Queued Commuter Counting System by using Caffe Deep Learning Technique

Chaiyanas Mopung*, Hanafee Hama, Pimlaphat Moonngam and Sirikan Chucherd

Center of Excellence in AI and Emerging Technology, School of Information Technology, Mae Fah Luang University, Chiang Rai, 57100, Thailand
*Corresponding author. E-mail: 6031502006@lamduan.mfu.ac.th

ABSTRACT

The problem of using public transport is that the volume of transport vehicles is insufficient for the number of commuters. Commuters who have to wait in line lose time and opportunity. Solving this problem by increasing vehicle estimates might not be easy due to budget constraints. Transportation management is essential in order to meet the needs of commuters as much as possible. In this article, we proposed a system for displaying the number of commuters queuing at tram stations using a visual identification technique. The system consists of Raspberry Pi 3, Web camera, Firebase application, Caffe model, personal computer and HTML. A key component of the system is the person identification by the Caffe deep learning technique. Commuters pictures at each station are captured and processed. Finally, the quantity of the commuters are displayed on a website. We tested the system in a case study of the electric cars transportation on campus. The experimental results showed that the system worked well with the nearest distance of the passenger to the camera. The lower number of passengers, the better accuracy of person detection (different testing period). The result of detection people without mask on showed 100% accuracy for all distances.

Keywords: *deep learning, Raspberry pi, python, Caffe model, Firebase*

INTRODUCTION

In Mae Fah Luang university, the electric cars, GEM cars, are provided for students and staff transportation. There are 16 GEM cars and 20 tram stops around campus. Gem cars have been served for all university community. Students can use it for going to the classes, to the sport complex, to the shopping area and to the hospital. It is free of charge to commute around campus with convenience, safety and reliability. Eventhough the university community's transportation by GEM cars has many advantages, but the number of GEM cars compare to the number of the commuters is

Article history:
Received 13 February 2021; Received in revised from 09 June 2021;
Accepted 10 June 2021; Available online 21 June 2021.

still the open problem. The Gem cars have been managed manually with a fixed schedule by the department of building of the university. During rush hours of every weekday, the GEM cars will be launched in every 5 minutes. Due to one GEM car has only 15 seats to be served but the number of commuters is sometimes in tens to hundreds digit. Especially when the university has some big events such as the fresher night, the graduation day and other events. Therefore, it could not be served to all passengers. Resulting in longer waiting time and crowded numbers of commuters within any particular stops.

With the above-mentioned reason, the way to fix the problem by increasing the number of the GEM cars is not easy because it will involve with the university plan and budget. Therefore, authors have the idea that if there is some website that can provide the support information for commuter's decision. It will be the better for people to make decisions of waiting at the tram stops according to the demand of the commuter as well as the vehicle operate. In this paper, we propose a queued commuter counting system by using Caffe deep learning technique. After that, the number of people waiting at the tram stop will be displayed via the web page. The system consists of a Raspberry Pi and a web camera to collect the pictures of human faces. Next the machine learning named the Caffe model is applied for the face detection part. The Caffe model is a great model in face detection; they are able to capture human faces while wearing a mask as well as able to detect with varying positions. For instance, human faces turning left, right, up and down position. Speaking of displaying commuter numbers. The number of commuters at every stop are displayed on the web application. Hence, people can make their decision whether they will keep waiting for GEM cars or take other transportations. Besides, staff can use this information to manage GEM car service to support all commuter waiting at each stop.

LITERATURE REVIEW

Shanmugavadivu and Ashish (2016) proposed the face of a human can be detected in real-time by using the Partially Occluded Face Detection (POFD). Their program worked with the facial features that are normalization and labeling eyes, nose, and mouth then combine with skin color detection using many complexing methods to contribute in a classification of the skin pixel. In the end, they were tested with the POFD in various conditions. Unfortunately, a few facials were difficult to label and unable to detect in real-time. It needed to be drawing first. Besides, their system had been trained only in a straight direction and the experiment did not show the result accuracy for any possible work of detection.

Ahmad *et al.* (2018) proposed the face detection on the matching method with the face movement. The reference image or an input image had been taken from the camera. For the detection, they tried to solve the three dimensions, lighting condition and direction to identify the face area. The template matching method was used for finding the closest area that matches with the faces. However, the system concerned about the skin features in intensity that all humans have a specific skin color. Therefore, the segmentation was applied to erase the unwanted area. Then, the Gaussian will obtain the pixel of images and change the RGB to the YCbCr color space. Finally, the histogram will be plotted to determine the skin region. After all process, the framework will get the model that can matched the face with rotation position and lighting control. Nevertheless, there have no experiment to insist that the model has worked well in intensity test or facial position.

Satyavolu and Bagubali, (2019) study frameworks in the eye of applications which are the Tensorflow and the Caffe algorithms. Their methods captured images from the camera. Later the input images were trained and tested with the appropriate algorithms particular with the fast region convolutional neural networks (fast R-CNN). The Python programming languages were supported in TensorFlow. Technically, the object detection and recognition involving objects such as persons with the Caffe algorithms have satisfying results with 96% accuracy of prediction. However, the TensorFlow framework achieved 85% that showed the lower accuracy compared with the caffe model in the object detection area. TensorFlow are good models with the voice recognition and Time series where they achieved higher accuracy compared to other models.

Dev *et al.* (2020) implemented the technology of detection to prevent the hazardous risk in fire extinguish, monitoring the garbage and air quality index. All data was gathered from the sensors to notify the fire accident air quality index and garbage overflow. Machine learning was applied to measure the air quality index (AQI). Both fire detection and garbage monitoring used the sensor in a specific function of their way. Supervise machine (SVM) algorithm also being used for classification and segmentation on the AQI. All of the work results were showed in their web application and the accuracy was sufficient enough to detect their interest. The contribution in the publication of their work is an inspiration for our work to accommodate people in our campus.

There is a splitting deep learning sequence of the Caffe framework by introducing a new layer and performing distributed processing between client side with Raspberry Pi and GPGPU NVIDIA GeForce GTX 980 as a Cloud side. The work was made by Ayae *et al.* (2016). They splitted into two portions size as Client side and the Cloud side that will be joined by "Sink" and "Source" layers respectively. Both corresponding layers were connected via TCP/IP particularly Acknowledgement (ACK) for forwarding and responding the data in pipeline ways. Beneficially, the approach methods protected the privacy by not forwarding raw data rather its sending feature value. Speaking of the processing time, their proposed work reduced the processing time as it was sent to others corresponding parts while maintaining the great accuracy even if the amount of time reduces. Practically, their method inspires the author by using Raspberry Pi 3 as a processing unit.

Raspberry Pi is a microprocessor often used in the face detection alongside Python programming languages, OpenCV and Numpy for processing image recognition. The database (SQLite) was used to store data of people's faces. This work was performed by Nafis *et al.* (2019). There were three segments of system architecture. First, the data segment was dedicated to store the personal identification such as ID, name, gender, age and departments. Second, the trainer segments mainly focused the train system to identify person faces in gray scaling format. Last, the system detection segment, it was a two-dimensional matrix coded by python languages. Later, Numpy makes it easy to process in multidimensional matrices. Where the manipulated matrices are compared with samples for identification matched. In any case, if there are changes in person face for instance make-up which is of lesser accuracy outcome. Otherwise, conditions of changing background while performing face recognition are great with matching scores.

METHODOLOGY

This paper presents a commuter counting system by using image processing that runs on the Raspberry Pi 3 along with the web camera. In image preparation, the image in each frame of video at the GEM stop will be taken as an input image. Each input image will be processed by the Caffe model. The face of the commuter will be detected by the model prediction. Next, the results will be submitted via sim card on the internet for sending the data to store in a firebase and display on the web application. The overall system will take 5 min for the appropriate time of update and prevent transportation errors. The proposed system is shown in Fig. 1.

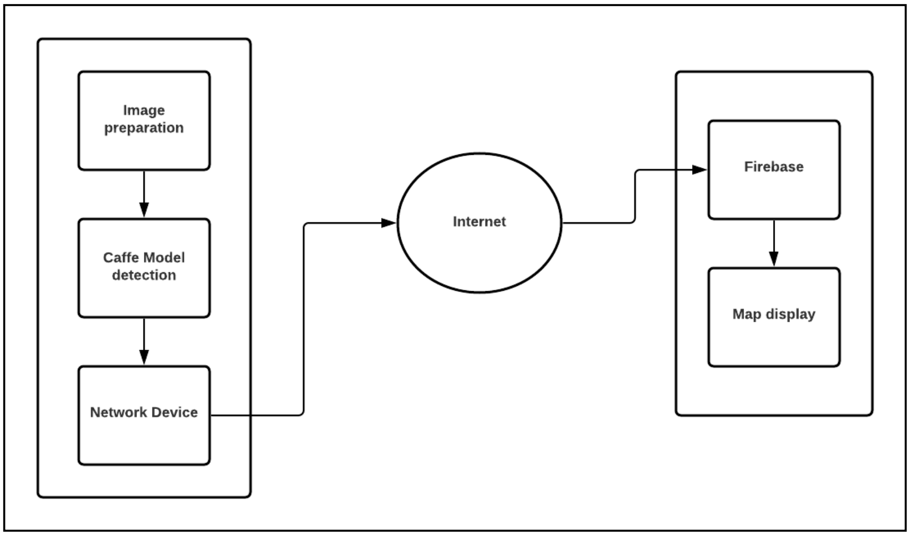


Figure 1. Proposed System.

Image Processing and Output.

The image processing is mainly using the Python programming languages with deep learning technique along with the Caffe Model (Yangqing, 2017). Caffe Model is a framework of deep learning that supports image classification. The model is an open-source library that gives open access to deep architectures (Emine *et al.* 2017). The system flow for image processing diagram is presented in Fig. 2. Deep learning is a neural network (Konstantinos *et al.* 2015). It has many layers which are the input layer, hidden layer, and output layer to categorize the features of a raw input image, letters, and faces. The hidden layer acts as a neural that processes and sends data to another layer. The neural network is inspired by the brain of humans and learning a lot of data as an experience to accomplish the task. In the method proposed, the system will prepare the images by input images from the frame of the video camera then using the deep learning technique to determine the test and train set of data. The input images of each frame in the video will be a test set of data. The Caffe model is a trained set of image data. The model implements the feature engineer algorithm for handy in labeling automatically of the region of interest (ROI). For the Caffe model, the region of interest (ROI) is the faces of humans that mask on and mask off. Furthermore, the model has learned the faces in various degrees for ease in detecting the ROI when there are moving images or video. In the end, the input image will be detected when some objects are similar to the ROI of the Caffe Model. In the part of sending data, the Python code will count and send an integer of the detected faces to the database after using the Caffe model algorithm with input images for detection.

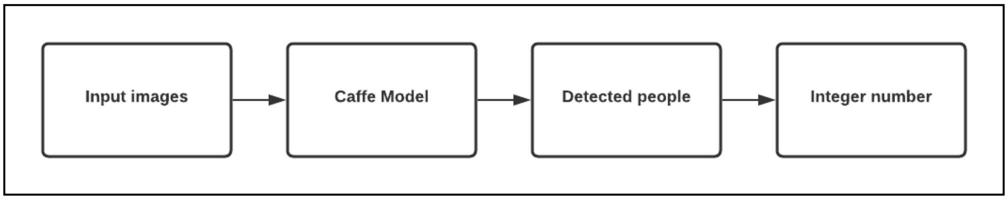


Figure 2. Image Processing.

Data storage and display.

To store information, the system will use the Firebase as a database to store all information after getting the integer number. Firebase is a free online database platform with various functionalities that can contain all information for free 30 GB per month. In addition, we use a Real-time database function to promptly inform the information at Gem stop. After that, the HTML programming language will retrieve data from the Firebase to display on the map as shown in Fig. 3.

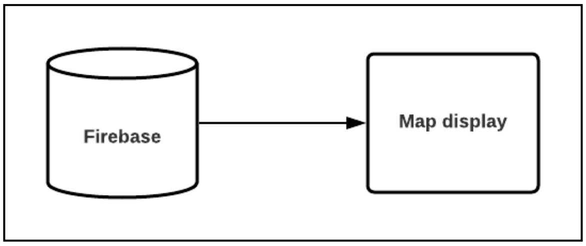
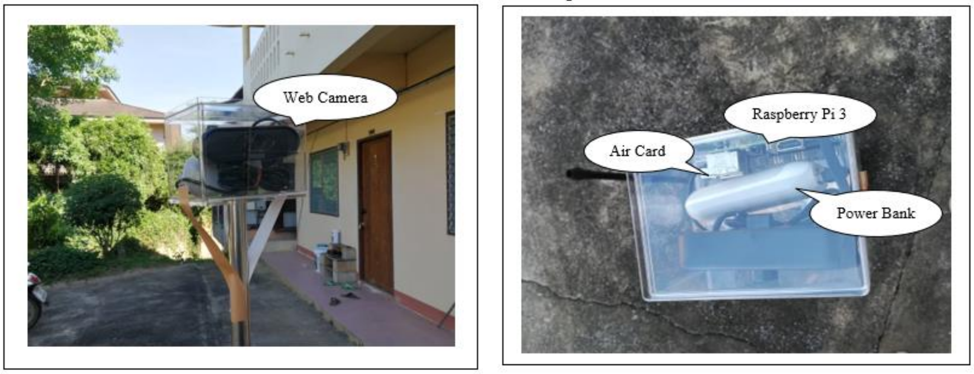


Figure 3. Data storage and display.

RESULTS AND DISCUSSIONS

System Implementation.

The observation and experiment are placed at the GEM stop (D1 building) in Mae Fah Luang University. This GEM stop usually has many commuters all day. It is the good location to put our testing devices considering all parameters. The web camera is tied up with a Raspberry pi 3, and WIFI module. They are put into a box and attached to a 1.5-meter pole. The side and top view of devices are shown in Fig. 4 (a) and (b) respectively. The camera is located in front of the 4m.* 2m. GEM stop with 3.5-meter distance from the center. It is an appropriate position to detect all commuters faces. Fig. 5 presents the example of the picture taken by the device.



(a) (b)

Figure 4. Invention (a) Side view, (b) Top view.

Figure 5. Example of picture taken from camera.

Evaluation of image processing.

The experiment initially evaluated the performance of human face detection by Caffe model. The faces of volunteers are examined and evaluated in various parameters such as facial posture, appearance, distance from camera and testing period of time. For the facial posture experiment, we have tested our system by taking the picture in the straight direction, the side direction and the above direction. For the appearance test, the volunteers are asked to be tested with and without wearing different colors masks.

(a) Straight direction without mask on

(b) Side direction without mask on

(c) Above direction without mask on

(d) Straight direction with light color mask on

(e) Side direction with light color mask on

(f) Above direction with light color mask on

(g) Straight direction with diverse color mask on

(h) Side direction with diverse color mask on

(i) Above direction with diverse color mask on

Figure 6. Face detection results.

Fig. 6 shows all detection results at the 2-meters distance from camera. If our proposed method detected the human face, it will show the rectangle around faces. The experiment has been divided to presents the face detection results without mask on as shown in Fig. 6 (a) straight direction, (b) side direction, and (c) above direction respectively. Meanwhile, Fig. 6 (d) to Fig. 6 (f) are the results with the light color mask on. Finally, Fig. 6 (g) to Fig. 6 (i) present the detection results with the different color mask on face in straight direction, side direction, and above direction respectively. To sum up, straight direction with camera, Caffe model are able in detecting faces perfectly with no mask as well as with many types of masks on. Side direction, without masks the caffe model are good in detecting. While, colors masks show non detectable faces including in above direction.

For the distance experiment, we have tested our system by taking the picture in straight direction with variety distance from the camera (2-4 meters). The experiment uses 3 volunteers to accommodate in a limited range of camera, appearance, and distance. Fig. 7 presents the average accuracy of the experiments of appearance (face with and without mask on) with difference distance from the camera (2-4 meters in straight direction).

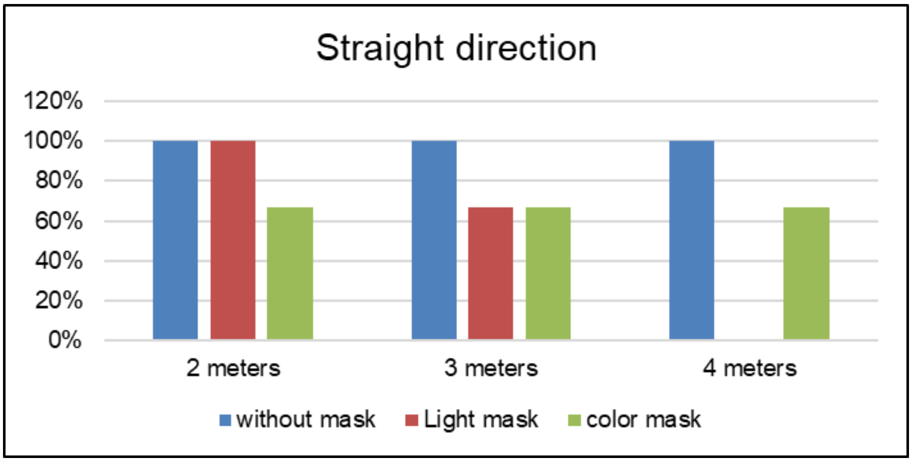


Figure 7. Average accuracy of the distance experiment.

Equation (1) is the formula for the percentage of the detection accuracy calculation. From Fig. 7, the performance of system is almost 100% of the face detection experiment of people without mask on in every distance. Meanwhile, the average accuracy of distance experiment (2-4 m.) when the face is color mask on in a straight direction are 74.04%, 51.84%, and 18.51% respectively. In other words, the accuracy will decrease when the distance increase and diverse direction.

$$Detection\ accuracy\ \% = \frac{Detected\ people}{Total\ people} * 100 \qquad (1)$$

According to the evaluation results, the confusion matrix is shown in Fig. 8 are the results taken from a straightforward distance with a camera of 2, 3, and 4 meters that occur in 54 possible ways of 3 volunteers. We account for detectable and undetectable distances of sample experiments. True positive (TP) achieved 40 where faces detectable correctly by Caffe model. False positives (FP) where the Caffe model detects other objects than sample faces taken into zero. False negatives (FN) where samples are along with the distance of 2 to 4 meter where Caffe model is not capable of detecting faces come as 14. Last column, True negative (TN), where models strongly able to detect human faces, zero were obtained.

	Actual Values	
Predicted Values	TP = 40	FP = 0
	FN = 14	TN = 0

Figure 8. Confusion matrix.

The 54 possible from the author test in 2 to 4 distance were taken as a dataset for the straightforward in the evaluation. For the precision, show how actually that we can detect correctly from the formula in Equation (2). The precision in evaluation results predicted 100% for FP equal to zero. The recall showed how high possible could be in Equation (3). There were undetected faces in the FN by 14 therefore the recall obtained 74%.

$$Precision\ \% = \frac{Detected\ people}{Detected\ people + Detect\ wrong\ object} * 100 \qquad (2)$$

$$Recall\ \% = \frac{Detected\ people}{Detected\ people + Undetected\ people} * 100 \qquad (3)$$

In addition, the experiments in different period of time had been evaluated the proposed system. The duration of the rush hour in the morning, noon, and evening period are 8:00-9:00 am, 12:00-1:00 pm, and 6:00-7:00 pm respectively. In each duration, the experiment is designed to evaluate the proposed system in every 10 minutes (six attempts in one hour). It is the average time of waiting for GEM car for one person. The experimental results of our work are the average accuracy for 89.2%, 76.5%, 82% in the morning, noon, and evening respectively. We also set up the hypothesize of our experiment that the number of people might affect the performance of face detection. Therefore, we have done another experiment in the regular duration (period between of rush hour such as 10:00 – 11:00 am. and 2:00-3:00 pm.). The result shown the 91.7% average accuracy which proved that the number of people slightly affects the performance of face detection.

The images in Fig. 9 show the results of performance in detection between rush and usual hour. The experiment took place around fully fourteens days duration with three-time duration morning, evening and noon respectively. The camera took 11 hours of the commuter's images each day. Fig. 9 (a) show that the waiting people in the usual time are mostly detected. Meanwhile, some people has not be detected in rush hour as shown in Fig. 9 (b). It may cause by the improper of facial posture or the density of commuters at that period of time.

(a) Usual Time (b) Rush Time

Figure 9. Example results in difference time.

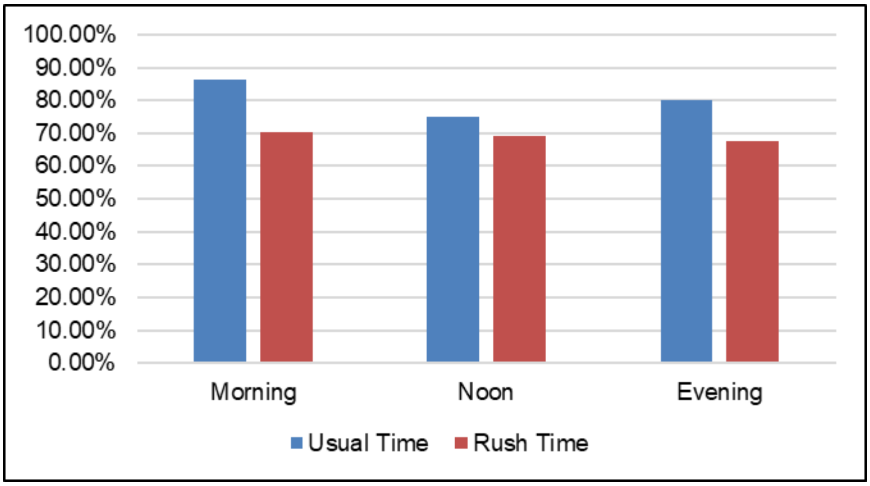


Figure 10. Comparison of usual time and rush time.

The charts in Fig. 10 gives the information about detectable face during rush time and usual time of Gem cars user in campus. In usual time and rush time, the experiment was 6 time attempted during 3-time duration morning, noon and evening respectively. Usual morning time the Caffe model achieved highest detectable of passenger with 86.36 percent compare to rush time by 70.47 percent. The usual noon time were comparable with rush time by 6 percent different, usual time takes 75 and rush time takes 68.96 percent of detectable faces respectively. Usual evening time were 80 percent detectable faces, where rush time could only 67.74 percent.

Web Application.

After human face have been detected by Caffe model, the data will be sent and stored in Firebase. These number of detected commuters have been retrieved from Firebase and display on the web application. Fig. 11 presents a Firebase real-time database display. The display shows the data received from the image processing system. Finally, when a GEM car icon on the map in Fig. 12 is clicked, a number of waiting commuters at each station will be shown along with that station name. This number will be updated every 5 minutes.

Figure 11. Firebase display.

Figure 12. Map display.

CONCLUSIONS

This paper proposed the method to detect the faces of commuters at the bus stop. The framework was developed by considering many parameters such as facial posture, appearance, distance from camera and testing period of time. The experimental results showed that our algorithm obtained good performance when the face picture was taken in the straight direction to the camera. Also, the accuracy is still high even if the person wears a colour mask with the long distance from the camera. The nearer distance, the better detection performance. The accuracy of the system in usual time is higher than the rush time because of the possibility of improper facial posture or the density of commuters. Overall, the system works well and satisfies all testing conditions. In addition, the satisfaction of the 100 commuters generally agree with our purpose methods and web page application where they answered questionnaires online. During rush hours, they had used ours web pages to make decisions for commuting on the campus. They were satisfied with 80 percent of usefulness with web pages. 74.4 percent of commuters had agreed that web pages had impacted their daily life in the university by making their lives easier. In future work, the authors plan to collect and analyze all data to establish the appropriate schedule of GEM cars services for the passenger's decision.

ACKNOWLEDGEMENT

This research was supported by Asst.Prof.Dr.Sirikan Chucherd, Assoc.Wg.Ldr.Dr. Tossapon Boongoen and Dr. Surapol Vorapatratorn. We also acknowledge the contribution of Mr. Lukman Chaiyarab for his kindness in this paper. Mr.Gaurav Dhungyel and Mr. Kelden Wangchuck Phuntshok for his recommendation of code in this HTML. At last, we would like to thank the Center of Excellence in AI and Emerging Technology, School of Information Technology of Mae Fah Luang University for allowing us to conduct an experiment as well as who helped, supported, and encouraged us to successfully finish this research.

REFERENCES

Yangqing, J. (2017). *A deep learning name Caffe Model*. Retrieved from https://caffe.berkeleyvision.org/

P, S., & Ashish, K. (2016). *Rapid face detection and annotation with loosely face geometry*. Retrieved from https://ieeexplore.ieee.org/document/7918032

Dev, V. S., Amogh, N. P., Kaustubh, Y. K., Jay, P. G. & Meera, N. (2020). *IoT and ML based Smart System for Efficient Garbage Monitoring: Real Time AQI monitoring and Fire Detection for dump yards and Garbage Management System*. Retrieved from https://ieeexplore.ieee.org/document/9214202

Ahmad, Z., Siti, N., Deris, S., Firdaus., Huda, U., Yogie, Sanjaya.. & Yesi, N. K. (2018) . *Face Movement Detection Using Template Matching*. Retrieved from https://ieeexplore.ieee.org/document/8605215

Ayae, I., Masato, O., Atsuko, T. & Hidemoto, N. (2016). *Evaluation of distributed processing of caffe framework using poor performance device - IEEE ConferencePublication*.Retrieve from https://ieeexplore.ieee.org/document/ 7841082

Nafis, M., Noushad, H., Mohammad, M. R., Nadimul, I., Zayed, H. S. & Asaduz, Z. M. (2020). *Face Recognition System Based on RaspberryPi Platform*. Retrieved from https: / / ieeexplore. ieee. org/ document/ 8934485/ authors#authors

Konstantinos, M., Konstantinos, K., Anastasios, D. & Nikolaos, D. (2015). *Deep supervised learning for hyperspectral data classification through convolutional neural networks*. Retrieved from https://ieeexplore.ieee.org/ document/7326945

Suguna, S., & A, B. (2019). *Implementation of Tensor Flow and Caffe Frameworks :in View of Application*. Retrieved from https://ieeexplore.ieee.org/document /8899466/authors#authors

Emine, C., Ahmet, C. & Erdal, *Ö*. (2017). *I mage classification with caffe deep learning framework*. Retrieved from https://www.researchgate.net/ publication/ 320829871_Image_classification_with_caffe_deep_learning_fra mework