Industrial Engineering and Applications
L.-C. Tang (Ed.)

doi:10.3233/ATDE230064

Edge Computing Based on Raspberry PI for People Counting in Smart Office

Khanista NAMEE [a,1], Rudsada KAEWSAENG-ON [b] , Jantima POLPINIJ [c],
Kavin RUEAGRAKLIKHIT [a], Chayapa NIMYUNGDEE [d] and
Somchai CHEINGPONGPAN[a]
[a] *King Mongkut's University of Technology North Bangkok, Thailand*
[b] *Prince of Songkla University, Thailand*
[c] *Mahasarakham University, Thailand*
[d] *Primaham(Thailand) Co., LTD., Thailand*
ORCiD ID: Khanista Namee https://orcid.org/0000-0001-6984-4015

Abstract. This research presents a technique for counting the number of people in a smart office room when walking in and out of the door by relying on a normal quality webcam. This means that the camera is not high resolution. The picture is not clear and sharp. Therefore, the image must be counted to adjust the color first before being processed in other steps. The designed system uses OpenCV and Raspberry Pi to help and is Edge Computing processing by comparing images in front view and top view. The front view is accurate at 58.44% for entering the room and 30.15% accurate for leaving the room. However, it was clear that the top view image would be more accurate at 87.82% for walking into the room and 80.75% out of 130 tests for walking out of the room. This system can also be used for counting people in other situations, such as counting people in factories, counting workers on the job site and counting people at trade fairs, etc.

Keywords. OpenCV, Internet of Things, Smart Office, Edge Computing

1. Introduction

This research uses simple webcams to help people count in smart office systems. Part of smart office systems is that we can automatically know the number of people in a room. Due to it will have a landing effect on adjusting the temperature in the working room. Facial recognition of who is in the office Security and much more [1]. This is the origin of this research in order to automatically count the number of people in the office which if the results of this experiment are satisfactory and practical. We were able to adapt this research to other work systems, such as counting people in a factory. Bringing people to the job site counting people who enter trade fairs, etc [2].

Counting people, there are many different techniques used to count people. Whether it is people counting from motion sensors, people counting from laser sensors, etc. This research presents how to count people in a room from conventional webcam cameras available in the market. That means, it is a cheap camera [3]. Relatively low resolution causing an effect on the image used for processing, it will be less clear as well. In addition, the processing in this system will be processed by the raspberry pi because we want the

[1] Corresponding Author: Khanista Namee, Khanista.N@fitm.kmutnb.ac.th

system to be small, simple, with the limitations of the raspberry pi in terms of processing speed, memory size [4]. The more it takes longer to process. This research is therefore challenging to some extent [5].

In this research, we designed the processing to take place on the raspberry pi for another reason, because we also want to test how efficient it is in edge computing. Can it be applied to work or not? OpenCV is used to help in the work. In this paper, we will explain the detailed steps of the developed system along with showing the results obtained from the experiment by comparing from the actual experiment and running from the clip video file [6, 7].

The rest of this paper is organized as follows. Section 2 provides a brief overview of some theory which had to be implemented within this research. Then, in Section 3, the system and network designs are presented in detail. The experimental deployment is presented in Section 4 to demonstrate the design. The measurement results and performance are presented in Section 4. Finally, conclusions are drawn in Section 5 and following with acknowledgement section.

2. Background

2.1. Image Processing

Image processing refers to the use of images to be processed or calculated by a computer to get the information we need both qualitatively and quantitatively.

The important steps are to make the image sharper. Removing noise from images Segmentation of the object that interests us from the picture. To bring the object image to be analyzed for quantitative data such as size, shape and direction of movement of objects in the picture [8]. We can then analyze these quantitative data and create a system to take advantage of various tasks such as fingerprint recognition system to check who the existing fingerprint image belongs to. Product quality inspection system in the manufacturing process of industrial plants Sorting system for grade or quality of agricultural crops automatic postal code reading system to sort out the destination of a large number of mails each day using a photo of the postal code on the envelope. The system collects data of vehicles entering and exiting the building using photographs of license plates for safety purposes [9]. The system monitors and monitors road traffic conditions by counting the number of vehicles on the road in CCTV footage at each moment. Facial recognition systems for surveillance of terrorists in key buildings or in immigration areas, for example, can be seen that these systems require a large amount of image processing. Moreover, it is a process that has to be repeated in most of the same ways. which works in these ways If the human analysis It is often time consuming and labor intensive. Also, if a large number of images are needed to be analyzed. The image analyzer itself may experience fatigue. result in an error. Therefore, computers play an important role in performing these functions instead of humans. Computers are capable of calculating and processing huge amounts of data in a very short time [10]. It is therefore very useful in optimizing image processing and analyzing data obtained from images in various systems.

2.2. Programming Language for Image Processing

Python is a high-level programming language designed to make code readable by using "lightboxes" or whitespaces to separate lines of code under which block, rather than using other language curly braces ({,}). It is based on low number of lines compared to C++ and Java. It has many features such as self-management of memory, support for multi-paradigm, object-oriented, functional [11]. Python interpreters are compatible with a wide range of OS, meaning they allow the running of python code on many systems today. They are open source software and community-based developers, managed by the Python Software Foundation.

2.3. OpenCV Library

OpenCV is a library developed with support from Intel Corporation Limited as a software. Open source code (Library Open Source) for use in image processing (Image Processing) in order to be able to develop programs easily. It can be used on operating systems that are Linux and Microsoft Windows and can develop programs in a variety of languages. The utilization for the development of computer vision programs (Computer Vision) is capable of processing digital images, both still images and animations such as Image from VDO camera or VDO File is easy [12]. There are ready-made functions for managing image data and basic image processing such as image edging, image filtering.

2.4. Edge Computing

Edge Computing is a network system that requires compute to be located as close to the source of data as possible in order to reduce latency and bandwidth used to send data or to understand it in another way, there are fewer processes running on the cloud and moving those processes to the source instead. Since bringing compute to the network edge reduces the communication distance between the client and the server [13].

A Network Edge is a device that can connect to the internet and is installed within various internet-related devices such as computers or processors in IoT cameras [5, 14]. Also, unlike Physical Server and Cloud Server, no matter where they are installed, they can communicate with the source as shown in Figure 1.

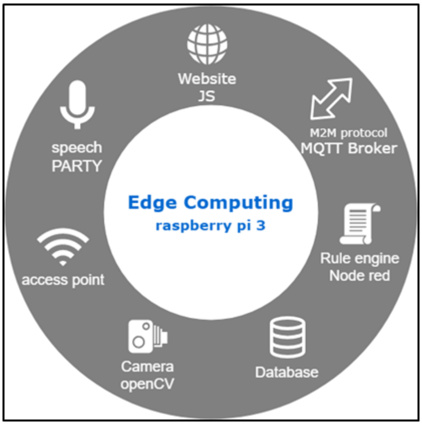

Figure 1. Edge Computing in Smart Office.

2.5. Node Red

Node-RED is a visual programming interface designed for the Internet of Things concept. It requires minimal coding experience and makes it easy to connect physical things to the cloud. Provide tools to make the process much less hassle [15].

Node-RED is an event handling and management tool based on Node.js. Node-RED applications typically run as web servers and users can customize and manage connections between different hardware and build workflows from the base any computer browser, no expensive software. However, it is easy and works in a web browser. This server application also has the ability to run on devices such as the Raspberry Pi.

The whole function of Node-RED is that the user no longer needs to enter all the codes. At least most of the time and works with existing code. Node-RED's user interface is simple and revealing, so users have little trouble developing IoT projects with it. Node-RED streams on GitHub are rendered in JSON (JavaScript Object Notation) and can be exported to the clipboard easily or can be imported into Node-RED or shared online [16].

3. System Design and Architecture

3.1. Using Raspberry Pi with a Camera to Create a Person Counting Program

The Raspberry Pi can work with cameras. We can use python to write processing commands, coupled with image processing libraries like OpenCV that can be used to capture people, capture various movements which may be applied [17]. For creating a program to count the number of people entering and exiting the door until obtaining the net number of the number of persons at that time. Then, perform installation and write commands to try using Raspberry Pi for counting the number of persons.

Camera Installation Procedure is divided into two different methods and will bring the results be compared to choose the best method out of the two methods:

Counting the number of people in front view.
Counting the number of people in a top view.

3.2. Camera Installation Procedure Counting the number of people in front view

Get all the devices ready and then take the Raspberry Pi that is already connected to the camera. Place it in front of the desired door in distance and perspective [18]. The whole door is seen in the center of the picture. Therefore, installing the camera in the front view can be easily and versatile, for example, the rear view in Figure 2.

3.3. Camera Installation Procedure Counting the number of people in top view

Get all the equipment ready and bring the Raspberry Pi connected to the camera already up to install on the ceiling above the door in a manner parallel to the earth. Bring blue tape to stick on the floor to cover the door area [19]. The tape must be blue only and must be longer than the width of the tape so that the program does not detect errors as shown in Figure 3.

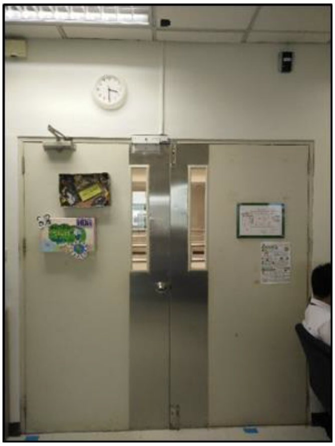

Figure 2. Camera installation in front view.

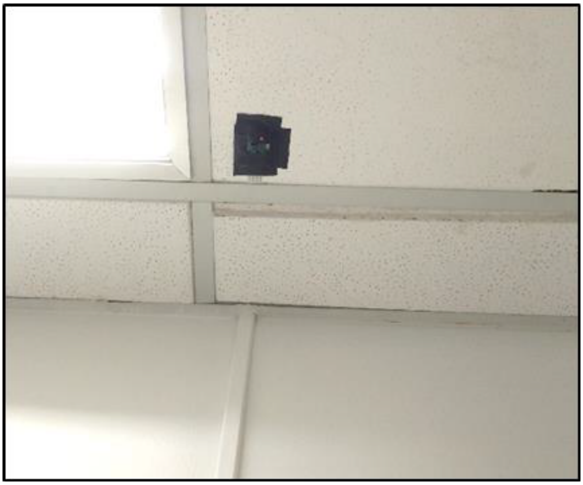

Figure 3. Camera installation in top view.

3.4. Develop a Program to Count People

Once installed, it's up to the process of developing the program. Which will be divided into 2 methods as well, namely.

- Counting the number of people in front view.
- Counting the number of people in a top view.

Program schematics and descriptions for counting the number of people in a top view.

Start the program.
Reading images from cameras or video clips.

Convert the resulting image. It's a black and white picture by making it black only for dark spots to focus attention on the hair.

Read the tape mark on the floor every 2 minutes or start the program for the first 20 frames by calling the checkMarkLine function.

Then detect movement with Background SubtractorMOG2 algorithm which will cut off other parts, leaving only the point where there is movement.
Sort the results by size.

To display the reference line according to the reading position.

Looping frames that can be motion detected.
Do check the size of each point. If the frame size is too large, it would be good to say with changing light conditions or moving the camera function will be called checkMarkLine to redraw the reference line. However, if the frame is too small, the program will ignore it that movement [20].
Calculate the center point of that frame.

Tracking with trackMove function.

Check that the first point. Is it too close to the reference line? To prevent duplicate tracking.

Check the condition of leaving the room. If the first point that appears inside the room and the latest point is outside the room. It will be considered leaving the room. After that, subtract the number of people by 1.

Check the condition of leaving the room. Where if the first point that appears outside the room and the latest point is inside the room. It will be considered to enter the room. After that, add the number of people to 1. Example result as shown in Figure 4.

3.5. Rule Engine

Node-RED is a brand of rule engine. It is a visual programming model. It uses conditional points and then connects lines to make it easier to understand. There are many API integrations and online services to choose from. Raspbian, the operating system for the Raspberry pi, comes pre-installed with Node-RED. This allows users to start using them immediately.

Implementation, design and setting of personnel conditions. By default, it gets the number of people from NETPIE to which the data was sent to the device. Count the number of people, and then check the following conditions in Figure 5.

Is there access to the room after working hours 17:30 – 07:30 Monday to Friday and 24 hours on Saturdays and Sundays or not if the above conditions are met? It will alert the intrusion to the NETPIE mobile app.
It has the room left until 0 people left or not if the conditions are met. It will alert you that no one is in the room and will turn off the lights.
Is the current number of people 1 and before the number of 0? If true, that person is considered to be the first to enter the room. Turn on the lights and air conditioning and notify the first person to arrive.

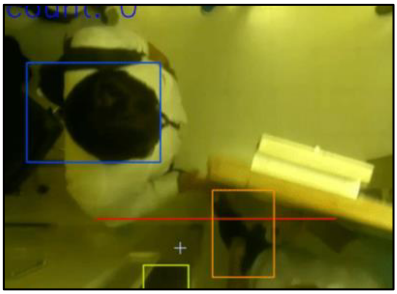

Figure 4. Camera in top view for checking the condition.

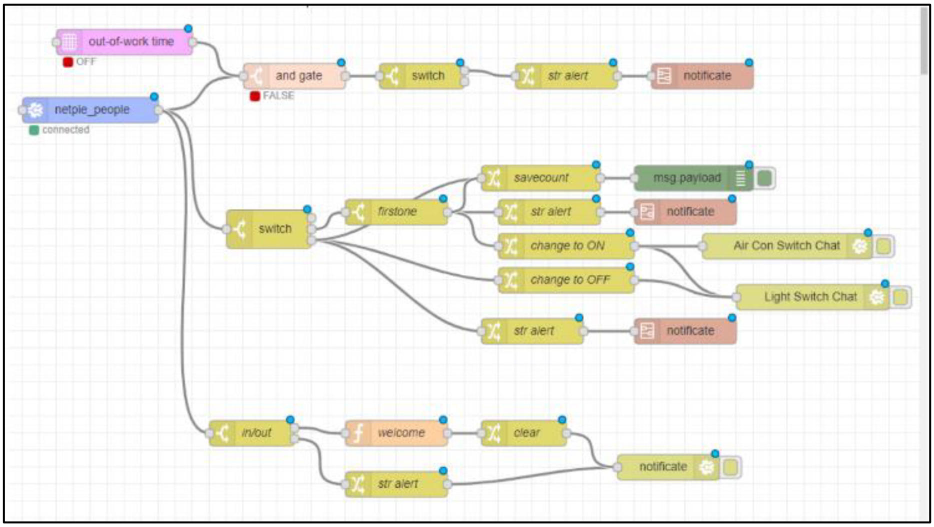

Figure 5. Setting conditions for the number of people.

4. Results

4.1. Test the number of people counting function in a room using a camera

Made by recording videos at different angles in real-life situations and bring the video to be compiled with the program to count the number of people. Then take the results from the program. Let's compare with the actual events in the video, then save the result. The results will be categorized as follows in Table 1:

Table 1. Different types of results from the program as compared to reality

Label	Activity	Result
A	IN	IN
B	EXIT	IN
C	IN	EXIT
D	EXIT	EXIT
E	-	IN
F	-	EXIT
G	IN	-
H	EXIT	-

Table 2. The example results of people counting program in front view with testing video

Video File	Length	Reality			Run Script			Result	Label
		IN	EXIT	Time	IN	EXIT	Time		
File1.mp4	10:17		/	3:18				FALSE	H
		/		4:20	/		4:22	TRUE	A
			/	7:58		/	8:02	TRUE	D
File2.mp4	5:30		/	0:07				FALSE	H
		/		0:29	/		0:29	TRUE	A
		/		0:47	/		0:48	TRUE	A
		/		0:49	/		0:52	TRUE	A
File3.mp4	6:34		/	2:50		/	2:52	TRUE	D
			/	2:53		/	2:59	TRUE	D
File4.mp4	3:02		/	0:37		/	0:41	TRUE	D

Table 3. The example results of people counting program in top view with testing video

Video File	Length	Reality			Run Script			Result	Label
		IN	EXIT	Time	IN	EXIT	Time		
File5.mp4	23:13		/	0:06		/	0:06	TRUE	D
		/		20:17	/		20:18	TRUE	A
File6.mp4	14:28		/	0:39		/	0:44	TRUE	D
			/	2:55				FALSE	H
					/		5:01	FALSE	E
						/	5:15	FALSE	F
File7.mp4	13:17		/	0:08		/	0:11	TRUE	D

4.2. The test of the program to count the number of people in front view

From the test with 51 front view video clips, total length 9 hours 29 minutes, summarizing all results. It is shown in Table 4. In all, the results obtained from the test were separated into two cases: traffic entering the room and traffic out of the room. In each case, the test results can be summarized as follows in Table 2 and 4.

Inbound traffic: A total of 77 walks into the room were counted correctly 45 times, representing 58.44%.

Outbound traffic: A total of 105 walks into the room were counted correctly 41 times, representing 30.15%.

When taking the accuracy obtained from both cases to calculate the average of all accuracy. The total accuracy is 44.29%.

Table 4. Summary of the test of the program to count the number of people in front view

Label	Activity	Result	Number of Times
A	IN	IN	45
B	EXIT	IN	1
C	IN	EXIT	0
D	EXIT	EXIT	41
E	-	IN	31
F	-	EXIT	10
G	IN	-	32
H	EXIT	-	94

4.3. The test of the program to count the number of people in top view

In all, the results obtained from the test were separated into two cases: traffic entering the room and traffic out of the room. In each case, the test results can be summarized as follows in Table 3.

Inbound traffic: A total of 157 walks into the room were counted correctly 137 times. The accuracy of traffic entering the room was 87.82%.

Outbound traffic: A total of 161 walks into the room were counted correctly 130 times. The accuracy of the traffic leaving the room was 80.75%.

5. Conclusion

This research presents edge computing using raspberry pi as a processor to count people in a smart office system where people walk in and out all the time by relying on a normal quality webcam. This means that the camera is not high resolution. The picture is not sharp. Therefore, the image must be counted to adjust the color first before being processed in other steps. The designed system uses OpenCV and Raspberry Pi to help, and is Edge Computing processing, comparing images in front view and top view. The results obtained from the experiment show that the top view image. The view is more accurate and accurate, at 87.82% for entering the room and 80.75% out of 130 tests for walking out of the room. Flickering from the camera trying to adjust the light. When a person walks by, causing the motion detection system by focusing on dark objects, not working normally. In addition, the Raspberry Pi is quite slow when working with high resolution image processing. This system can also be used for counting people in other situations, such as counting people in factories, counting workers on the job site, etc.

Acknowledgment

This research was funded by King Mongkut's University of Technology North Bangkok. Contract no KMUTNB-64-DRIVE-34. We would like to deliver our greatest appreciation for their support.

References

[1] T. Parthornratt, N. Burapanonte and W. Gunjarueg, "People identification and counting system using raspberry Pi (AU-PiCC: Raspberry Pi customer counter)," 2016 International Conference on Electronics, Information, and Communications (ICEIC), 2016, pp. 1-5, doi: 10.1109/ELINFOCOM.2016.7563020.

[2] K. Rantelobo, M. A. Indraswara, N. P. Sastra, D. M. Wiharta, H. F. J. Lami and H. Z. Kotta, "Monitoring Systems for Counting People using Raspberry Pi 3," 2018 International Conference on Smart Green Technology in Electrical and Information Systems (ICSGTEIS), 2018, pp. 57-60, doi: 10.1109/ICSGTEIS.2018.8709141.

[3] C. Rajasekaran, P. Akshaya, S. Gokilavani, V. Gokulraja and B. P. Hari, "Smart digital hundi for identification of Indian coin and currency using Image processing," 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2018, pp. 1181-1184, doi: 10.1109/ICECA.2018.8474692.

[4] J. Wittayaprapakorn and T. Yooyativong, "Vehicle Seat Vacancy Identification using Image Processing Technique," 2017 International Conference on Digital Arts, Media and Technology (ICDAMT), 2017, pp. 152-155, doi: 10.1109/ICDAMT.2017.7904953.

[5] S. S. A. Abbas, M. Anitha and X. V. Jaini, "Realization of multiple human head detection and direction movement using Raspberry Pi," 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), 2017, pp. 1160-1164, doi: 10.1109/WiSPNET.2017.8299946.

[6] Q. Dong, Y. Yang, W. Hongjun and X. Jian-Hua, "Fall alarm and inactivity detection system design and implementation on Raspberry Pi," 2015 17th International Conference on Advanced Communication Technology (ICACT), 2015, pp. 382-386, doi: 10.1109/ICACT.2015.7224824.

[7] S. Mittal and V. S. Rajput, "Gender and Age based Census System for Metropolitan Cities," 2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), 2020, pp. 1094-1097, doi: 10.1109/ICRITO48877.2020.9198030.

[8] P. Cunha and D. C. Moura, "A scalable and privacy preserving approach for counting pedestrians in urban environment," 2015 12th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), 2015, pp. 1-6, doi: 10.1109/AVSS.2015.7301806.

[9] K. Yawale and S. Thorat, "A Smart System for the People Enduring through Alzheimer's Disease," 2019 International Conference on Intelligent Computing and Remote Sensing (ICICRS), 2019, pp. 1-5, doi: 10.1109/ICICRS46726.2019.9555852.

[10] F. T. Espinoza, B. G. Gabriel and M. J. Barros, "Computer vision classifier and platform for automatic counting: More than cars," 2017 IEEE Second Ecuador Technical Chapters Meeting (ETCM), 2017, pp. 1-6, doi: 10.1109/ETCM.2017.8247454.

[11] E. P. Myint and M. M. Sein, "People Detecting and Counting System," 2021 IEEE 3rd Global Conference on Life Sciences and Technologies (LifeTech), 2021, pp. 289-290, doi: 10.1109/LifeTech52111.2021.9391951.

[12] H. Yuliandoko, M. D. Ayatullah and F. E. Purwita, "Automatic Vehicle Counting Using Raspberry pi and Background Subtractions Method in The Sidoarjo Toll Road," 2019 2nd International Conference of Computer and Informatics Engineering (IC2IE), 2019, pp. 83-86, doi: 10.1109/IC2IE47452.2019.8940899.

[13] M. C. Le, M. -H. Le and M. -T. Duong, "Vision-based People Counting for Attendance Monitoring System," 2020 5th International Conference on Green Technology and Sustainable Development (GTSD), 2020, pp. 349-352, doi: 10.1109/GTSD50082.2020.9303117.

[14] J. W. Choi, X. Quan and S. H. Cho, "Bi-Directional Passing People Counting System Based on IR-UWB Radar Sensors," in IEEE Internet of Things Journal, vol. 5, no. 2, pp. 512-522, April 2018, doi: 10.1109/JIOT.2017.2714181.

[15] S. Matuska, R. Hudec and P. Kamencay, "IoT Based System for Detecting the Number of People and their Distribution in Classroom," 2022 ELEKTRO (ELEKTRO), 2022, pp. 1-4, doi: 10.1109/ELEKTRO53996.2022.9803353.

[16] J. Grönman, P. Sillberg, P. Rantanen and M. Saari, "People Counting in a Public Event—Use Case: Free-to-Ride Bus," 2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2019, pp. 1055-1059, doi: 10.23919/MIPRO.2019.8756921.

[17] I. Udrea, C. G. Alionte, G. Ionaşcu and T. C. Apostolescu, "New research on People Counting and Human Detection," 2021 13th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), 2021, pp. 1-6, doi: 10.1109/ECAI52376.2021.9515115.

[18] J. K. D. Lagman, A. B. Evangelista and C. C. Paglinawan, "Unmanned Aerial Vehicle with Human Detection and People Counter Using YOLO v5 and Thermal Camera for Search Operations," 2022 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS), 2022, pp. 113-118, doi: 10.1109/I2CACIS54679.2022.9815490.

[19] D. R. Lakshmi, K. Sangeetha, S. R. Krishnan, Saranraj and S. Ganesh, "Entry Screening Device for Public Transport During Pandemic Period," 2022 International Conference on Communication, Computing and Internet of Things (IC3IoT), 2022, pp. 1-4, doi: 10.1109/IC3IOT53935.2022.9768001.

[20] Y. Z. Quak, Y. X. Loke, Z. Y. Chan, S. Q. Chew and P. C. Ooi, "IoT-Based Tracing and Communication Platform for Disease Control," 2021 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET), 2021, pp. 1-6, doi: 10.1109/IICAIET51634.2021.9574005.