

Face Recognition System Design and Implementation using Neural Networks

Jamil Abedalrahim Jamil Alsyayadeh^{1*}, Chang Kai Xin⁴

Department of Electronics & Computer Engineering Technology, Fakulti Teknologi Kejuruteraan Elektrik & Elektronik (FTKKE), Universiti Teknikal Malaysia Melaka (UTeM), Melaka, Malaysia

Irianto²

Department of General Education, Faculty of Resilience
Rabdan Academy, Abu Dhabi, United Arab Emirates

Azwan Aziz³

Department of Industrial Technology, Faculty of
Mechanical and Manufacturing Engineering Technology
Universiti Teknikal Malaysia Melaka (UTeM)
Melaka, Malaysia

A. K. M. Zakir Hossain⁵

Centre for Telecommunication Research & Innovation
(CeTRI), Fakulti Teknologi Kejuruteraan Elektrik &
Elektronik (FTKKE), Universiti Teknikal Malaysia Melaka
(UTeM), Melaka

Safarudin Gazali Herawan⁶

Industrial Engineering Department, Faculty of Engineering
Bina Nusantara University, Jakarta, Indonesia 11480

Abstract—Face recognition technology is used in biometric security systems to identify a person digitally before granting the access to the system or the data in it. There are many kidnappings or abduction cases happen around us, however, the kidnap suspects will be set free if there is lack of evidence or when the victims are not able to testify in court because they suffer from post-traumatic stress disorder (PTSD). The objectives of this study are, to develop a device that will capture the image of a kidnapper as evidence for future reference and send the captured image to the family of the victim through email, to design a face recognition system to be used in searching kidnap suspects and to determine the best training parameters for the convolution neural network (CNN) layers used by the proposed face recognition system. The accuracy of the proposed system is tested with three different datasets, namely the AT&T database, face database from [23] and a custom face dataset. The results are 87.50%, 92.19% and 95.93% respectively. The overall face recognition accuracy of the proposed system is 98.48%. The best training parameters for the proposed CNN model are kernel size of 5x5, 32 and 64 filters for first and second convolutional layers and learning rate of 0.001.

Keywords—Face recognition system; biometric identification; face detection; image processing; convolutional neural networks

I. INTRODUCTION

There are many kidnappings or abduction cases happen around us. The fate of the victim remains unknown until the suspect is found and arrested. In the cases of victims being rescued in time by the authorities, it is likely that the victims who suffer from post-traumatic stress disorder (PTSD) could not recognise or remember the face of the kidnapper during their testimony in court. Therefore, the purpose of this work is to develop a device to capture the image of a kidnapper and reduce the time taken for the authorities to find the suspect.

One of the most widely used technologies in the world today is the facial recognition technology. With the use of

facial recognition, a biometric system can identify a person digitally before granting the access to it or the data in it. Complicated and unrealistic biometric security systems are often portrayed by computer graphics in many futuristic movies; however, Apple had taken one step forward and release a face unlock feature for its iPhone X in 2017. This breakthrough uses a sensor to scan the face of the user and saves it as the face ID. The phone can be unlocked when the face of the person unlocking the phone matched with the face ID. The release of this new authentication method has made a big impact in the smartphone industry and all the latest smartphones have started to implement the same face unlock feature in their systems. Facial recognition technology is also used in other systems, for instance, it is used in airport security, law enforcement, attendance systems and to search for a person. This system uses neural network in the development and implementation of face recognition system. Neural network can be trained to process and analyse data, recognise patterns, and make prediction about specific operations. Generally, the programmer needs to provide numerous examples to train neural network, in order for it to learn the patterns [1].

We proposed an idea which combines both hardware and software, where the hardware will capture image of the suspect and the software will perform facial recognition. Our approach allows the hardware to not only capture the image of the suspect but also send the captured image to notify the victim's family that the victim is in danger so that the family can report to the authorities and rescue the victim in a shorter period. The image is then be used as an input to the face recognition system to identify the suspect. The accuracy of the proposed system is 87.50% when tested with the AT&T database, 92.19% with face database from [23] and 95.93% with custom face database. The potential users of the proposed system could be women and children as they are often the target of abduction.

*Corresponding Author.

The device can be attached to the users' belongings and the users can press the button to activate the device when they are in danger and their family will be notified in no time. In addition to all this, the best training parameters for the convolutional neural network layers used by the proposed face recognition system will be determined.

The remain of this paper has been organized as follows: Section 2 discusses the related works. The background of the study is described in Section 3. Section 4 described the results and discussion and finally, the conclusion is described in Section 5.

II. RELATED WORK

In [2], the factors that might affect the accuracy of face recognition systems are classified into two main categories, intrinsic and extrinsic factors. Physical conditions of the human faces are considered as intrinsic factors, for example, aging and facial expressions. Extrinsic factors are made up of partial occlusion, pose variance and illumination. The authors in [3] stated that a useful face recognition system must fulfill the following characteristics: firstly, it must be able to work well with both images and videos, secondly, it must be capable to process in real time, thirdly, it must be robust in illumination variation, fourthly it needs to perform its task without being affected by hair, ethnicity or gender of a person and lastly, it must be able to work with faces detected from all angles. They also stated that a robust face recognition system is made up of three basic steps: face detection, feature extraction and face recognition. There are various techniques that can be used for face recognition, such as Eigenface, Neural Network, Hidden Markov Model (HMM) and Support Vector Machine (SVM).

The Eigenface technique is used in [4] for building face recognition software and the average accuracy for their face recognition software is 85%. Next, a hybrid approach which consists of the Haar Cascades and Eigenface methods that can detect multiple faces in a single detection process is proposed in [5]. The accuracy of this proposed solution was reported to be 91.67%.

Another technique that can be used to develop a face recognition system is called the neural networks. This technique was used in [6] and the average accuracy of the proposed system was 96.84%. The authors in [7] proposed to improve the backpropagation artificial neural network (BP-ANN) for a better performance of the face recognition system. The proposed system yielded a success ratio of 82%. In [8], the researchers used a hybrid approach which includes Elman Neural Network, Curvelet transform and HSI colour space. The resulting accuracy obtained by the authors was 94%. In addition, the authors of [9] proposed an effective method for face recognition that uses Principal Component Analysis (PCA) and models trained with Feed Forward Back Propagation Learning (FFBPL) and Elman Neural Network. The results of FFBPL were 98.33% and 98.80% while the results of Elman Neural Networks were 98.33% and 95.14%.

Next, convolutional neural networks are widely used in research these days, in [10], a real-time face recognition system is built using CNN. The maximum accuracies of the proposed system are 98.75% for standard datasets and 98.00% for real-

time inputs. An algorithm for face detection and recognition based on the same concept in [11][12] gives the accuracy of 97.9%. Besides, to develop a face recognition with small dataset, the authors proposed a method that uses a modified deep learning neural network in [13][14]. The accuracy of the proposed system achieved 99.6%. A deep convolutional neural network-based face recognition system that uses transfer learning approach is proposed in [15][16]. The accuracy of the proposed algorithm was 99.06%. In [17], an attendance system with face recognition based on deep learning technique. The overall accuracy obtained by the proposed system in a real-time environment was 95.02%. Similarly, an intelligent face recognizing attendance system that can identify several people simultaneously that is built based on CNN is proposed in [18]. The proposed system was tested with frontal view, side view and downwards view. The accuracies obtained for these three conditions were 81.25%, 75.00% and 43.75% respectively. A multi-face recognition system is proposed by the researchers in [19][20] to detect the prisoners in jail and the accuracy was 87%. Next, the researchers proposed a face recognition using a CNN model in [21][22]. The highest accuracy achieved by the proposed system was 98.3%. The authors in [23] proposed a deep CNN based face recognition system that can identify an individual in all possible conditions that might affect the accuracy of the face recognition system. The accuracies of the proposed system were 99.7% and 94.02% respectively. The authors of [24] proposed a home security system that uses face recognition technology developed by CNN. The Raspberry Pi was used as a microcontroller so that when the face of the homeowner was detected, the door will be unlocked automatically. The proposed system was able to achieve 97.5% accuracy.

A combination of 2D Hidden Markov Model (2DHMM) and Expectation-Maximalization (EM) algorithm is used in the face recognition system in [25] and its recognition rate was 99%. Besides that, in [26], two-dimensional Hidden Markov Models was used for face recognition. The recognition rates for 2D images were 93% and 95%. Next, the recognition rates for 3D images achieved 94% for both UMB-DB and FRGC databases. Lastly, the recognition rate for 2D+3D images was 96% for both databases.

By using wavelet Gabor filter and SVM, the authors in [27] have successfully built a 3D facial recognition system. The highest accuracy that the proposed system achieved was 97.3%. The authors in [28] proposed a face recognition that uses SVM with implementation of kernel as the classification method to identify lookalike faces. Two types of kernels were used in the proposed system, namely, the Radial Basis Function kernel and the polynomial kernel. The accuracy for both kernels is 94%.

III. BACKGROUND OF THE STUDY

This system two uses publicly available face databases, namely AT&T database and a database provided by the author of [29]. Besides, a hardware device is used to capture the image of the kidnapper, the image will be used as input to the face recognition system, therefore a custom face dataset needs to be created to test the accuracy of the proposed system. The components used for the hardware are the ESP32-CAM, FTDI

adapter and push button. The block diagram of the ESP32-CAM is shown in Fig. 1. The hardware is programmed using Arduino IDE while the face recognition system is built in MATLAB.

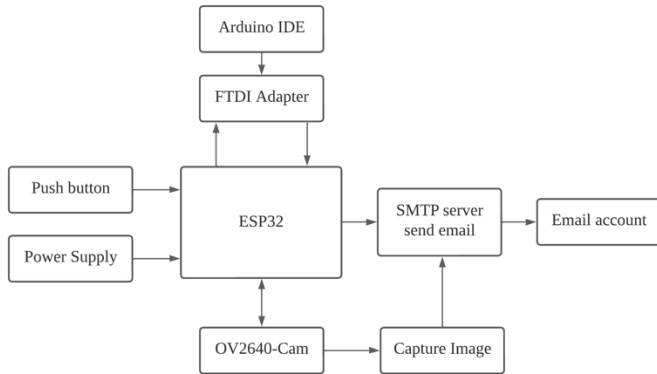


Fig. 1. ESP32-CAM Block Diagram.

The proposed system is developed according to the flowcharts shown in Fig. 2 to Fig. 4. Fig. 2 shows the flowchart of creating a custom face dataset. A custom face dataset is created because an ESP32-cam will be used to capture images of people, it is certain that during the testing process, the face captured by the device is not in the pre-curated face databases, such as the AT&T Database that contains a total of 400 images of 40 individuals. Hence, in order to yield higher accuracy, other than using the publicly available face dataset, we decided to create a custom face dataset for the system. After creating a face dataset, the face recognition system can be trained. The images in the face dataset created manually will be separated into two folders, one for testing purpose and another for training purpose. The same image cannot exist in both folders, in other words, all images of each individual in both folders must be different in terms of postures, lighting condition, facial expressions. The proposed system will be trained and tested with the images available before testing with the image captured directly from the ESP32-CAM.

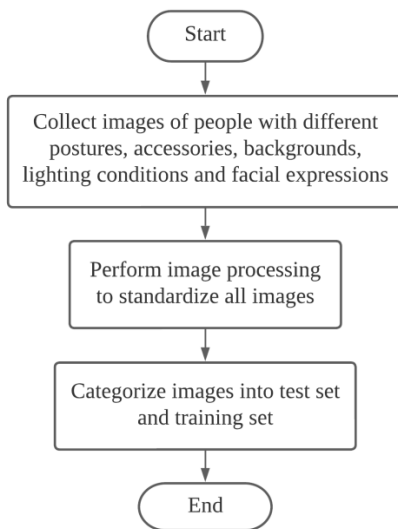


Fig. 2. Flowchart of Creating a Custom Face Dataset.

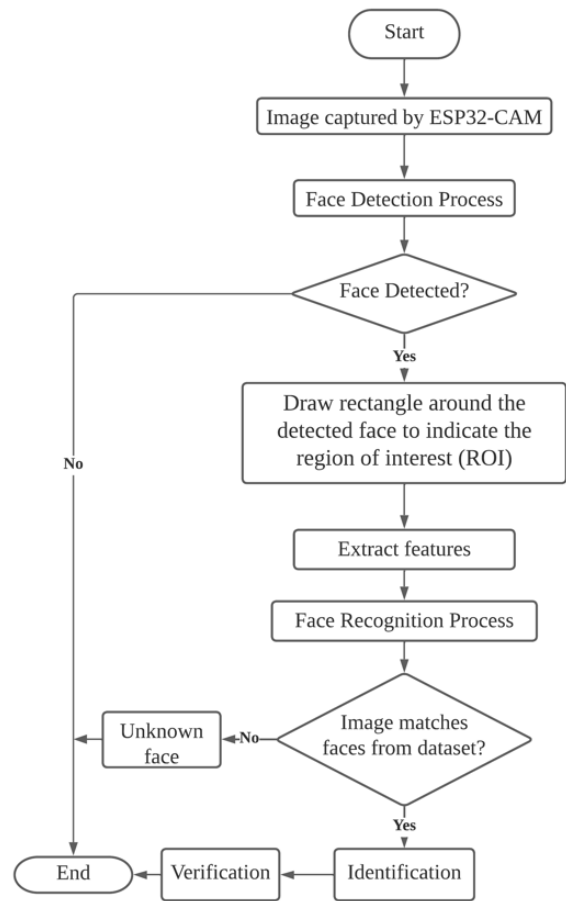


Fig. 3. Face Recognition System Programming Flowchart.

Fig. 3 shows the programming flowchart of the ESP32-CAM using Arduino IDE. The initialization process includes assigning the SSID and password to the device so that it can connect to the Wi-Fi or mobile hotspot and setting up the email account that will be receiving the captured image. Initially, the device will be in deep sleep mode and when it is triggered by a button, it will wake from the deep sleep mode to perform its function, which is capturing images. After an image is captured, the image will be sent to the assigned email account through SMTP server. After sending the email successfully, the device will go back to deep sleep mode again.

Fig. 4 shows the programming flowchart of the proposed face recognition system. Firstly, the image captured by the ESP32-CAM must be downloaded from the email account before it can go through the face detection process. In face detection process, if the program detected a face, it would draw a rectangle around the face to mark it as a region of interest (ROI) so that the program can omit other unwanted parts to focus on features extraction. After extracting the facial features, the program will perform the face recognition process which uses the Neural Network as foundation. When the captured image matches the faces from dataset, the system will identify and verify the identity of the person being captured in the image. If the image does not match any of the faces from the dataset, it is an unknown face to the system and the program will terminate.

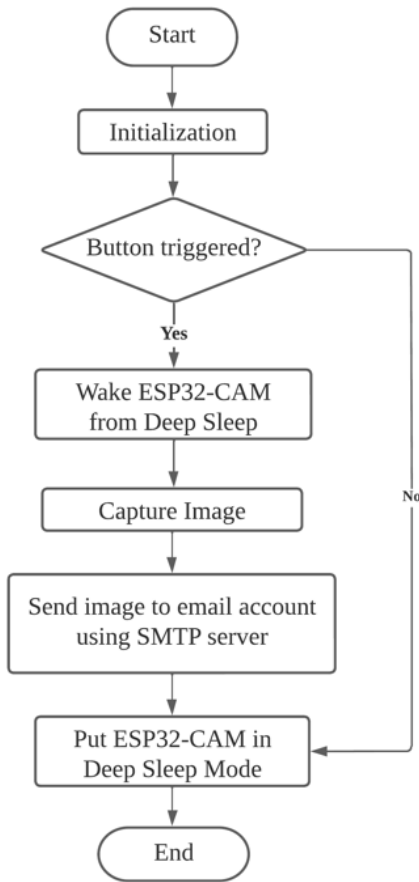


Fig. 4. ESP32-CAM Programming Flowchart.

Fig. 5 shows the circuit connection between ESP32-CAM, FTDI Adapter and an external push button. A jumper wire is connected from GPIO 0 to GND for programming purpose and can be removed once the programming is finished. The push button is used to wake the ESP32-CAM from deep sleep mode and it is connected to the GPIO 13 pin. The TX pin of ESP32-CAM is connected to the RX pin of the FTDI Programmer and the RX pin of ESP32-CAM is connected to the TX pin of the FTDI Programmer so that data can be exchanged between these two devices in serial communication.

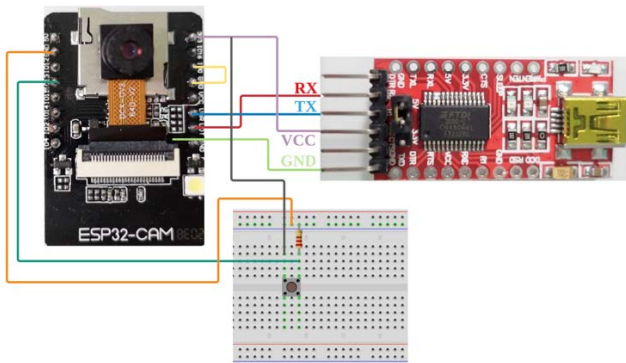


Fig. 5. Circuit Design.

IV. RESULTS AND DISCUSSION

A good face recognition system must fulfill several criteria; the most important is the accuracy of the result. Face recognition systems are widely used in law enforcement, a slightly inaccurate result might cause an innocent person being wanted and wrongly accused of crime that he or she did not commit. To test the functionality of a face recognition system, face datasets are used. There are many different face datasets available publicly such as LFW, Yale and AT&T databases. This system uses AT&T databases and a custom face dataset. This section explains the development of a face recognition system and the combination of the system with some hardware that will be discussed in the following section.

A. Hardware Implementation

The hardware device is configured so that when the external button is pressed, the ESP32-CAM will take a picture, connect to the internet and send the picture to an assigned email address. Fig. 6 shows the image taken by the ESP32-CAM being sent to the assigned email address.

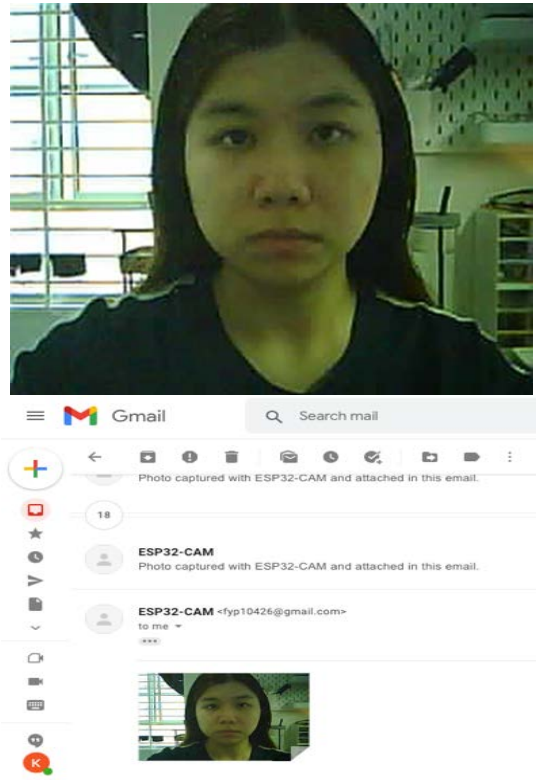


Fig. 6. Image Taken by the ESP32-CAM Sent to an Assigned Email Address.

B. Training Parameters for the Proposed System

The researcher carried out an experiment to find out the values of the parameters that will give the CNN model the best accuracy in face recognition. The parameters chosen are the kernel size, the number of filters for convolutional layers and the learning rate.

Fig. 7 shows the validation accuracy and average elapsed training time of the CNN model. For easier understanding, the four values of the x-axis from the graph will be referred as set

A, B, C and D in the following discussion. From the graph, it can be concluded that in general, the validation accuracy for learning rate of 0.03 is the lowest compared to other learning rates, except for set B because the learning rate that gives the lowest validation accuracy for set B is 0.001. Thus, learning rate of 0.03 is the first to be eliminated.

Next, the total time elapsed for training range between 4 minutes and 9 minutes. The CNN model that uses the shortest average training time is set A while the longest is set D. Set C has the second shortest average training time, and compared to set A, the validation accuracy of set C is higher in overall. The aim is to find the value that will give the highest accuracy without compromising the training speed, set C meets the requirements, and from all the learning rates in set C, 0.001 gives the highest accuracy, 90.48%. Therefore, the parameters that will be used for training the CNN model are, kernel size of 5x5, 32 filters on the first convolutional layer and 64 filters on the second convolutional layer and learning rate of 0.001.

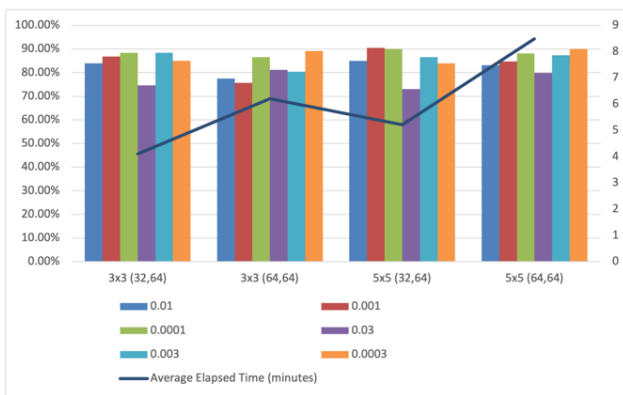


Fig. 7. Validation Accuracy and Average Training Time of the CNN Model with different Training Parameters.

TABLE I. LAYERS OF THE PROPOSED CNN MODEL

Layer	Type	Activations	Learnable
1	Input	112 x 92 x 1	-
2	Convolution	112 x 92 x 32	Weights 5 x 5 x 1 x 32 Bias 1 x 1 x 32
3	Batch Normalization	112 x 92 x 32	Offset 1 x 1 x 32 Scale 1 x 1 x 32
4	ReLU	112 x 92 x 32	-
5	Max Pooling	56 x 46 x 32	-
6	Convolution	56 x 46 x 64	Weights 5 x 5 x 32 x 64 Bias 1 x 1 x 64
7	Batch Normalization	56 x 46 x 64	Offset 1 x 1 x 64 Scale 1 x 1 x 64
8	ReLU	56 x 46 x 64	-
9	Max Pooling	28 x 23 x 64	-
10	Fully Connected	1 x 1 x 57	Weights 57 x 41216 Bias 57 x 1
11	Softmax	1 x 1 x 57	-
12	Classification Output	1 x 1 x 57	-

Table I shows the CNN structure made up of 12 layers, which are the input layers, two convolutional layers, two batch normalization layers, two ReLU layers, two pooling layers, a fully connected layer, a Softmax layer and an output layer.

C. Software Implementation

Face detection is the first task to be done by a face recognition system after getting the input image. The proposed system uses the built-in cascade object detector for face detection. This detector uses Viola-Jones algorithm and it can be called by the vision.CascadeObjectDetector function. By default, the classification model of this function detects upright face that is facing forward, the classifiers used in this model are the weak classifiers based on classification and regression tree analysis (CART). There is another classification model that detects the same object but with classifiers which use local binary patterns (LBP) for facial features encoding. The difference between CART and LBP based classifiers is, the classification model that uses CART has the ability to model higher-order dependencies between facial features while the classification model that uses LBP is more robust to variation in illumination. There are others classification models to detect upper body, eyes, mouth and nose. Since the proposed system focuses on the face area, therefore the default classification model is used.

Next, to test the accuracy of the proposed face recognition system, folders of testing images which are not seen by the trained model were created. Fig. 8 shows some of the face recognition outcomes of the proposed system. The results obtained for testing the proposed system with three different face databases are recorded in Table II to Table IV while the comparison of the results is shown in Table V.

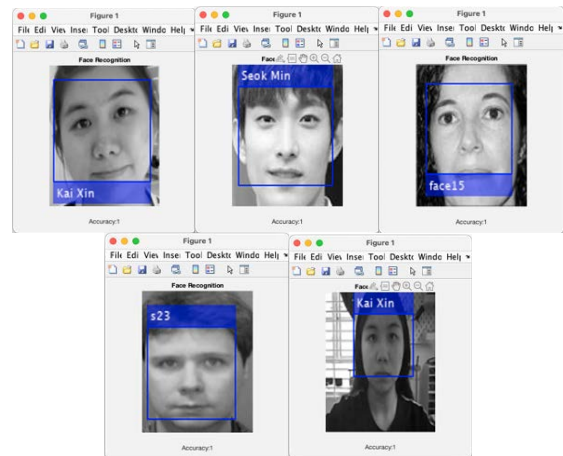


Fig. 8. Face Recognition Outcomes of the Proposed System.

The results of the proposed system are classified into three groups, the true positive (TP), false negative (FN) and false positive (FP). The result is a TP when the predicted identity matches the actual identity. FP happens when the system predicted person A but the actual identity is person B, or in other words, the input image is the face of person B but the output result returns person A. Lastly, the FN result is caused by failure in detecting and recognising faces that are known by the system, that is, when the input image is person A which is known by the system, the proposed system failed to detect and

recognise the face of the person and return the face recognition result as “Unknown”.

The face recognition accuracy is calculated by the following formula.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

There are two classes in the custom face dataset, Class 1 is made up of images of celebrity while Class 2 is made up of the images of one of the authors. From Table II, Class 1 is tested by 20 images and the accuracy is 100%. On the other hand, the proposed system is able to recognise 239 out of 250 testing images correctly from Class 2, the accuracy is 95.60%. There are four FN results, and these might be caused by the low image resolution as the images are taken by a 2-megapixel ESP32-CAM. The average accuracy of the proposed system when tested with custom face dataset is 95.93%.

The proposed software system is also tested with two different pre-curated face databases that are publicly available. Table III shows the face recognition result when the system is tested with the dataset from reference [23]. The dataset is made up of 16 classes and the proposed system is tested by four images of each class. There are 59 TP and 5 FN results, yielding an average accuracy of 92.19%. Lastly, Table IV shows the face recognition result of AT&T database with total of 40 classes. The proposed system was tested with two images of each class and successfully recognized 70 out of 80 images. The average accuracy is 87.50%.

TABLE II. FACE RECOGNITION RESULT OF CUSTOM FACE DATASET

Class	Testing Images	TP	FN	FP	Accuracy
1	20	20	0	0	100.00%
2	250	239	4	7	95.60%
Total		259	4	7	-
Average Accuracy					95.93%

TABLE III. FACE RECOGNITION RESULT OF DATASET FROM REFERENCE [23]

Class	Testing Images	TP	FN	FP	Accuracy
1	4	3	1	0	75.00%
2	4	4	0	0	100.00%
3	4	4	0	0	100.00%
4	4	4	0	0	100.00%
5	4	4	0	0	100.00%
6	4	4	0	0	100.00%
7	4	4	0	0	100.00%
8	4	4	0	0	100.00%
9	4	4	0	0	100.00%
10	4	4	0	0	100.00%
11	4	4	0	0	100.00%
12	4	4	0	0	100.00%
13	4	2	2	0	50.00%
14	4	2	2	0	50.00%
15	4	4	0	0	100.00%
16	4	4	0	0	100.00%
Total		59	5	0	-
Average Accuracy					92.19%

TABLE IV. FACE RECOGNITION RESULT OF AT&T DATABASE

Class	Testing Images	TP	FN	FP	Accuracy
1	2	2	0	0	100.00%
2	2	2	0	0	100.00%
3	2	1	1	0	50.00%
4	2	2	0	0	100.00%
5	2	2	0	0	100.00%
6	2	2	0	0	100.00%
7	2	2	0	0	100.00%
8	2	1	1	0	50.00%
9	2	2	0	0	100.00%
10	2	2	0	0	100.00%
11	2	2	0	0	100.00%
12	2	2	0	0	100.00%
13	2	2	0	0	100.00%
14	2	2	0	0	100.00%
15	2	2	0	0	100.00%
16	2	1	1	0	50.00%
17	2	2	0	0	100.00%
18	2	2	0	0	100.00%
19	2	1	1	0	50.00%
20	2	2	0	0	100.00%
21	2	2	0	0	100.00%
22	2	2	0	0	100.00%
23	2	2	0	0	100.00%
24	2	2	0	0	100.00%
25	2	2	0	0	100.00%
26	2	1	1	0	50.00%
27	2	2	0	0	100.00%
28	2	2	0	0	100.00%
29	2	2	0	0	100.00%
30	2	2	0	0	100.00%
31	2	1	1	0	50.00%
32	2	2	0	0	100.00%
33	2	2	0	0	100.00%
34	2	0	2	0	0.00%
35	2	1	1	0	50.00%
36	2	2	0	0	100.00%
37	2	1	1	0	50.00%
38	2	2	0	0	100.00%
39	2	2	0	0	100.00%
40	2	2	0	0	100.00%
Total		70	10	0	-
Average Accuracy					87.50%

TABLE V. COMPARISON BETWEEN THE FACE RECOGNITION ACCURACY OF DIFFERENT FACE DATABASES

Face Database	Total Training Images	Total Testing Images	Accuracy
AT&T	320	80	87.50%
Dataset from reference [23]	244	64	92.19%
Custom face dataset	1250	250	95.93%

From Table V, the face recognition result for the images from the custom dataset is the highest, with the accuracy of 95.93%, while face recognition result using AT&T database is the lowest, with 87.50%. The accuracy of the face recognition for the custom face dataset is the highest because the two other face databases were trained with less than 20 images for a subject while the custom face dataset created by the researcher has 90 training images for the first class and 1160 for the second class. This is to ensure the result for images taken by the ESP32-CAM to have a higher accuracy. Therefore, with the face recognition result, it can be concluded that the goal is achieved. In short, the overall accuracy of the proposed system is 98.48%.

Furthermore, an experiment is carried out to find out the maximum distance between the camera and the face so that the face can be detected and recognised by the proposed system. The ESP32-CAM is placed at a certain distance from the face and five images are taken for each distance. The effect of distance between face and the ESP32-CAM on face detection rate and face recognition accuracy is recorded in Table VI. The results show that when the distance is beyond 200 cm, the face cannot be detected by the proposed system and the face recognition process cannot be carried out. This limitation could be overcome in future research by using a higher resolution camera module.

Table VII shows the accuracies of different face recognition systems that use custom face dataset. The proposed system is made up of hardware to capture the image of a kidnap suspect and software to identify the kidnap suspect, therefore a custom face dataset is created to test the functionality of the whole system. The accuracy of the proposed system is higher than the previous studies shown in Table VII.

TABLE VI. EFFECT OF DISTANCE BETWEEN THE FACE AND THE ESP32-CAM ON FACE DETECTION RATE AND FACE RECOGNITION ACCURACY

Distance (cm)	Testing Images	Images with detected face	Face detection success rate	Face recognition Accuracy
30	5	5	100.00%	100.00%
40	5	5	100.00%	100.00%
50	5	5	100.00%	100.00%
60	5	5	100.00%	100.00%
70	5	5	100.00%	100.00%
80	5	5	100.00%	100.00%
90	5	5	100.00%	100.00%
100	5	5	100.00%	100.00%
110	5	5	100.00%	100.00%
200	5	5	100.00%	100.00%
300	5	0	0%	0%

TABLE VII. ACCURACY OF FACE RECOGNITION SYSTEMS WITH CUSTOM FACE DATASET

Method	Accuracy
Eigenface [3]	85.00%
BPANN [6]	82.00%
ENN, Curvelet transform, HSI Colour Space [7]	94.00%
Deep learning [13]	95.02%
Deep CNN [14]	81.25% (Frontal view) 75.00% (Side view) 43.75% (Downwards) view)
CNN [15]	87.00%
Proposed method	95.93%

V. CONCLUSION

This article presents the development and implementation of face recognition system using neural networks. The proposed system is made up of a hardware that can capture a picture and send it to an assigned email; and a software built in MATLAB for face recognition process. The findings of this research suggested that the best training parameters for the proposed system are kernel size of 5x5, number of filters of 32 for first convolutional layer, number of filters of 64 for second convolutional layer and initial learning rate of 0.001. The proposed system is robust as its overall face recognition accuracy is 98.48%. The limitation of the system is when the distance between face and ESP32-CAM is beyond 200 cm, face detection and recognition process cannot be carried out. The recommendations for future research include, using a higher resolution camera module, larger custom face dataset and hybrid approach of face recognition techniques that can increase the face recognition accuracy of the system.

ACKNOWLEDGMENT

The authors would like to thank Centre for Research and Innovation Management (CRIM) for the support given to this research by Universiti Teknikal Malaysia Melaka (UTeM). We thank also those who contributed in any other forms for providing their continuous support throughout this work.

REFERENCES

- [1] M. S. B. Maind and M. P. Wankar, "Research Paper on Basic of Artificial Neural Network", IJRITCC, vol. 2, no. 1, pp. 96-100, Jan. 2014.
- [2] M. Sharif, F. Naz, M. Yasmin, M. A. Shahid and A. Rehman, "Face Recognition: A Survey", Journal of Engineering Science and Technology Review, vol. 10, no. 2, p.p 166-177, 2017.
- [3] Y. Kortli, M. Jridi, A. Al Falou, M. Atri, "Face Recognition Systems: A Survey", Sensors, vol. 20, no. 2, p.p 342, 2020.
- [4] R. Rosnelly, M. S. Simanjuntak, A. Clinton Sitepu, M. Azhari, S. Kosasi, and Husen, "Face Recognition Using Eigenface Algorithm on Laptop Camera," in 2020 8th International Conference on Cyber and IT Service Management, CITSM 2020, 2020, pp. 1-4.
- [5] T. Mantoro, M. Ayu, Suhendi, "Multi-Faces Recognition Process Using Haar Cascades and Eigenface Methods", 2018 6th International Conference on Multimedia Computing and Systems (ICMCS), pp. 1-5.
- [6] T. H. Le, "Applying Artificial Neural Networks for Face Recognition", Advances in Artificial Neural Systems, vol. 2011, Article ID 673016, 16 pages, 2011. <https://doi.org/10.1155/2011/673016>.
- [7] M. M. Hussein, A. H. Mutlag, H. Shareef, "Developed artificial neural network based human face recognition", Indonesian Journal of Electrical Engineering and Computer Science, vol. 16, no. 3, pp 1279-1285, 2019.

- [8] A. S. Abdullah, M. A. Abed, I. Al_Barazanchi, "Improving face recognition by elman neural network using curvelet transform and HSI color space", *Periodicals of Engineering and Natural Sciences*, vol. 7, no. 2, pp 430-437, 2019.
- [9] S. A. Baker, H. H. Mohammed, H.A. Aldabagh, "Improving face recognition by artificial neural network using principal component analysis", *TELKOMNIKA Telecommunication, Computing, Electronic and Control*, vol. 18, no. 6, pp 3357-3364, 2020.
- [10] K.B Pranav, J. Manikandan, "Design and Evaluation of a Real-Time Face Recognition System using Convolutional Neural Networks", *Procedia Computer Science*, vol. 171, pp 1651-1659, 2020.
- [11] M. Z. Khan, S. Harous, S. U. Hassan, M. U. Ghani Khan, R. Iqbal, and S. Mumtaz, "Deep Unified Model for Face Recognition Based on Convolution Neural Network and Edge Computing," *IEEE Access*, vol. 7, pp. 72622–72633, 2019.
- [12] Fedorchenko, I., Oliinyk, A., Stepanenko, A., Zaiko, T., Korniienko, S., Burtsev, N. "Development of a genetic algorithm for placing power supply sources in a distributed electric network". *European Journal of Enterprise Technologies*, issue 5/101, 6–16 (2019), doi: 10.15587/1729-4061.2019.180897.
- [13] U. Aiman, V. Vishwakarma, "Face recognition using modified deep learning neural network", 2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT), pp. 1-5.
- [14] Fedorchenko, I., Oliinyk, A., Stepanenko, A., Svyrydenko, A., Goncharenko, D. "Genetic method of image processing for motor vehicle recognition". 2019 2nd International Workshop on Computer Modeling and Intelligent Systems, CMIS, 2019, Zaporizhzhia, April 15-19, CEUR Workshop Proceedings, Vol. 2353, pp. 211-226.
- [15] S. AbdELminaam D, Almansori AM, Taha M, Badr E, "A deep facial recognition system using computational intelligent algorithms", *PLoS ONE*, vol. 15, no. 12, pp. 1-27, 2020.
- [16] Oliinyk, A., Fedorchenko, I., Stepanenko, A., Katschan, A., Fedorchenko, Y., Kharchenko, A., Goncharenko, D. "Development of genetic methods for predicting the incidence of volumes of emissions of pollutants in air". 2019 2nd International Workshop on Informatics and Data-Driven Medicine, IDDM, CEUR Workshop Proceedings, 2019, Vol.2488, pp. 340–353.
- [17] M. Arsenovic, S. Sladojevic, A. Anderla, S. Darko, "FaceTime—Deep learning-based face recognition attendance system," 2017 IEEE 15th International Symposium on Intelligent Systems and Informatics (SISY), 2017, pp. 53-58, doi: 10.1109/SISY.2017.8080587.
- [18] Nurkhamid, P. Setialana, H. Jati, R. Wardani, Y. Indrihapsari, N. Norwawi, "Intelligent Attendance System with Face Recognition using the Deep Convolutional Neural Network Method", *Journal of Physics: Conference Series*, 1737, 012031, 2021.
- [19] I. G. S. M. Diyasa, A. Fauzi, M. Idhom and A. Setiawan, "Multi-face Recognition for the Detection of Prisoners in Jail using a Modified Cascade Classifier and CNN", 2020 2nd International Conference on Science & Technology, 1884, 012005, 2021, doi:10.1088/1742-6596/1844/1/012005.
- [20] Oliinyk, A., Fedorchenko, I., Stepanenko, .Rud M., Goncharenko, D. Implementation of evolutionary methods of solving the travelling salesman problem in a robotic warehouse // *Lecture Notes on Data Engineering and Communications Technologies*, 2021, 48, P. 263–292.
- [21] P. Kamenecay, M. Benco, T. Mizdos, R. Radil, "A New Method for Face Recognition using Convolutional Neural Network", *Advances in Electrical and Electronic Engineering*, vol. 15, no. 4, pp. 663.672, 2017, DOI: 10.15598/aeee.v15i4.2389.
- [22] Fedorchenko, I., Oliinyk, A., Stepanenko, Zaiko, T., Korniienko S., Kharchenko, A. Construction of a genetic method to forecast the population health indicators based on neural network models // *Eastern-European Journal of Enterprise Technologies*, 2020, 1 (4-103), P. 52–63. DOI: 10.15587/1729-4061.2020.197319.
- [23] M. Alghaili, Z. Li, H. Ali, "FaceFilter: Face Identification with Deep Learning and Filter Algorithm", *Hindawi Scientific Programming*, 7846264, pp. 1-9, 2020, doi.org/10.1155/2020/7846264.
- [24] N. S. Irjanto and N. Surantha, "Home Security System with Face Recognition based on Convolutional Neural Network", *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 11, p.p 408-412, 2020.
- [25] M. Srivasan and N. Ravichandran, "A new technique for Face Recognition using 2D-Gabor Wavelet Transform with 2D-Hidden Markov Model approach", *International Conference on Signal Processing, Image Processing and Pattern Recognition 2013, ICSIPR 2013*, vol. 1, p.p 151-156.
- [26] J. Bobulski, "Multimodal face recognition method with two-dimensional hidden Markov model", *Bulletin of the Polish Academy of Sciences Technical Sciences*, vol. 65, no. 1, p.p 121-128, 2017.
- [27] J. Moreano and N. Palomino, "Global Facial Recognition Using Gabor Wavelet, Support Vector Machines and 3D Face Models", *Journal of Advances in Information Technology*, vol. 11, no. 3, p.p 143-148, 2020.
- [28] Z. Rustam and R. Faradina, "Face Recognition to Identify Look-Alike Faces using Support Vector Machine", *Journal of Physics: Conference Series*, vol. 1108, 012071, 2018.
- [29] F. Hashmi (2021), Face recognition using deep learning CNN in python [Online]. Available: <https://thinkingneuron.com/face-recognition-using-deep-learning-cnn-in-python/>. [Accessed: 11-Dec-2021].