Academic Journal of Electrical and Computer Engineering
Vol. 02, No. 01, August 2026
https://doi.org/10.31272/ajece.48
https://ajece.uomustansiriyah.edu.iq
ISSN 3079-2878
Research Article

Facial Recognition Utilizing the SqueezeNet Deep Learning Model

Hussein A. Hussein Al-Delfi

Electrical Engineering Department, College of Engineering, Mustansiriyah University, Baghdad, Iraq
*Email: ha65@uomustansiriyah.edu.iq

Received: 29/07/2026 Revised: 18/08/2026 Accepted: 22/08/2026

***Abstract*— Face recognition remains one of the most challenging areas in image analysis and computer vision, given its broad practical applications in biometrics, information security, law enforcement, and surveillance systems. Recent advancements in deep learning and neural networks have made it possible to achieve significant results in various fields, such as pattern recognition and image processing. Deep learning methods accelerate the learning process and facilitate data creation. This technology identifies people by analyzing facial images, with recent progress largely attributed to convolutional neural networks (CNNs). In this paper, we employ a transfer learning technique to enhance system performance by utilizing the pre-trained SqueezeNet CNN model within MATLAB Apps, which is well-suited for handling our dataset. Upon verifying the system's functionality, it was found to be highly effective in the intended facial recognition process.**

keywords— Facial Recognition; SqueezeNet; Convolutional Neural Network; Deep Learning;

I. INTRODUCTION

Facial recognition technology has become quite common in today's world. Its use in face detection and recognition systems is widespread, often going unnoticed by many people, even as they use it on their smartphones. Advanced software designed for face detection and recognition is also employed by different surveillance operators to identify criminal faces in crowds [1]. face detection and recognition systems are crucial components in security systems, biometric identification, and human-computer interface applications. Over time, numerous methods and algorithms have been developed to tackle these challenges, each offering unique advantages and limitations. This research utilizes MATLAB programming to conduct face detection and recognition, employing both traditional methods and cutting-edge techniques. A comprehensive analysis is conducted within the study to assess face detection and recognition methods using MATLAB, examining both existing and innovative approaches [2]. Modern face detection and recognition systems face multiple challenges, particularly with facial occlusions, changes in expression, and managing large image databases. To address these issues, scientists need to develop new algorithms and improved models, along with advanced implementation techniques. Future research aims to integrate data from 3D sensors to create robust systems capable of performing effectively under diverse environmental conditions [3]. There are also specific architectures designed for specific applications; for instance, Convolutional Neural Networks (CNNs) are more efficient at learning features with spatial structure, such as in computer vision, which include object detection and recognition of faces. Transfer learning is a technology of growing importance for facial recognition, a well-studied technology which is essential for surveillance, security and human-computer interaction. Transfer learning not only saves training time, but also makes it easier to implement and results in good accuracy even when the number of training samples is small, compared to the technique of training networks from scratch with random weight initialization.

II. RELATED WORK

The Transfer Learning approach was adopted with a CNN architecture (VGG16) to create a real-time face recognition system. The image preprocessing was performed using the popular open-source package OpenCV and the CNN was the basic deep learning framework for feature extraction and facial classification [4].In this paper, the different techniques and algorithms used in face detection using the Python language with the basic principle of feature extraction and pattern recognition is evaluated. This study compares the performance, pros, and cons of the three key frameworks OpenCV, Dlib and the face_recognition library used for facial recognition [5]. The aim of the following paper is to develop a deep learning based facial recognition system, which includes face detection, feature extraction and classification. The validity of the system results is tested on the standard FEI face database, against various detection frameworks and compared to existing literature to guarantee a rigorous evaluation [6].In this research, face recognition model based on Convolutional Neural Network (CNN) is proposed which is highly efficient and has achieved good recognition rate on AT&T and AR face dataset [7].Gao et al. gave a detailed overview of face recognition with deep learning methods, pointing out important developments and new trends in the industry. They reviewed the literature systematically and classified existing methodologies and identified ongoing challenges for guiding future research. But with the continuous advancement of deep learning, some of the latest advances could go beyond their knowledge [8]. The main aspect of this work is facial recognition and the goal is to integrate what is being done in this area and to bring attention to the recent progress in this area. The authors discuss in-depth the deep learning architectures and methodologies used in the field of facial recognition, highlighting the significant advances, the current challenges and future directions for research. The review is, in some places, not critical enough though, and some of the approaches discussed are not the latest in the best of the art [9] The authors of Jia et al. introduced a 3D deep learning model for face recognition, which uses the spatial binarization model. Their method combines advanced deep learning architectures with 3D facial representations, making it effective to combine local binarization models to obtain strong facial feature representations. It is, however, strictly based on 3D geometric information, therefore it cannot be used directly for the conventional 2D image pipelines [10]. Li et al. presented an improved method for face recognition based on multi-connection Convolutional Neural Networks (CNNs), which was able to jointly learn the complex combination of facial shape, texture and color profile. They found the validation test to be of high performance, but the high computational workload of the model means it is not easy to deploy in resource-limited environments. In addition, there were longer and less predictable training periods and doubts about generalization to the real world due to potential sampling bias and insufficient data on which to rely. Structural optimization and wider evaluation in reality are still important research areas and need to be addressed to overcome these constraints [11]. This paper proposes a face recognition and identification system based on deep learning and details the design and implementation of this system. This paper proposes the end-to-end design and implementation of a deep learningbased face recognition and identification system. It describes the entire development process from data pre-processing and training of Convolutional Neural Network (CNN) to real-time face recognition and classification [12]. In this paper, an embedded face recognition and execution control system based on edge computing architecture, which is composed of a Raspberry Pi and an STM32F103C8T6 microcontroller, is presented. A hardware-software co-design is achieved, with the Raspberry Pi being responsible for image acquisition and the deep learning facial recognition and the STM32 handling the low-level actuators based on the output of the facial recognition. [13]

III. Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNNs) are a specialized class of deep learning architectures widely applied in computer vision. Their name derives from the inclusion of distinct convolutional hidden layers designed to automatically extract spatial features. These hidden layers typically comprise convolutional, pooling, batch normalization, and fully connected (dense) layers. Essentially, this setup means that convolution and pooling operations are employed as activation functions, in contrast to the standard activation functions mentioned earlier. Figure (1) show an example of CNN

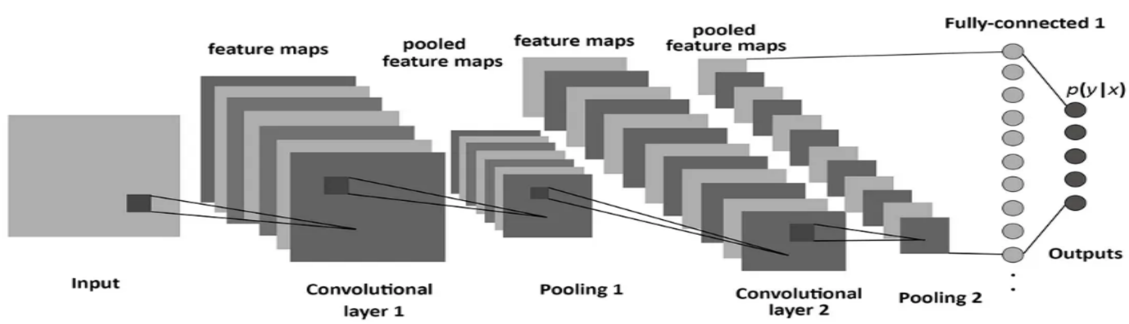

Fig. 1. An example of CNN [14]

A. Convolution layer

The convolutional layer serves as the core functional building block of the network. Instead of processing entire images at once, it slides small filter kernels—typically with odd dimensions such as 3 × 3 or 5 × 5 across the input volume. By computing local operations over these receptive fields, the layer progressively extracts high-level, abstract features from localized spatial regions [15]. The depth of the node matrix typically increases after passing through a convolutional layer because, during processing, the convolutional layer generally makes the node matrix deeper. [16]

B. Pooling layer

To make a visualization of spatial down sampling, think on transforming a high-resolution image to a lower resolution image, with the same number of color channels but with reduced spatial resolution. In the case of Convolutional Neural Networks (CNN), this reduction in the number of parameters is implemented by pooling layers that systematically sum the feature maps to reduce their spatial resolution and the number of parameters passed to the fully connected layers that follow [17].

C. Fully Connected layer

Following multiple stages of feature extraction via alternating convolutional and pooling layers, one or two fully connected layers produce the final classification output. While the convolutional and pooling operations progressively reduce spatial dimensions while encoding higher-level semantic features, the network relies on fully connected (dense) layers to map these abstracted features into final class predictions [18][19].

Various pre-trained network architectures are available for computer vision applications, including image classification, neural style transfer, generative modeling, captioning, and anomaly detection:

1. VGG19
2. (GoogLeNet)
3. ResNet50
4. EfficientNet
5.Squeeznet

SqueezeNet is an 18-layer Convolutional Neural Network (CNN) pre-trained on over one million images from the ImageNet database. Capable of classifying images across 1,000 object categories—ranging from everyday items to diverse animal species—the network has learned rich, generalized feature representations. This implementation utilizes SqueezeNet v1.1, which preserves the accuracy of SqueezeNet v1.0 while significantly reducing floating-point operations (FLOPs) per prediction. The network expects an input image resolution of 227 × 227 pixels.

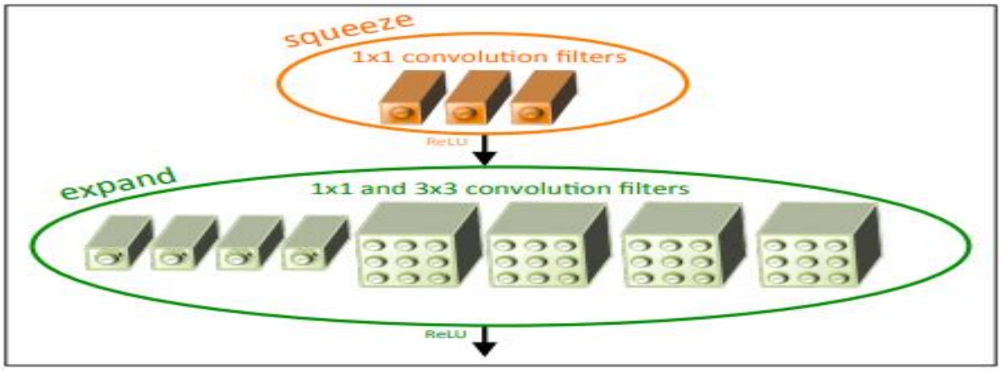

Fig.2 Overall structure of a Fire module in SqueezeNet model [20]

IV. PROPOSED METHOD:

In this paper, face recognition is carried out using the SqueezeNet pretrained model within MATLAB 2023b APPS. (Figure-3) illustrates the proposed flowchart for developing a face recognition system utilizing the MATLAB Deep Network Designer's pretrained SqueezeNet model. This workflow follows MathWorks' recommended approach for creating inference applications. The flowchart outlines a basic functional structure for data management, allowing labeled images to be imported directly from a folder for training or evaluation. Beyond using only the pretrained network, additional functionality for transfer learning is introduced, enabling the application to fine-tune SqueezeNet on a new dataset of faces. Fine-tuning a network through transfer learning is typically much faster and simpler than training a network from scratch. It allows quick transfer of learned features to a new task using a smaller number of training images.

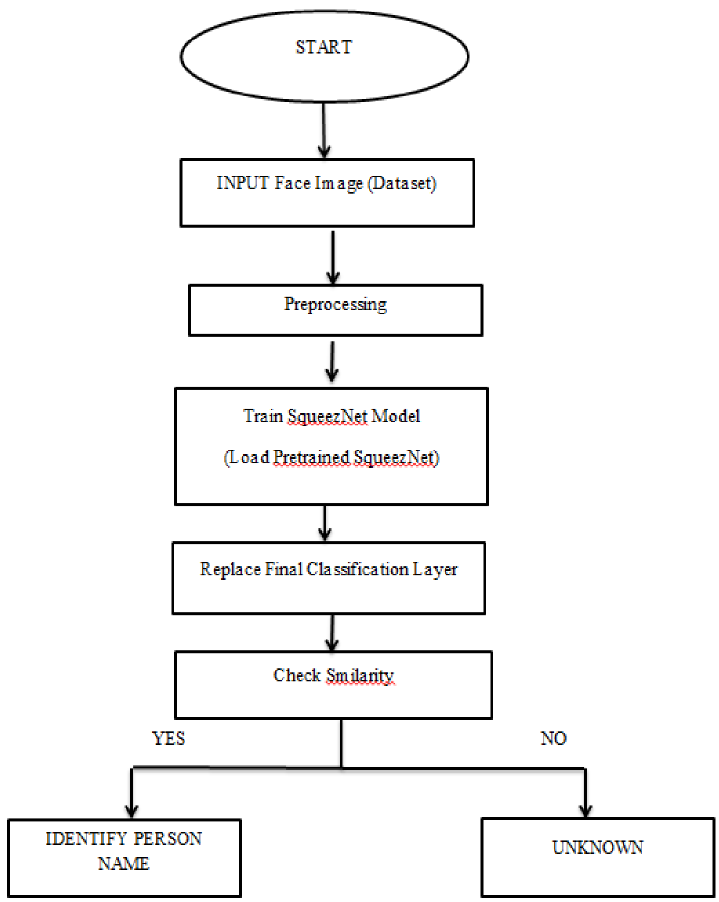

Fig.3. proposed flowchart for building a face recognition using SqueezNet

V. RESULTS AND DISCUSSION

In this paper, we consider the feasibility of SqueezeNet for the problem of face recognition, focusing on resource-constrained applications where model size and computational tractability are crucial. From the experimental results, it is shown that the architecture is an effective trade off between compactness of the model and high recognition accuracy. We used additional image data created for multiple people. The data was then processed and is displayed as in (Figure- 4), after loading the model that was already trained. The next step, as shown in (Figure- 5), was to tune the SqueezeNet architecture layers and parameters in order to be able to reflect the research objectives. This was followed by training process as shown in (figure- 6), which involved 70% training data and 30% testing data. Following the execution of the model, we demonstrated the effectiveness of our method, as illustrated in the figures below.

Fig. 4. The Data Loaded For Training

Fig.5. SqueeZnet Architecture involved modifying the layers and adjusting the parameters to suit the research objective

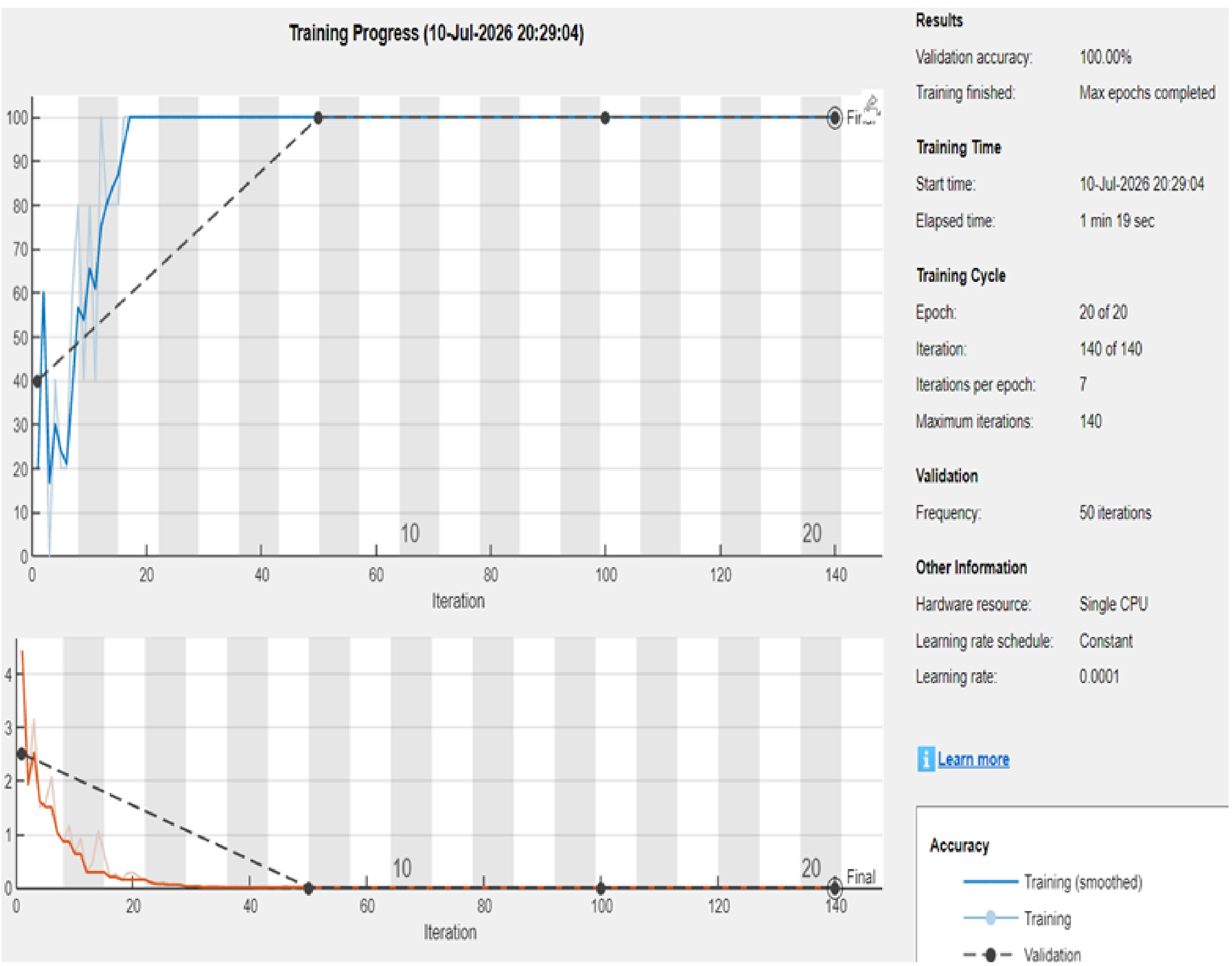

Fig.6. Training Process

VI. CONCLUSION AND FUTURE WORK:

This study identifies lightweight CNNs, with a focus on SqueezeNet, as a leading option for efficient and reliable face recognition. The innovative Fire module utilizes squeezed filters expanded by varied convolution kernels, achieving a major breakthrough. 100% accuracy on the test set. This demonstrates that smart architecture can effectively compensate for limited computational resources. Although these findings are conclusive, the next challenge is in adaptive intelligence. Future research will thus shift toward continuous learning strategies and employ a head-to-head benchmarking framework to rigorously compare SqueezeNet with its counterparts—such as MobileNet and ShuffleNet—to determine the top lightweight contender.

REFERENCES

[1] Kumar, Ashu, Amandeep Kaur, and Munish Kumar. "Face detection techniques: a review." Artificial Intelligence Review 52.2 (2019): 927- 948

[2] R. Girshick, "Fast R-CNN," in Proceedings of the IEEE International Conference on Computer Vision, IEEE, Dec. 2015, pp. 1440–1448, doi: 10.1109/ICCV.2015.169.

[3] A. Kumar and S. Srivastava, "Object Detection System Based on Convolution Neural Networks Using Single Shot Multi-Box Detector," Procedia Computer Science, vol. 171, pp. 2610–2617, 2020, doi: 10.1016/j.procs.2020.04.283.

[4] Dr.Vinayak Bharadi, Mr.Rutik Sansare, Mr.Tushar Padelkar, Mr.Vishant Shinde "Real Time Face Recognition System Using Convolutional Neural Network" International Journal of Creative Research Thoughts (IJCRT), 2022 Volume 10, Issue 4 April 2022 ,P.P 75-79| ISSN: 2320-2882

[5] Hakam Hammoud, Hamed Salman, Haidara Sakhr, Prof.Chaluva Raju P.P "Facial Detection Using Python" International Journal of Current Science (IJCSPUB), Volume 13, Issue 2 May 2023 P.P.844-849

[6] Htwe Pa Pa Win, Phyo Thu Thu Khine, Khin Nwe Ni Tun "Face Recognition System based on Convolution Neural Networks" I.J. Image, Graphics and Signal Processing, Volume 13 (2021), Issue 6,P.P 23-29

[7] Syed Muhammad Daniyal, Atiya Masood, Mansoor Ebrahim, Syed Hasan Adil, Kamran Raza" An Improved Face Recognition Method Based on Convolutional Neural Network" Journal of Independent Studies and Research Computing (JISR-C), Volume 22 Issue 1,January – June 2024 P.P 103-110

[8] Guo, Y.; Zhang, L.; Hu, Y.; He, X.; Gao, J. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In European Conference on Computer Vision; Springer: Berlin/Heidelberg, Germany, 2016; pp. 87–102.

[9] Hu, M.; Zhang, Z.; Lu, J. Face recognition using deep learning: A survey. J. Ambient. Intell. Humaniz. Comput. 2019, 10, 1–14.

[10] Jia, X.; Li, X.; Li, Y.; Li, B.; Yan, W. 3D deep learning-based facial expression recognition with spatial-temporal local binary pattern. Signal Process. Image Commun. 2020, 82, 115738.

[11] Li, S.; Wang, Z.; Tan, X.; Huang, K. Face recognition based on multi-feature fusion using deep convolutional neural network. J. Vis. Commun. Image Represent. 2018, 57, 366–376.

[12] KH Teoh2, RC Ismail, SZM Naziri, R Hussin, MNM Isa and MSSM Basir "Face Recognition and Identification using Deep Learning Approach" 5th International Conference on Electronic Design (ICED) 2020

[13] Chen Zhen, Yuru Wang "Design of face recognition system based on deep learning" MLNN 2025, April 18–20, 2025, Suzhou, China ,P.P 103-108

[14] Nigam, V. (2018, September 10). Understanding Neural Networks. From neuron to RNN, CNN, and Deep Learning. Medium. vidhya/understanding-neural-networks-from-neuron-to-rnn-cnn-and-deep-learning-cd88e90e0a90 Ray,

[15] H. ben Fredj, S. Sghaier, and C. Souani, "An efficient face recognition method using CNN," in 2021 International Conference of Women in Data Science at Taif University, WiDSTaif 2021, Institute of Electrical and Electronics Engineers Inc., Mar. 2021. doi: 10.1109/WIDSTAIF52235.2021.9430209.

[16] A. George, Z. Mostaani, D. Geissenbuhler, O. Nikisins, A. Anjos, and S. Marcel, "Biometric Face Presentation Attack Detection With Multi-Channel Convolutional Neural Network," IEEE Transactions on Information Forensics and Security, vol. 15, no. 1, pp. 42–55, 2020, doi: 10.1109/TIFS.2019.2916652.

[17] J. H. Kim, B. G. Kim, P. P. Roy, and D. M. Jeong, "Efficient facial expression recognition algorithm based on hierarchical deep neural network structure," IEEE Access, vol. 7, pp. 41273–41285, 2019, doi: 10.1109/ACCESS.2019.2907327.

[18] Y. Ma, H. J. Wang, J. Wan, Z. H. Wang, Y. Yang, and C. Huang, "Design and implementation of face recognition system based on convolutional neural network," in Journal of Physics: Conference Series, IOP Publishing Ltd, Sep. 2021. doi: 10.1088/1742-6596/2029/1/012096.

[19] W. Liu, L. Zhou, and J. Chen, "Face recognition based on lightweight convolutional neural networks," Information (Switzerland), vol. 12, no. 5, May 2021, doi: 10.3390/info12050191.

[20] Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J. and Keutzer, K., "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and< 0.5 mb model size", arXiv preprint arXiv:1602.07360, (2016).