's appearance. The size of the object would change depending on the distance. The size of an object decreased with distance. Since the camera being utilized was not an infrared camera that could capture objects in the dark, light intensity was also a significant factor in addition to distance.

The criteria used for assessing light intensity were those established by the Illuminating Engineering Society (IES). According to IES, bright light intensity is defined as more than 100 lux, dim light intensity is defined as 20 to 99 lux, and dark light intensity is defined as less than 20 lux [18]. Table 2 below displays the outcomes of evaluating the system's range and light output.

Table 2. Distance and light intensity test result

Distance	Bright >100 lux	Low Light 20 – 99 lux	Dark <20 lux

Distance	**Bright >100 lux**	**Low Light 20 – 99 lux**	**Dark <20 lux**
50 cm	Detected	Detected	Not Detected
100 cm	Detected	Detected	Not Detected
150 cm	Detected	Detected	Not Detected
200 cm	Detected	Detected	Not Detected
250 cm	Detected	Not Detected	Not Detected
300 cm	Detected	Not Detected	Not Detected

Based on the experimental findings, it was concluded that SPAM could detect human objects at a distance of between 50 and 300 cm under a variety of lighting situations. The device could still detect objects up to 200 cm in both high and low-light environments.

However, at a distance of 250 cm and 300 cm, the system can only detect objects when the light is brilliant; in low light, it is unable to do so. On the other side, the system is unable to detect things within a radius of 50 cm to 300 cm when dark light intensity is present.

5. Conclusion

This research has been successful in creating a design to identify items captured by the ESP32 camera through several stages. The examination of the experimental findings revealed that the SPAM system effectively spotted human things accurately captured on camera based on the experimental results of more than 15 trials. Although there was a delay of between 20 and 40 seconds, notifications were correctly sent. This was because of the choice of internet service provider and the data processing capabilities of the Raspberry Pi. On the other hand, after trying with another ISP connection, the average delay was reduced to under 30 seconds. The system correctly recognized an object when the light was bright with the lux > 100 and able evaluating the level of light intensity at a distance of 50 cm to 300 cm. However, when the light level was below 20 lux, the system was unable to identify the object. The system could still detect in low-light situations at less than 200 cm but beyond that, it was unable to identify human objects. In addition, this system was still unable to detect objects that had more than one individual as testing data. This could be the future work of this research.

6. Acknowledgement

We appreciate Semarang State Polytechnic for providing the money for this study. We also want to express our gratitude to the entire team for their unwavering support in putting this research into practice.

References

[1] Pudaruth, I. and B. , "A Unified Intrusion Alert System using Motion Detection and Faces Recognition," in *2nd International Conference on Machine Learning and Computer Science (IMLCS)*, Kuala Lumpur, 2013.

[2] Namrata, P. and S. R. , "Cognitive Security System Based on Image Comparison and Motion Detection with Able Memory Usage," in *International Journal of Advances in Engineering & Technology*, 2013.

[3] Rozi, A. F. and P. , "Home Security Menggunakan Arduino Berbasis Internet of Things," *Jurnal Inovasi Vokasional dan Teknologi,* vol. 18, no. 2, pp. 17-24, 2018.

[4] A. Karima, L. B. Handoko and P. , "Rancangan Sistem Keamanan Rumah Burung Walet Berbasis Motion Detection dan SMS Gateway," *Techno.Com,* vol. 13, no. 1, 2014.

[5] Mekonnen, H. K. and Y. , "SleepyCAM: Power management mechanism for wireless video-surveillance camera," in *IEEE International Conference on Communication Workshops - ICC Workshops*, 2017.

[6] Alfirman, "Penggunaan Port Parallel Komputer Dan Sensor Untuk Sistem Keamanan Rumah," *Jurnal Teknologi Informasi & Pendidikan,* vol. 5, no. 2, pp. 91-101, 2012

[7] Abidin, "Sistem Keamanan dan Monitoring Rumah Pintar secara online menggunakan Perangkat Mobile," *Jurnal Teknik Komputer Universitas Komputer Indonesia,* vol. 3, no. 2, pp. 13-17, 2014.

[8] Harum, A. Z. and A. , "Smart Surveillance System using Background Substraction Technique in IoT Application," *International Journal of Advances Computer Science and Application,* vol. 9, no. 12, pp. 122-128, 2018.

[9] S. Sengan, O. I. Khalaf, P. S. D. K. Sharma, A. K. and A. A. Hamad, "Smart Healthcare Security Device on Medical IoT Using Raspberry Pi," *International Journal of Reliable and Quality E-Healthcare,* vol. 11, no. 3, 2022.

[10] S. E. Mathe, A. C. Pamarthy, H. K. Kondaveeti and S. Vappangi, "A Review on Raspberry Pi and its Robotic Applications," in *International Conference on Artificial Intelligence and Signal Processing (AISP)*, 2022.

[11] R. D. Ramadhani, A. N. A. Thohari and N. A. S. Nugraha, "Sistem Keamanan Ruangan Berbasis Internet of Things Mengginakan Single Board Computer," *InfoTekJar : Jurnal Nasional Informatika dan Teknologi Jaringan,* vol. 4, no. 2, 2020.

[12] M. J. Leming, S. Baron-Cohen and J. Suckling, "Single-participant structural similarity matrices lead to greater accuracy in classifcation of participants than function in autism in MRI," *Leming et al. Molecular Autism* , p. 1, 2021.

[13] R. Fauzan, D. Siahaan, S. Rochimah and E. Triandini, "Automated Class Diagram Assessment using Semantic and Structural Similarities," *International Journal of Intelligent Engineering & System,* pp. 52-66, 2020.

[14] A. K. Venkataramanan, C. Wu, A. C. Bovik, I. Katsavounidis and Z. Shahid, "A Hitchhiker's Guide to Sructural Similarity," *IEEE Access,* pp. 1-26, 2016.

[15] C. A. d. Costa, C. F. Pasluosta, B. Eskofier, D. B. da Silva and R. d. R. Righi, "Internet of Health Things: Toward intelligent vital signs monitoring in hospital wards," Artificial Intelligence In Medicine, vol. 89, pp. 61-69, 2018.

[16] P. K. O, A. Pal, K. Kafle, A. Anand and N. P, "IIoT Based Health Monitoring System," Perspectives in Communication, Embedded-Systems and Signal-Processing (PiCES), vol. 2, no. 5, 2018.

[17] M. Wulandari, "Pengukuran SSIM dan Analisis Kinerja Metode Interpolasi Untuk Peningkatan Kualitas Citra Digital," *Jurnal Muara,* vol. 1, no. 1, pp. 184-195, 2017.

[18] D. DiLaura, The Lighting Handbook, North America: Illuminating Engineering Society of North America, 2011.