

Est.
1841

YORK
ST JOHN
UNIVERSITY

Mohammad, Ahmad Saeed ORCID

logoORCID: <https://orcid.org/0000-0001-6141-2605>, Jarullah, Thoalfeqar G. ORCID logoORCID: <https://orcid.org/0009-0000-2548-9648>, Al-Kaltakchi, Musab T. S. ORCID logoORCID: <https://orcid.org/0000-0001-5542-9144>, Alshehabi Al-Ani, Jabir ORCID logoORCID: <https://orcid.org/0000-0002-0553-2538> and Dey, Somdip ORCID logoORCID: <https://orcid.org/0000-0001-6161-4637> (2024) IoT-MFaceNet: Internet-of-Things-Based Face Recognition Using MobileNetV2 and FaceNet Deep-Learning Implementations on a Raspberry Pi-400. *Journal of Low Power Electronics and Applications*, 14 (3). p. 46.

Downloaded from: <https://ray.yorks.ac.uk/id/eprint/10668/>

The version presented here may differ from the published version or version of record. If you intend to cite from the work you are advised to consult the publisher's version:

<https://doi.org/10.3390/jlpea14030046>

Research at York St John (RaY) is an institutional repository. It supports the principles of open access by making the research outputs of the University available in digital form. Copyright of the items stored in RaY reside with the authors and/or other copyright owners. Users may access full text items free of charge, and may download a copy for private study or non-commercial research. For further reuse terms, see licence terms governing individual outputs. [Institutional Repository Policy Statement](#)



RaY

Research at the University of York St John

For more information please contact RaY at ray@yorks.ac.uk

Article

IoT-MFaceNet: Internet-of-Things-Based Face Recognition Using MobileNetV2 and FaceNet Deep-Learning Implementations on a Raspberry Pi-400

Ahmad Saeed Mohammad ¹, Thoalfeqar G. Jarullah ¹, Musab T. S. Al-Kaltakchi ², Jabir Alshehabi Al-Ani ³
and Somdip Dey ^{3,4,*}

¹ Department of Computer Engineering, College of Engineering, Mustansiriyah University, Baghdad 10047, Iraq; ahmad.saeed@uomustansiriyah.edu.iq (A.S.M.); egm2030@uomustansiriyah.edu.iq (T.G.J.)

² Department of Electrical Engineering, College of Engineering, Mustansiriyah University, Baghdad 10047, Iraq; m.t.s.al_kaltakchi@uomustansiriyah.edu.iq

³ Department of Data Science, York St. John University, York YO31 7EL, UK; j.alshehabialani@yorks.ac.uk

⁴ Nosh Technologies, 14 Miranda Walk, Colchester, Colchester CO4 3SL, UK

* Correspondence: s.dey@yorks.ac.uk or dey@nosh.tech

Abstract: IoT applications revolutionize industries by enhancing operations, enabling data-driven decisions, and fostering innovation. This study explores the growing potential of IoT-based facial recognition for mobile devices, a technology rapidly advancing within the interconnected IoT landscape. The investigation proposes a framework called IoT-MFaceNet (Internet-of-Things-based face recognition using MobileNetV2 and FaceNet deep-learning) utilizing pre-existing deep-learning methods, employing the MobileNetV2 and FaceNet algorithms on both ImageNet and FaceNet databases. Additionally, an in-house database is compiled, capturing data from 50 individuals via a web camera and 10 subjects through a smartphone camera. Pre-processing of the in-house database involves face detection using OpenCV's Haar Cascade, Dlib's CNN Face Detector, and Mediapipe's Face. The resulting system demonstrates high accuracy in real-time and operates efficiently on low-powered devices like the Raspberry Pi 400. The evaluation involves the use of the multilayer perceptron (MLP) and support vector machine (SVM) classifiers. The system primarily functions as a closed set identification system within a computer engineering department at the College of Engineering, Mustansiriyah University, Iraq, allowing access exclusively to department staff for the department rapporteur room. The proposed system undergoes successful testing, achieving a maximum accuracy rate of 99.976%.

Keywords: Internet-of-Things applications; MobileNetV2; FaceNet; deep learning; Raspberry Pi type-400; facial identification; mobile application



Citation: Mohammad, A.S.; Jarullah, T.G.; Al-Kaltakchi, M.T.S.; Alshehabi Al-Ani, J.; Dey, S. IoT-MFaceNet: Internet-of-Things-Based Face Recognition Using MobileNetV2 and FaceNet Deep-Learning Implementations on a Raspberry Pi-400. *J. Low Power Electron. Appl.* **2024**, *14*, 46. <https://doi.org/10.3390/jlpea14030046>

Academic Editor: Stefania Perri

Received: 6 August 2024

Revised: 27 August 2024

Accepted: 30 August 2024

Published: 5 September 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction and Literature Review

Face recognition technology accurately identifies individuals by analyzing their facial features. Its applications include security, surveillance, authentication, and personalized user experiences. Currently, it is being actively developed and deployed in various scenarios, enhancing security measures and enabling convenient authentication methods in consumer electronics. Additionally, it aids in personalized advertising and customer engagement by analyzing demographic data. Overall, face recognition technology evolves and finds new applications across diverse industries, improving security, efficiency, and the user experience [1–10].

Significant advancements are being achieved in biometric security, surveillance, and personalized user experiences. This is attributed to the intricate algorithms present in deep-learning models, resulting in a substantial enhancement of our ability to comprehend facial features [11–13]. Facial recognition technology powered by deep-learning algorithms is

integrated into IoT systems, thereby revolutionizing interactions and services across industries. This integration facilitates personalized experiences, customized services, and secure access management through real-time analysis of facial features. Consequently, the way the interconnected world is experienced and interacted with is being redefined. This transformative technology reshapes perceptions and engagements within the interconnected landscape of IoT applications across various sectors, including retail, smart homes, healthcare, and beyond [14–18].

One of the primary challenges of computing face recognition on resource-constrained IoT devices such as the Raspberry Pi-400 or other ARM big.LITTLE architecture-based multiprocessor-system-on-a-chip (MPSoC) is the limited processing power and memory [19–24]. These devices must handle intensive computations required by deep-learning models, which can lead to slower processing times and reduced real-time performance. Additionally, the energy consumption of running such models on low-powered devices can be significant, which is a crucial factor for battery-operated IoT devices. Our proposed work (IoT-MFaceNet) addresses these challenges by optimizing the face recognition models through techniques like TensorFlow Lite [25,26] and model complexity reduction (squeezing) [27]. By reducing the model size and improving inference speed, we ensure that our face recognition system operates efficiently on the Raspberry Pi-400 without compromising accuracy.

The benefits of computing face recognition on IoT-based devices such as the Raspberry Pi-400 include enhanced accessibility and cost-effectiveness. These devices are affordable and can be easily integrated into various applications, making advanced facial recognition technology accessible for small-scale projects and educational purposes. Furthermore, deploying face recognition on edge devices like the Raspberry Pi-400 reduces the dependency on cloud services, enhancing privacy and security by keeping data processing local. This also minimizes latency, leading to faster response times and improved user experiences in real-time applications [21,24]. Our work demonstrates that with proper optimization, these benefits can be fully realized without sacrificing the performance of facial recognition systems.

The review article in [28] extensively discusses the wide-ranging applications of deep learning in computer vision, natural language processing, illness prediction, drug development, and healthcare. The surge in deep learning's popularity is linked to the escalating volumes of big data, IoT, interconnected devices, and robust processing technologies such as graphics processing units (GPUs) and tensor processing units (TPUs). The primary data sources for these systems include medical IoT, digital images, electronic health records (EHRs), genetic data, and centralized medical databases. Examining 44 research articles published between 2010 and 2020 within this framework highlights the pivotal role that deep learning plays in Internet-of-Things applications for healthcare systems. The study analyzes crucial applications, technical categorizations, and theoretical principles, and conducts an assessment of the benefits, limitations, constraints, and recommendations for future research. The exploration of additional deep-learning applications in various domains of machine learning, artificial intelligence, and signal processing is discussed [29–31].

The unique framework for IoT image-based identification across all IoT domains, as introduced in the study in [32], involves a fusion of principal component analysis (PCA) and deep learning. By proficiently extracting distinctive image features using PCA and image modification techniques, consistently robust experimental results are obtained. The significant dispersion following projection notably facilitates image identification within the context of IoT. High recognition performance is observed with PCA, particularly with a dimension of 25, while the recommended convolutional neural network (CNN) algorithm surpasses conventional methods. Future explorations aim at utilizing existing image data for extracting picture features through linear discriminant analysis (LDA) and improving the approach with adaptive activation functions and a hybrid machine-learning approach for broader IoT applications.

The challenge of handling extensive medical data is tackled in the research [33], emphasizing the importance of employing advanced computational techniques such as machine learning and big data analytics. The research focuses on the utilization of big data analysis to improve decision making in healthcare by adapting machine-learning algorithms. It extensively examines the role of big data in healthcare, outlining different methodologies for extracting insights from enormous datasets.

The research presented in [34] introduces an innovative framework that integrates PCA and deep learning for an IoT image-based identification system applicable across various industries. This framework proficiently extracts unique image characteristics, consistently producing robust experimental outcomes by employing image modification techniques and PCA algorithms. Post-projection notably assists in picture recognition within the Internet-of-Things context. Excellent recognition outcomes are achieved using PCA, particularly with a dimension of 25, while the proposed CNN algorithm surpasses conventional methods. Future directions include exploring visual feature extraction with LDA and harnessing existing image data. Recommendations involve enhancing the system by incorporating adaptive activation functions and employing a hybrid machine-learning approach to broaden its applicability in the IoT domain.

This paper aims to create a resilient facial recognition pipeline system tailored for practical application on low-powered devices for security and enrollment purposes. In addition, our work delves into diverse facial recognition methodologies, ranging from manually constructed features to deep learning and CNNs. We also include augmentation and pre-processing techniques to improve accuracy and broaden the training dataset. Furthermore, optimization strategies are proposed to enhance the speed of the facial recognition system during inference. The primary contributions encompass several key aspects:

- Proposed framework called **IoT-MFaceNet** (Internet of Things-based face recognition using **MobileNetV2** and **FaceNet** deep-learning) that integrates diverse applications, such as IoT-based mobile devices, were utilized to support facial recognition tasks. IoT-MFaceNet utilizes deep-learning networks based on various structures of MobileNetV2 and FaceNet, extensively evaluating them using powerful tools.
- A newly curated household image database undergoes preprocessing before applying deep-learning techniques from pre-trained ImageNet and FaceNet databases. These techniques, implemented through diverse hidden layer structures in MobileNetV2 and FaceNet models, enable compression using the TensorFlow Lite for evaluation on the Raspberry Pi 400 platform.
- Optimization techniques like TensorFlow Lite and model complexity reduction (squeezing) are applied to create new models that retain the original accuracy while improving inference speed and reducing storage requirements. Custom-designed setups for MobileNetV2 vary in layer and neuron counts, optimizing performance.

This paper's structure is as follows: Section 1 provides background to the work and highlights the contributions. Section 2 explores the implementation of related research, covering both software and hardware aspects. It highlights the utilization of MobileNetV2 and FaceNet in the software domain while focusing on the Raspberry Pi's role in the hardware context. Section 3 outlines the framework structure of IoT-MFaceNet and elucidates the functioning of the proposed system. Section 4 presents the databases used and the experimental outcomes, and conducts an analysis. Finally, Section 5 summarizes the conclusions drawn from this study.

2. Related Work Based on the Software and Hardware Implementation

The hardware representation involves using the Raspberry Pi 400, while the software incorporates MobileNetV2 and FaceNet techniques. This section primarily focuses on the work of other researchers in the field who currently utilize Internet-of-Things applications for face recognition tasks. In this paper, to facilitate comprehension of the proposed system, the software and hardware frameworks are separated into two primary subsections.

Consequently, the related work is categorized into two subsections, corresponding to the software and hardware frameworks.

2.1. MobileNetV2 and FaceNet Approaches

MobileNetV2, a deep neural network architecture, is specifically developed for embedded and mobile devices. It represents an improvement over the initial MobileNet, providing lighter and more efficient models suitable for use on devices with limited processing capabilities, particularly for tasks such as object recognition and image classification [35]. Further research in terms of mobile environment applications is discussed in [36,37].

MobileNetV2 utilizes depthwise separable convolutions, a technique that splits the conventional convolution process into two distinct layers: depthwise convolution and pointwise convolution. This method maintains competitive performance in computer vision tasks while significantly reducing computational costs. Overall, MobileNetV2 gains recognition for its ability to effectively process and analyze visual input on resource-constrained devices, making it popular for various mobile and edge computing applications [38–41]. The MobileNetV2 method is widely adopted for various applications, as evidenced in the following studies: [42–54].

In the study referenced in [27], TensorFlow Lite and squeezing optimization techniques are utilized to develop models that boast enhanced accuracy, inference speed, and decreased storage demands. Custom designed setups for MobileNetV2, varying from large to small setups, are employed.

FaceNet, a face recognition system developed by Google, employs deep-learning techniques to generate highly discriminative embeddings and numerical representations of faces within a high-dimensional space termed as “face space”. To accomplish face mapping in this space, it utilizes a deep neural network architecture known as a Siamese network, ensuring that disparate faces are positioned farther apart while similar ones are grouped closely. Due to its embedding space, FaceNet proves to be a powerful tool for tasks related to faces in computer vision and biometrics, facilitating precise face verification, identification, and clustering [55–59].

2.2. Raspberry Pi Type-400

The Raspberry Pi type-400, a compact computer integrated into a keyboard, is powered by a Raspberry Pi 4 Model B board. This configuration provides comparable capabilities but in a convenient form resembling a keyboard. This integrated design encompasses a quad-core ARM Cortex-A72 (ARM v8) 64-bit MPSoC @ 1.8 GHz, 4 GB of LPDDR4-3200 SDRAM, storage, diverse ports (like USB and HDMI), and wireless connectivity features. As an accessible and user-friendly device, the Raspberry Pi 400 proves ideal for programming education, DIY projects, and basic computing activities [60,61].

The affordability, adaptability, and accessibility of the Raspberry Pi 400 make it beneficial for facial recognition tasks. Its cost-efficiency, sufficient computing capabilities, and potential for integration position it as an excellent platform for educational purposes, experimentation, and small-scale facial recognition initiatives. This facilitates innovation and exploration within the field [62–66].

3. Proposed Framework: IoT-MFaceNet

Figure 1 depicts the primary proposed framework, IoT-MFaceNet, comprising the following segments: Section (A) denotes the image database; Section (B) symbolizes the labeled faces in the wild (LFW) database; Section (C) represents the in-house database comprising 50 subjects captured via a web camera; Section (D) portrays the in-house database for 10 subjects obtained from a smartphone camera; and Section (E) encompasses the pre-processing phase integrating OpenCV’s Haar cascade classifier, Dlib’s CNN face detector, and Mediapipe’s face detector. Section (F) covers pre-trained deep learning, encompassing subsections (F1) MobileNetV2 technique and (F2) FaceNet technique. Section (G) illustrates the evaluation process using the Raspberry Pi type 400. Section (H) contains

the classification stage with support vector machine (SVM) and multi-Layer perceptron (MLP) classifiers, both featuring three variable hidden layers. Finally, Section (I) showcases the display of the identified person using a mobile application. Moreover, Figure 2 depicts the architecture for both the MobileNetV2 approach and VGG16 and VGG19 of the FaceNet. The framework introduces several crucial procedures. It integrates IoT and mobile technologies for facial recognition tasks, employing deep-learning networks such as MobileNetV2 and FaceNet. The proposed system operates as a closed-set identification tool within a specific university department, controlling access exclusively for authorized personnel. Testing demonstrates nearly perfect accuracy.

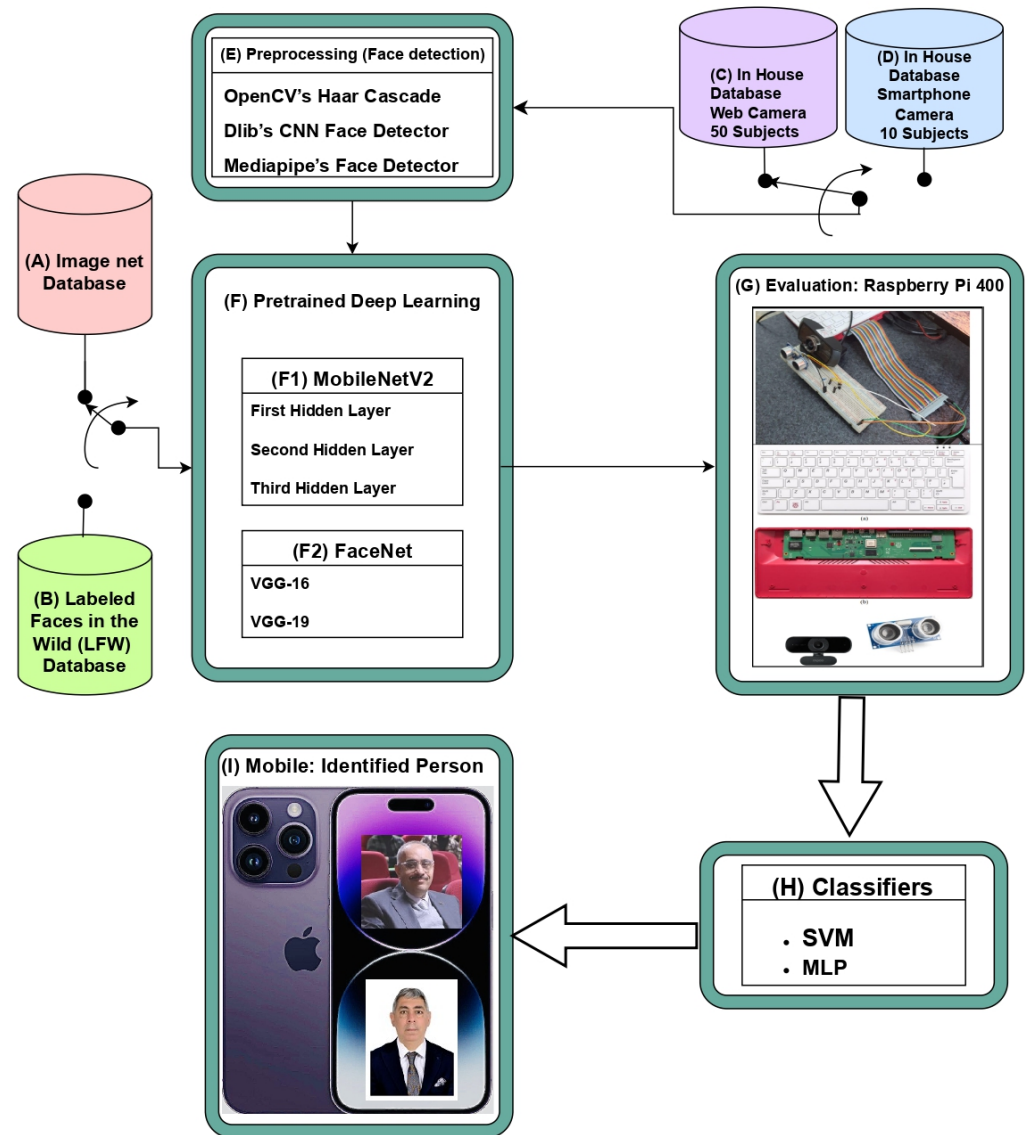


Figure 1. Proposed Framework (IoT-MFaceNet): where (A) represents the image database, (B) represents the labeled faces in the wild (LFW) database, (C) represents the in-house database with 50 subjects for the web camera, with 10 subjects, (D) represents the in-house database with 10 subjects for a smartphone camera, (E) represents pre-processing, (F) represents pre-trained deep learning, including (F1) the MobileNetV2 technique (F2) and the FaceNet technique, (G) represents the evaluation process represented by the Raspberry Pi type 400, (H) represents classification, and (I) represents the identified person displayed by exploiting the mobile application.

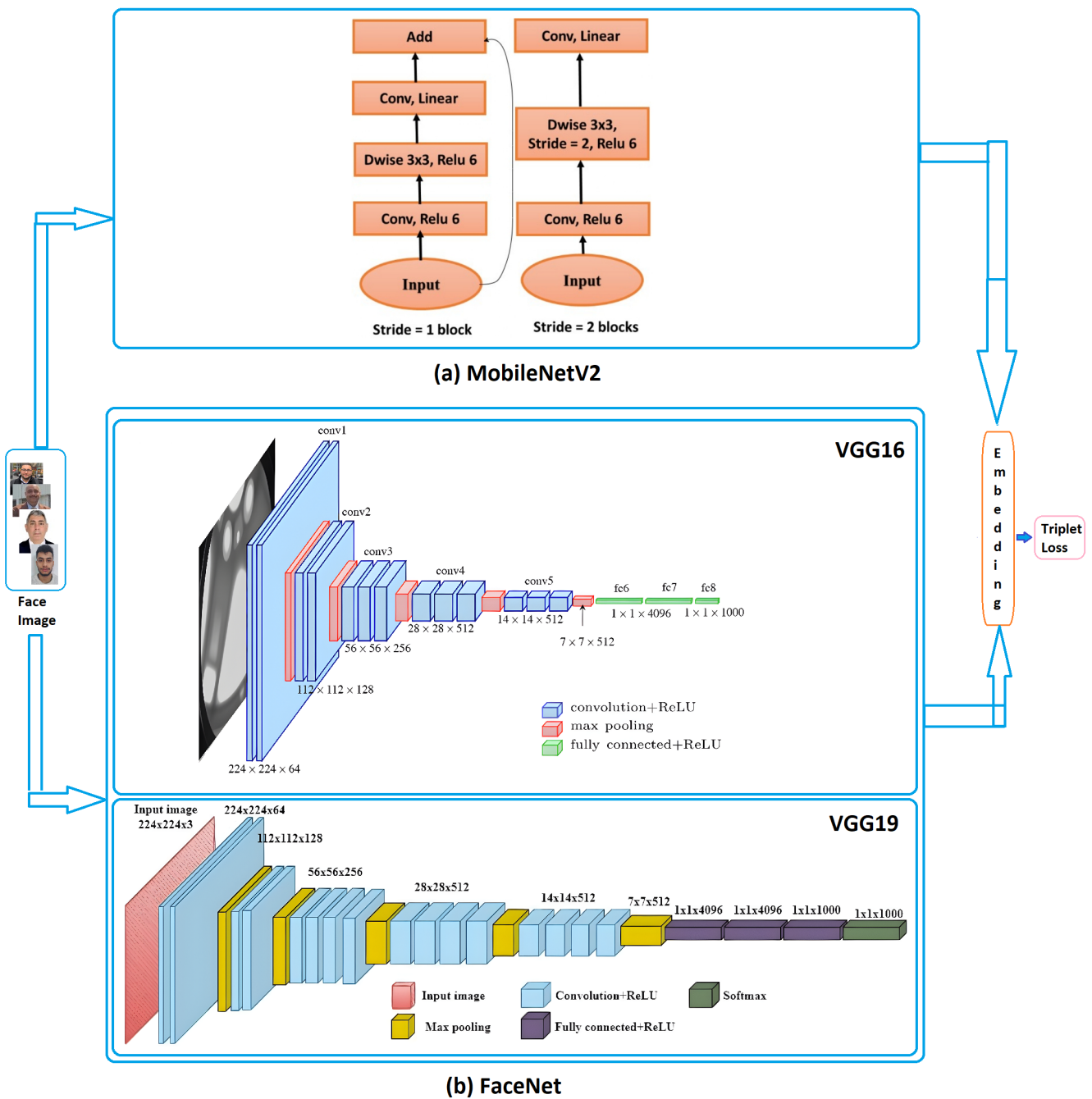


Figure 2. The architecture of the MobileNetV2 technique (a) and the architecture of VGG16 and VGG19 of the FaceNet technique (b).

The proposed architecture encompasses the compilation of a household image database, its pre-processing, and the application of deep-learning techniques sourced from established databases. These processed data undergo compression using TensorFlow Lite to enable deployment on the Raspberry Pi type 400 platform. Classifiers like SVM and MLP are utilized for individual identification, and the identified person’s image is displayed on a mobile device.

The study employs optimization methods like TensorFlow Lite and model complexity reduction, referred to as “squeezing”, to develop a new model that maintains nearly identical accuracy to the original but offers faster inference and reduced storage needs.

Google developed TensorFlow, a crucial machine-learning library utilized in this research. It introduces TensorFlow Lite specifically for optimizing models to shorten inference times.

TensorFlow Lite implements strategies like quantization, pruning, and clustering to shrink model sizes and enhance operational efficiency. In this study, weight quantization is the primary technique employed, reducing the bit allocation for storing model weights. For instance, it utilizes sixteen-bit floats, sixteen-bit integers, or eight-bit integers instead of the default 32-bit floats. This modification simplifies arithmetic operations for the central processing unit (CPU) or graphics processing unit (GPU) managing weights and input images, resulting in faster processing due to fewer bits.

Additionally, the process of squeezing a model is employed to streamline its architecture by reducing the number of layers, neurons, or both until it achieves acceptable accuracy or meets specified criteria related to size, inference speed, or memory consumption. The optimization methods discussed present various custom-designed configurations for MobileNetV2: the largest model consists of three layers, each with 384 neurons; an intermediate size configuration such as (192-256-128); and the smallest with (128-128-128). TensorFlow-Lite quantization is applied to all three networks.

Figure 3 depicts the performance of these three distinct models in terms of frames per second (FPS), a metric used to measure the speed at which a sequence of images, or frames, is displayed in media like movies, animations, or video games. Higher FPS values indicate smoother and more fluid motion in visual content. A higher frame rate per second (FPS) is often sought after in gaming and video production to deliver an enhanced viewing or gaming experience. Additionally, the FPS provides a holistic view that could describe the overall system more visually.

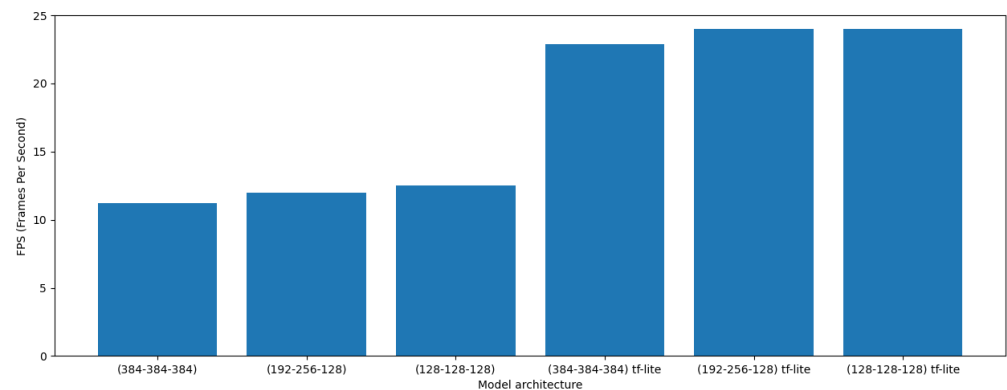


Figure 3. Performance of the different models in FPS, where the FPS results are the average FPS of 100 recorded frames. Three different custom-made tops were used to test the proposed performance using the custom-trained CNN with MobileNetV2. The first top has three hidden layers with 384 neurons each, making it the largest custom-made top for MobileNetV2 in this example (384-384-384). The second top, with a configuration of 192-256-128, is the second largest custom-made top. The third top, with a configuration of 128-128-128, is the smallest custom-made top. Additionally, TensorFlow-lite quantization was applied to all three networks.

Furthermore, Figure 4 illustrates the size of the models stored on disk, measured in megabytes (MB). The hardware arrangement in this study comprises a Raspberry Pi type 400 coupled with an ultrasonic distance sensor. This sensor gauges the distance between the camera and the subject intended for identification. The system functions by situating the ultrasonic sensor in front of the webcam to evaluate the subject's proximity to the camera. Once the subject's distance is verified, the system captures images that are then routed to the face recognition pipeline for analysis.

Figure 5 depicts a simple hardware setup involving the Raspberry Pi type 400 and a breadboard, suitable for placement on walls or desks. For precise face recognition, subjects only need to briefly face the camera from an optimal distance, typically within 80 cm, to ensure dependable outcomes. Although the system can recognize subjects beyond this

range, its confidence decreases, triggering a graphical user interface (GUI) message to alert users. The system’s performance speed relies on the hardware; a PC equipped with a Ryzen 5 3600 CPU maintains an average of 25 frames per second (FPS), while the Raspberry Pi type 400 achieves 6 FPS with multithreading.

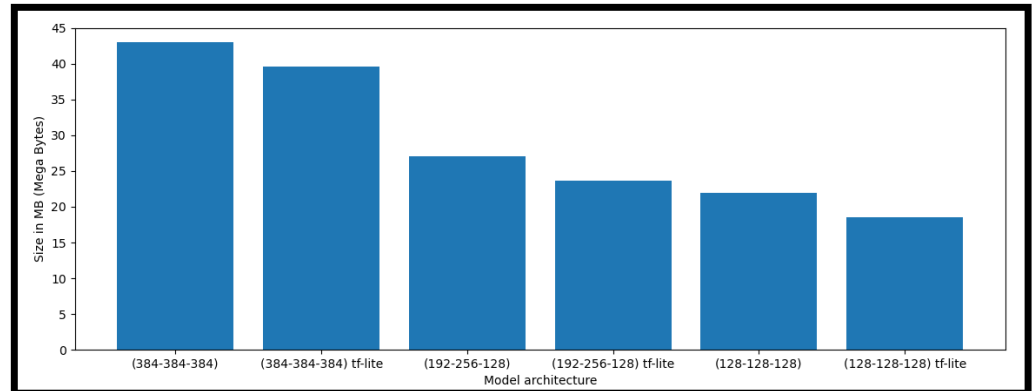


Figure 4. Size of the models created on disk in MB.

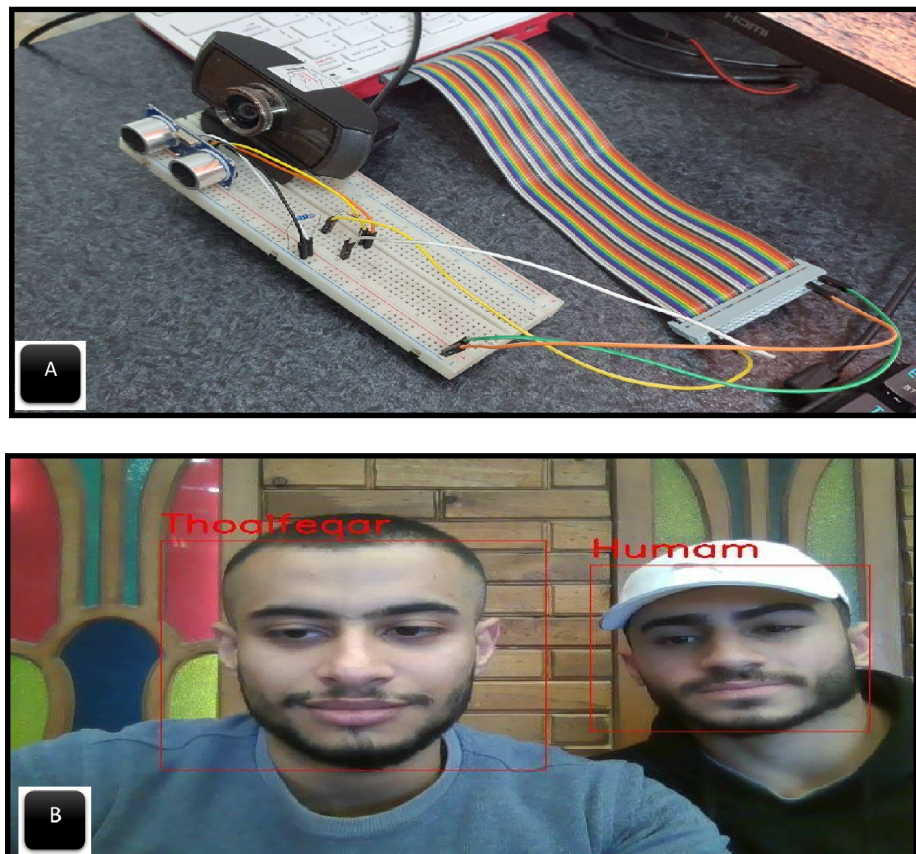


Figure 5. The hardware design. (A) Example of the hardware connection. The hardware system combines an ultrasonic sensor and webcam to measure and monitor subject distance, capturing images for face recognition. Optimal recognition occurs at distances below 80 cm, with lower confidence beyond that, signaled on the GUI. (B) An instance of the system operating with two subjects is evident. The system accurately identifies the faces of Thoalfeqar and Humam. Both serve as representatives of the fourth class in the Department of Computer Engineering at Mustansiriyah University. Typically, class representatives, including Thoalfeqar and Humam, have access to the department’s rapporteur room along with the department’s staff.

4. Database, Experimental Results, Analysis, and Discussion

This section initially describes the databases used in this paper before presenting the results.

4.1. Database

This subsection is divided into two main components: pre-training the proposed models' databases and testing databases. The proposed system utilizes labeled faces in the wild (LFW) and ImageNet databases to pre-train the proposed models. In contrast, the in-house database is employed to assess the proposed methodology.

In the LFW and ImageNet databases, labeled faces in the wild serves as a collection of face photographs created to explore unconstrained face recognition challenges. This dataset comprises over 13,000 facial images gathered from the internet, with each face annotated with the corresponding person's name. Among the individuals depicted, 1680 have two or more separate photos within the dataset. The sole limitation of these facial images is that they are identified using the Viola-Jones face detector [67], which exhibits relatively lower accuracy compared to more advanced facial recognition algorithms. The LFW database is accessible on October 29th, 2019 online through the provided links (<https://vis-www.cs.umass.edu/lfw/>), or (<https://www.kaggle.com/datasets/jessicali9530/lfw-dataset>).

The ImageNet dataset encompasses 14,197,122 images annotated according to the WordNet hierarchy. Since 2010, this dataset has been employed in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), serving as a benchmark for tasks like image classification and object detection. The publicly available dataset includes a collection of manually annotated training images [68]. The ImageNet database is accessible online since 3rd of March 2010 via the provided links (<https://paperswithcode.com/dataset/imagenet>), or (<https://www.image-net.org/download.php>).

However, the tested databases consist of in-house database Types I and II (refer to Data Availability Atatement. **(Type-I) The in-house database with 50 Subjects:** Initially, this database contains 50 subjects and 5586 images taken via a webcam without utilizing image augmentation to increase the image count per subject. Subsequently, adjustments are made to augmentation techniques, resulting in a total of 29,086 images. These new techniques encompass horizontal flipping, contrast limited adaptive histogram equalization (CLAHE), random modifications in brightness and contrast, Gaussian blur, PCA, Gaussian noise, JPEG compression, median blur, and interpolation. **(Type-II) The in-house database of 10 individuals recorded using a high-quality smartphone camera:** This database comprises 24,300 images captured using a high-quality smartphone camera and includes data from 10 individuals.

4.2. Experiment (1) Custom-Trained CNN Using MobileNetV2 and MLP Classifier

The MobileNet model is pre-trained by the ImageNet dataset. In this experiment, a customized CNN model, specifically MobileNetV2, is utilized, and in-house databases are employed to assess this tailored CNN model. Due to our hardware limitations, we opted for MobileNetV2 due to its lightweight nature and faster training capabilities compared to other deep CNNs like ResNet152 or ResNet101. To customize the model, a custom fully connected layer is crafted, comprising three hidden layers labeled as L1, L2, and L3, respectively, specifically designed for MobileNetV2. The training involves utilizing pre-existing weights from the ImageNet database to expedite the establishment of feature extraction filters.

The experiment focuses on custom-trained CNNs, incorporating three networks with differing sizes for their final classification layers. Initially trained on the in-house database (Type I-50 Subjects), these networks are later expanded by integrating data from ten new subjects captured using a phone camera (Type II) rather than a webcam. The discussion centers on comparing results before and after optimization for these three networks.

Figure 6 illustrates the receiver operating characteristic (ROC) without optimization for the custom-trained CNN utilizing MobileNetV2 and MLP classifiers for three variations

in customs: (a) (128-128-128) custom top, (b) (192-256-128) custom top, and (c) (384-384-384) custom top.

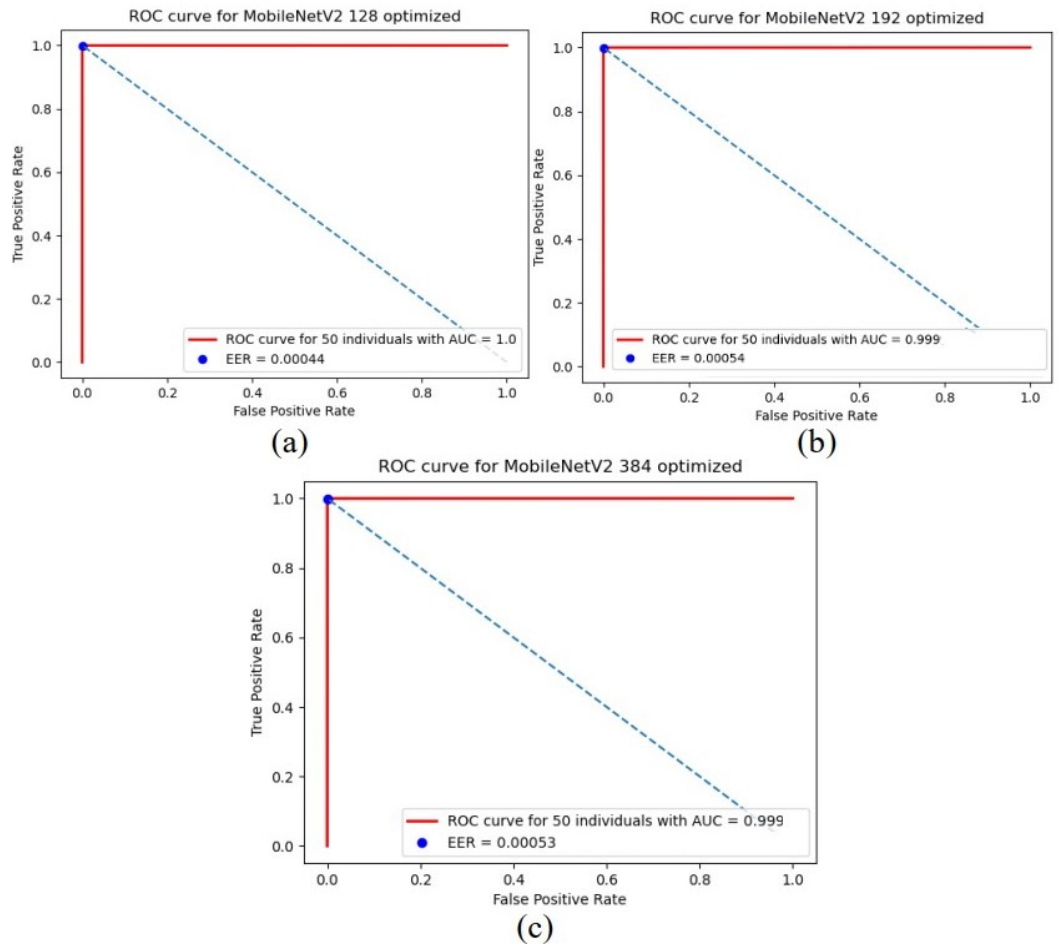


Figure 6. The receiver operating characteristic (ROC) without optimization for a custom-trained CNN using MobileNetV2 with an MLP classifier for (a) (128-128-128) custom-made top, (b) (192-256-128) custom-made top, and (c) (384-384-384) custom-made top.

Table 1 displays the performance of the custom-trained CNN system using MobileNetV2 with an MLP classifier without optimization.

Table 1. The performance system custom-trained CNN using MobileNetV2 with an MLP classifier without optimization.

Custom-Made Top	F1-Score	Precision-Score	Recall-Score	Accuracy%
128-128-128	0.998	0.998	0.998	99.913%
192-256-128	0.997	0.997	0.996	99.862%
384-384-384	0.999	0.999	0.999	99.976%

This paper develops a face-recognition pipeline suitable for edge devices like the Raspberry Pi 400 or smartphones. However, real-time execution may be hindered due to the model’s slow inference time. To address this, optimization techniques are employed, including TensorFlow-lite optimization, weight quantization, and model squeezing. Weight quantization reduces the precision of weight storage, while model squeezing simplifies the model’s complexity. These optimizations are demonstrated using the MobileNetV2 CNN with custom-made tops and TensorFlow-lite quantization.

Subsequently, we use the CNN-MobileNetV2 custom optimizer optimization to enhance system performance. Figure 7 depicts the ROC with optimization for a custom-trained

CNN using MobileNetV2 and MLP classifier for three custom types: (a) (128-128-128) custom-made top, (b) (192-256-128) custom-made top, and (c) (384-384-384) custom-made top.

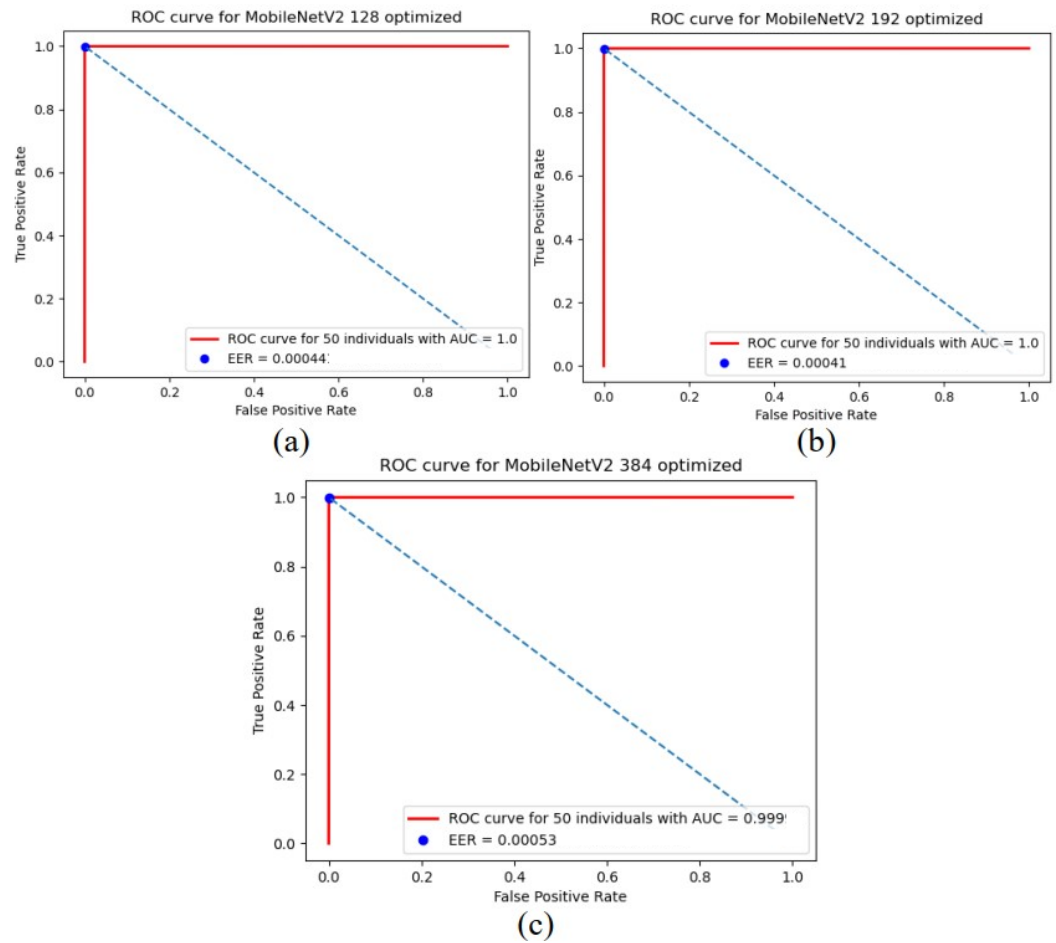


Figure 7. The receiver operating characteristic (ROC) with optimization for a custom-trained CNN using MobileNetV2 with an MLP classifier (a) (128-128-128) custom-made top, (b) (192-256-128) custom-made top, and (c) (384-384-384) custom-made top.

Furthermore, we measure the performance of the custom-trained CNN system using MobileNetV2 and an MLP classifier, but this time we employ the optimization as indicated in Table 2.

Table 2. System performance evaluation: custom-trained CNN utilizing MobileNetV2 and MLP classifier with optimization.

Custom-Made Top	F1-Score	Precision-Score	Recall-Score	Accuracy%
128-128-128	0.998	0.998	0.998	99.913%
192-256-128	0.998	0.998	0.998	99.971%
384-384-384	0.999	0.999	0.999	99.974%

According to Table 2, the findings indicate that the optimization does not impact the accuracy of models (a) and (b). However, model (c) exhibits marginal enhancements in accuracy, which aligns with the intention to reduce inference time without compromising performance. This slight improvement suggests a favorable outcome, maintaining or slightly enhancing performance while focusing on efficiency. The assessment of performance concerning facial recognition tasks is presented through the utilization of the subsequent measures:

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{3}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

where true positives (*TP*) are the number of correctly predicted positive instances, true negatives (*TN*) are the number of correctly predicted negative instances, false positives (*FP*) are the number of negative instances wrongly predicted as positive, and false negatives (*FN*) are the number of positive instances wrongly predicted as negative.

4.3. Experiment (2) FaceNet and SVM Classifier

The FaceNet model is pre-trained by the LFW dataset. Utilizing the SVM classifier, the FaceNet approach is applied to an in-house database that has 24,300 photos taken by 10 different subjects with a high-resolution smartphone camera. Figure 8 demonstrates the ROC curve for the proposed FaceNet with SVM classifier. Different classification methods can be employed [69–71], but the algorithms proposed in the study are highlighted for their successful effect to improve the system performance as shown in this section.

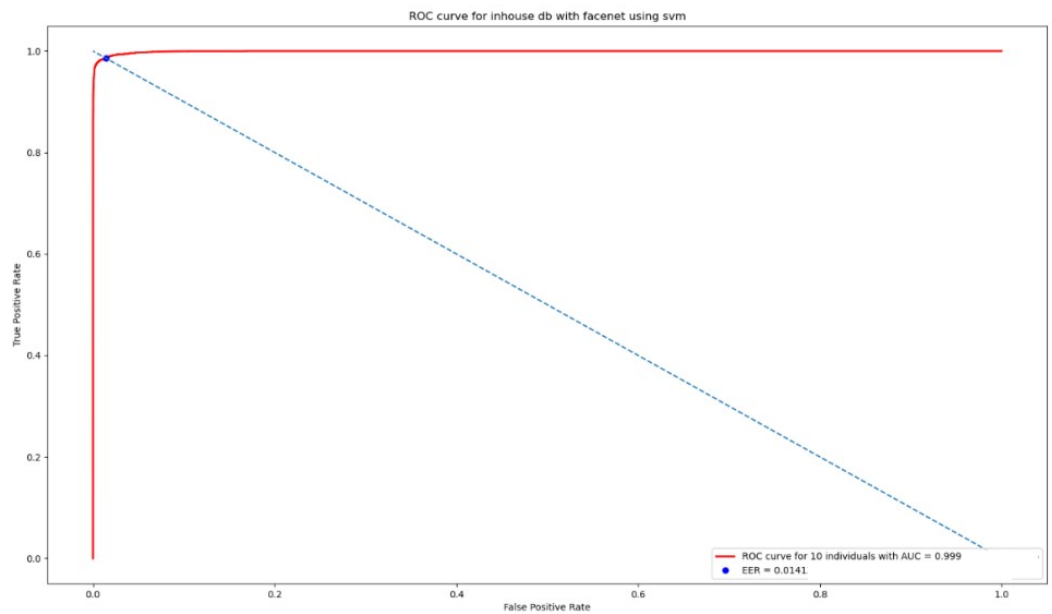


Figure 8. The receiver operating characteristic (ROC) for the FaceNet with the SVM classifier.

The performance evaluation is displayed in terms of the F1-score, precision score, recall score, and accuracy score in Table 3. Table 4 shows the comparison between the proposed work and the state-of-the-art work. The system presented is benchmarked against the current leading methods, as illustrated in Table 4. The results in Table 4 indicate that the proposed system surpasses other existing approaches, achieving an accuracy of 99.976%. In this trend, the SVM and MLP classifiers could provide linear and non-linear separation for the data. Also, the MLP could deploy a multiple layer scheme in which both SVM and MLP contain a high complex kernel.

Table 3. System performance evaluation: FaceNet and SVM classifier.

FaceNet	F1-Score	Precision-Score	Recall-Score	Accuracy %
	0.997	0.997	0.996	99.862%

Both FaceNets (VGG-16 and VGG-19) are pre-trained models as deep-feature extractors, along with MobileNetV2 (small structure). In this trend, we compared these schemes to obtain the highest deep features which give the highest accuracy with a low hardware cost that could be deployed in a low-power device such as Raspberry Pi.

Table 4. Comparison of the proposed approach with other state-of-the-art works.

Reference	Database	Model Used	Accuracy in %
Proposed (IoT-MFaceNet)	In-house + LFW	MobileNetV2 + MLP	99.976%
Proposed (IoT-MFaceNet)	In-house + LFW	FaceNet + SVM	99.862%
[64]	Collection of indoor and outdoor images	CNN implemented on RaspberryPi	97.00%
[65]	ORL face	PCA implemented on RaspberryPi	82.50%
[72]	AT, and T database	PCA and LDA	97.00%
[73]	Collected	faster R-CNN	99.30%
[74]	NA	Yolo and Haar	96.00%
[75]	Collected from the Internet and Kaggle	Deep learning	97.00%
[76]	LFW	Face mesh and Deep Learning	94.23%
[77]	Collected	LBPH	92.00%
[78]	EFLBPTThe Yale Face	LFE Hybrid	95.00%
[79]	Collected	CNN	99.00%

5. Conclusions

The proposed facial recognition system, IoT-MFaceNet, has demonstrated exceptional performance, operating as a closed-set identification tool within the Computer Engineering Department at Mustansiriyah University, Iraq. It effectively controls access to the department's rapporteur room for staff members, achieving an outstanding accuracy rate of 99.976%.

This success can be attributed to the newly curated household image database, which includes images collected from both web and smartphone cameras. These images underwent thorough pre-processing before being processed with advanced deep-learning techniques sourced from pre-trained ImageNet and FaceNet databases. Utilizing various hidden layer structures in MobileNetV2 and FaceNet models facilitated efficient feature extraction and representation.

Key to this achievement was the use of TensorFlow Lite, which compressed the processed data, making it suitable for evaluation on the Raspberry Pi 400 platform. The system employed two primary classifiers, SVM and MLP, for person identification, with the identified individual's image displayed on the rapporteur's mobile device.

Optimization techniques, including TensorFlow Lite and model complexity reduction (termed squeezing), played a crucial role in enhancing system performance. Squeezing streamlined the model structures by reducing layers, neurons, or both, balancing accuracy with speed and memory requirements. Custom-designed setups for MobileNetV2 encompassed various configurations in layer and neuron counts, optimizing performance across different scenarios.

IoT-MFaceNet not only meets the real-time facial recognition requirements but also addresses the challenges associated with resource-constrained IoT devices like the Raspberry Pi 400. By implementing these optimization strategies, the system ensures efficient, accurate, and fast facial recognition, demonstrating the viability of deploying advanced deep-learning models on low-powered edge devices for practical applications.

Author Contributions: Conceptualization, A.S.M., T.G.J., M.T.S.A.-K., J.A.A.-A. and S.D.; methodology, A.S.M., T.G.J. and M.T.S.A.-K.; software, T.G.J., A.S.M. and M.T.S.A.-K.; validation, A.S.M., T.G.J. and M.T.S.A.-K.; formal analysis, A.S.M., T.G.J. and M.T.S.A.-K.; investigation, A.S.M., T.G.J. and M.T.S.A.-K.; resources, A.S.M., M.T.S.A.-K. and J.A.A.-A.; data curation, A.S.M., T.G.J. and J.A.A.-A.; writing—original draft preparation, M.T.S.A.-K. and A.S.M.; writing—review and editing, M.T.S.A.-K., A.S.M. and S.D.; visualization, A.S.M., M.T.S.A.-K. and T.G.J.; supervision, A.S.M.; project administration, A.S.M.; funding acquisition, A.S.M., J.A.A.-A. and S.D. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by Nosh Technologies under Grant nosh/agri-tech-000001.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: The program source code to implement the proposed framework, IoT-MFaceNet, and the data provided in the study could be accessed from <https://github.com/Thoalfeqar-gata/Face-recognition-on-low-powered-devices> (accessed on 6 August 2024).

Acknowledgments: The authors would like to thank the department of Computer Engineering, and the department of Electrical Engineering at Mustansiriyah University, College of Engineering, Baghdad, Iraq for their constant support, and encouragement.

Conflicts of Interest: The authors declare no conflict of interest. In addition, this research was pursued such that part of the proposed methodology could be implemented as a feature in the commercial mobile application named ‘nosh’ for food waste reduction: <https://nosh.tech> (accessed on 6 August 2024).

References

- Huang, P.; Lai, Z.; Gao, G.; Yang, G.; Yang, Z. Adaptive linear discriminant regression classification for face recognition. *Digit. Signal Process.* **2016**, *55*, 78–84. [\[CrossRef\]](#)
- Kutlugün, M.A.; Şirin, Y. Augmenting the training database with the method of gradual similarity ratios in the face recognition systems. *Digit. Signal Process.* **2023**, *135*, 103967. [\[CrossRef\]](#)
- Badr, I.S.; Radwan, A.G.; El-Rabaie, E.S.M.; Said, L.A.; El Banby, G.M.; El-Shafai, W.; Abd El-Samie, F.E. Cancellable face recognition based on fractional-order Lorenz chaotic system and Haar wavelet fusion. *Digit. Signal Process.* **2021**, *116*, 103103. [\[CrossRef\]](#)
- Boukabou, W.R.; Bouridane, A.; Al-Maadeed, S. Enhancing face recognition using directional filter banks. *Digit. Signal Process.* **2013**, *23*, 586–594. [\[CrossRef\]](#)
- Liao, M.; Gu, X. Face recognition based on dictionary learning and subspace learning. *Digit. Signal Process.* **2019**, *90*, 110–124. [\[CrossRef\]](#)
- Taskiran, M.; Kahraman, N.; Erdem, C.E. Face recognition: Past, present and future (a review). *Digit. Signal Process.* **2020**, *106*, 102809. [\[CrossRef\]](#)
- Li, J.; Sang, N.; Gao, C. Face recognition with Riesz binary pattern. *Digit. Signal Process.* **2016**, *51*, 196–201. [\[CrossRef\]](#)
- Büyüктаş, B.; Erdem, Ç.E.; Erdem, T. More learning with less labeling for face recognition. *Digit. Signal Process.* **2023**, *136*, 103915. [\[CrossRef\]](#)
- Zhang, G.; Zou, W.; Zhang, X.; Hu, X.; Zhao, Y. Singular value decomposition based sample diversity and adaptive weighted fusion for face recognition. *Digit. Signal Process.* **2017**, *62*, 150–156. [\[CrossRef\]](#)
- Wang, W.; Qin, J.; Zhang, Y.; Deng, D.; Yu, S.; Zhang, Y.; Liu, Y. TNNL: A novel image dimensionality reduction method for face image recognition. *Digit. Signal Process.* **2021**, *115*, 103082. [\[CrossRef\]](#)
- Sharifani, K.; Amini, M. Machine Learning and Deep Learning: A Review of Methods and Applications. *World Inf. Technol. Eng. J.* **2023**, *10*, 3897–3904.
- Alzu'bi, A.; Albalas, F.; Al-Hadhrami, T.; Younis, L.B.; Bashayreh, A. Masked face recognition using deep learning: A review. *Electronics* **2021**, *10*, 2666. [\[CrossRef\]](#)
- Qinjun, L.; Tianwei, C.; Yan, Z.; Yuying, W. Facial Recognition Technology: A Comprehensive Overview. *Acad. J. Comput. Inf. Sci.* **2023**, *6*, 15–26.
- Awajan, A. A novel deep learning-based intrusion detection system for IOT networks. *Computers* **2023**, *12*, 34. [\[CrossRef\]](#)

15. Islam, M.R.; Kabir, M.M.; Mridha, M.F.; Alfarhood, S.; Safran, M.; Che, D. Deep Learning-Based IoT System for Remote Monitoring and Early Detection of Health Issues in Real-Time. *Sensors* **2023**, *23*, 5204. [[CrossRef](#)]
16. Javed, A.; Awais, M.; Shoaib, M.; Khurshid, K.S.; Othman, M. Machine learning and deep learning approaches in IoT. *PeerJ Comput. Sci.* **2023**, *9*, e1204. [[CrossRef](#)] [[PubMed](#)]
17. Zakariah, M.; Almazyad, A.S. Anomaly Detection for IOT Systems Using Active Learning. *Appl. Sci.* **2023**, *13*, 12029. [[CrossRef](#)]
18. Ma, X.; Yao, T.; Hu, M.; Dong, Y.; Liu, W.; Wang, F.; Liu, J. A survey on deep learning empowered IoT applications. *IEEE Access* **2019**, *7*, 181721–181732. [[CrossRef](#)]
19. Dey, S. Novel DVFS Methodologies for Power-Efficient Mobile MPSoC. Ph.D. Thesis, University of Essex, Colchester, UK, 2023.
20. Isuwa, S.; Dey, S.; Singh, A.K.; McDonald-Maier, K. Teem: Online thermal-and energy-efficiency management on cpu-gpu mpsoCs. In Proceedings of the 2019 Design, Automation & Test in Europe Conference & Exhibition (DATE), Florence, Italy, 25–29 March 2019; pp. 438–443.
21. Singh, A.K.; Dey, S.; McDonald-Maier, K.; Basireddy, K.R.; Merrett, G.V.; Al-Hashimi, B.M. Dynamic energy and thermal management of multi-core mobile platforms: A survey. *IEEE Des. Test* **2020**, *37*, 25–33. [[CrossRef](#)]
22. Dey, S.; Singh, A.K.; Wang, X.; McDonald-Maier, K. User interaction aware reinforcement learning for power and thermal efficiency of CPU-GPU mobile MPSoCs. In Proceedings of the 2020 Design, Automation & Test in Europe Conference & Exhibition (DATE), Grenoble, France, 9–13 March 2020; pp. 1728–1733.
23. Dey, S.; Isuwa, S.; Saha, S.; Singh, A.K.; McDonald-Maier, K. CPU-GPU-memory DVFS for power-efficient MPSoC in mobile cyber physical systems. *Future Internet* **2022**, *14*, 91. [[CrossRef](#)]
24. Dey, S.; Saha, S.; Singh, A.; McDonald-Maier, K. Fruitvegenn: Power-and memory-efficient classification of fruits & vegetables using cnn in mobile mpsoC. In Proceedings of the 2020 IEEE 17th India Council International Conference (INDICON), New Delhi, India, 10–13 December 2020; pp. 1–7.
25. Louis, M.S.; Azad, Z.; Delshadtehrani, L.; Gupta, S.; Warden, P.; Reddi, V.J.; Joshi, A. Towards deep learning using tensorflow lite on risc-v. In Proceedings of the Third Workshop on Computer Architecture Research with RISC-V (CARRV), Phoenix, AZ, USA, 22 June 2019; Volume 1, p. 6.
26. Warden, P.; Situnayake, D. *Tinyml: Machine Learning with Tensorflow Lite on Arduino and Ultra-Low-Power Microcontrollers*; O'Reilly Media: Sebastopol, CA, USA, 2019.
27. Mohammad, A.S.; Rattani, A.; Derakhshani, R. Comparison of squeezed convolutional neural network models for eyeglasses detection in mobile environment. *J. Comput. Sci. Coll.* **2018**, *33*, 136–144.
28. Bolhasani, H.; Mohseni, M.; Rahmani, A.M. Deep learning applications for IoT in health care: A systematic review. *Inform. Med. Unlocked* **2021**, *23*, 100550. [[CrossRef](#)]
29. Mohammad, A.S.; Al-Kaltakchi, M.T.; Alshehabi Al-Ani, J.; Chambers, J.A. Comprehensive Evaluations of Student Performance Estimation via Machine Learning. *Mathematics* **2023**, *11*, 3153. [[CrossRef](#)]
30. Al-Kaltakchi, M.T.; Mohammad, A.S.; Woo, W.L. Ensemble System of Deep Neural Networks for Single-Channel Audio Separation. *Information* **2023**, *14*, 352. [[CrossRef](#)]
31. Al-Nima, R.R.O.; Al-Kaltakchi, M.T.; Han, T.; Woo, W.L. Road tracking enhancements for self-driving cars applications. In Proceedings of the International Conference on Innovations in Science, Hybrid Materials, and Vibration Analysis: Icishva 2022, Pune, India, 16–17 July 2022; AIP Conference Proceedings; AIP Publishing: Melville, NY, USA, 2023; Volume 2839.
32. Jacob, I.J.; Darney, P.E. Design of deep learning algorithm for IoT application by image based recognition. *J. ISMAC* **2021**, *3*, 276–290. [[CrossRef](#)]
33. Rghioui, A.; Lloret, J.; Oumnad, A. Big data classification and internet of things in healthcare. *Int. J. E-Health Med Commun. (IJEHMC)* **2020**, *11*, 20–37. [[CrossRef](#)]
34. Balaji, P.; Sri Revathi, B.; Gobinathan, P.; Shamsudheen, S.; Vaiyapuri, T. Optimal IoT Based Improved Deep Learning Model for Medical Image Classification. *Comput. Mater. Contin.* **2022**, *73*, 2275–2291. [[CrossRef](#)]
35. Sandhiyasa, I.M.S.; Waas, D.V. Real Time Face Recognition for Mobile Application Based on Mobilenetv2. *J. Multidisiplin Madani* **2023**, *3*, 1855–1864. [[CrossRef](#)]
36. Mohammad, A.S. Multi-Modal Ocular Recognition in Presence of Occlusion in Mobile Devices. Ph.D. dissertation, University of Missouri-Kansas City, Kansas City, MO, USA, 2018.
37. Mohammad, A.S.; Al-Ani, J.A. Convolutional neural network for ethnicity classification using ocular region in mobile environment. In Proceedings of the 2018 10th Computer Science and Electronic Engineering (CEECE), Colchester, UK, 19–21 September 2018; pp. 293–298.
38. Al-Rammahi, A.H.I. Face mask recognition system using MobileNetV2 with optimization function. *Appl. Artif. Intell.* **2022**, *36*, 2145638. [[CrossRef](#)]
39. Kumar, B.A.; Bansal, M. Face mask detection on photo and real-time video images using Caffe-MobileNetV2 transfer learning. *Appl. Sci.* **2023**, *13*, 935. [[CrossRef](#)]
40. Sukkar, V.; Ercelebi, E. A Real-time Face Recognition Based on MobileNetV2 Model. *Balk. J. Electr. Comput. Eng. (BAJECE)* **2023**, *1*, 31–39.
41. Gunawan, I.; Liman, M.A.; Ryan, G.; Purnomo, F. Face Mask Detection for COVID-19 Prevention using Computer Vision. *Procedia Comput. Sci.* **2023**, *227*, 1143–1152. [[CrossRef](#)]

42. Baran, K. Smartphone thermal imaging for stressed people classification using CNN+ MobileNetV2. *Procedia Comput. Sci.* **2023**, *225*, 2507–2515. [[CrossRef](#)]
43. Durga, G.L.; Potluri, H.; Vinnakota, A.; Prativada, N.P.; Yelavarti, K. Face Mask Detection using MobileNetV2. In Proceedings of the 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS), Coimbatore, India, 23–25 February 2022; pp. 933–940.
44. Adhinata, F.D.; Tanjung, N.A.F.; Widayat, W.; Pasfica, G.R.; Satura, F.R. Comparative study of VGG16 and MobileNetv2 for masked face recognition. *J. Ilm. Tek. Elektro Komput. Dan Inform. (JITEKI)* **2021**, *7*, 230–237.
45. Almghraby, M.; Elnady, A.O. Face mask detection in real-time using MobileNetv2. *Int. J. Eng. Adv. Technol.* **2021**, *10*, 104–108. [[CrossRef](#)]
46. Hussain, D.; Ismail, M.; Hussain, I.; Alroobaea, R.; Hussain, S.; Ullah, S.S. Face mask detection using deep convolutional neural network and MobileNetV2-based transfer learning. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 1–10. [[CrossRef](#)]
47. Gupta, V.; Rajput, R. Face mask detection using MTCNN and MobileNetV2. *Int. Res. J. Eng. Technol. (IRJET)* **2021**, *8*, 309–312.
48. Dang, T.V. Smart attendance system based on improved facial recognition. *J. Robot. Control (JRC)* **2023**, *4*, 46–53. [[CrossRef](#)]
49. Gulzar, Y. Fruit image classification model based on MobileNetV2 with deep transfer learning technique. *Sustainability* **2023**, *15*, 1906. [[CrossRef](#)]
50. Shivaprasad, S.; Sai, M.D.; Vignasahithi, U.; Keerthi, G.; Rishi, S.; Jayanth, P. Real time CNN based detection of face mask using mobilenetv2 to prevent Covid-19. *Ann. Rom. Soc. Cell Biol.* **2021**, *25*, 12958–12969.
51. Nguyen, H. Fast object detection framework based on mobilenetv2 architecture and enhanced feature pyramid. *J. Theor. Appl. Inf. Technol.* **2020**, *98*, 812–824.
52. Kocacinar, B.; Tas, B.; Akbulut, F.P.; Catal, C.; Mishra, D. A real-time cnn-based lightweight mobile masked face recognition system. *IEEE Access* **2022**, *10*, 63496–63507. [[CrossRef](#)]
53. Saxen, F.; Werner, P.; Handrich, S.; Othman, E.; Dinges, L.; Al-Hamadi, A. Face attribute detection with mobilenetv2 and nasnet-mobile. In Proceedings of the 2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA), Dubrovnik, Croatia, 23–25 September 2019; pp. 176–180.
54. Nan, Y.; Ju, J.; Hua, Q.; Zhang, H.; Wang, B. A-MobileNet: An approach of facial expression recognition. *Alex. Eng. J.* **2022**, *61*, 4435–4444. [[CrossRef](#)]
55. Alharbi, B.; Alshanbari, H.S. Face-voice based multimodal biometric authentication system via FaceNet and GMM. *PeerJ Comput. Sci.* **2023**, *9*, e1468. [[CrossRef](#)]
56. Dang, T.V.; Tran, L.H. A Secured, Multilevel Face Recognition based on Head Pose Estimation, MTCNN and FaceNet. *J. Robot. Control (JRC) ISSN* **2023**, *2715*, 2. [[CrossRef](#)]
57. Rosid, J.; Sakti, D.M.; Murti, W.S.; Kurniasari, A. Face recognition dengan metode Haar Cascade dan Facenet. *Indones. J. Data Sci.* **2022**, *3*, 30–34.
58. Deng, Z.Y.; Chiang, H.H.; Kang, L.W.; Li, H.C. A lightweight deep learning model for real-time face recognition. *IET Image Process.* **2023**, *17*, 3869–3883. [[CrossRef](#)]
59. Nagrath, P.; Jain, R.; Madan, A.; Arora, R.; Kataria, P.; Hemanth, J. SSDMNv2: A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2. *Sustain. Cities Soc.* **2021**, *66*, 102692. [[CrossRef](#)]
60. Salih, T.A.; Gh, M.B. A novel Face Recognition System based on Jetson Nano developer kit. In *IOP Conference Series: Materials Science and Engineering*; IOP Publishing: Bristol, UK, 2020; Volume 928, p. 032051.
61. Nagpal, G.S.; Singh, G.; Singh, J.; Yadav, N. Facial detection and recognition using OPENCV on Raspberry Pi Zero. In Proceedings of the 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Greater Noida, India, 12–13 October 2018; pp. 945–950.
62. Gupta, I.; Patil, V.; Kadam, C.; Dumbre, S. Face detection and recognition using Raspberry Pi. In Proceedings of the 2016 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE), Pune, India, 19–21 December 2016; pp. 83–86.
63. Dolas, P.; Ghogare, P.; Kshirsagar, A.; Khadke, V.; Bokefode, S. Face Detection and Recognition Using Raspberry Pi. In *Techno-Societal 2020: Proceedings of the 3rd International Conference on Advanced Technologies for Societal Applications*; Springer: Berlin/Heidelberg, Germany, 2021; Volume 1, pp. 351–354.
64. Dürr, O.; Pauchard, Y.; Browarnik, D.; Axthelm, R.; Loeser, M. Deep Learning on a Raspberry Pi for Real Time Face Recognition. In *Eurographics (Posters)*; Winterthur, Switzerland; 2015 ; pp. 11–12.
65. Mohammed, I.M.; Al-Dabagh, M.Z.N.; Ahmad, M.I.; Isa, M.N.M. Face recognition using PCA implemented on raspberry Pi. In *Proceedings of the 11th National Technical Seminar on Unmanned System Technology 2019: NUSYS'19*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 873–889.
66. Dhobale, M.R.; Biradar, R.Y.; Pawar, R.R.; Awatade, S.A. Smart Home Security System using Iot, Face Recognition and Raspberry Pi. *Int. J. Comput. Appl.* **2020**, *176*, 45–47.
67. Huang, G.B.; Mattar, M.; Berg, T.; Learned-Miller, E. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In Proceedings of the Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition, Marseille, France, 16 September 2008.

68. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
69. Al-Kaltakchi, M.T.S.; Al-Sumaidae, S.A.M.; Al-Nima, R.R.O. Classifications of signatures by radial basis neural network. *Bull. Electr. Eng. Inform.* **2022**, *11*, 3294–3300. [[CrossRef](#)]
70. Al-Kaltakchi, M.T.S. Robust Text Independent Closed Set Speaker Identification Systems and Their Evaluation. Ph.D. Thesis, Newcastle University, Callaghan, Australia, 2018.
71. Al-Kaltakchi, M.T.; Woo, W.L.; Dlay, S.S.; Chambers, J.A. Multi-dimensional i-vector closed set speaker identification based on an extreme learning machine with and without fusion technologies. In Proceedings of the 2017 Intelligent Systems Conference (IntelliSys), London, UK, 7–8 September 2017; pp. 1141–1146.
72. Borkar, N.R.; Kuwelkar, S. Real-time implementation of face recognition system. In Proceedings of the 2017 International Conference on Computing Methodologies and Communication (ICCMC), Erode, India 18–19 July 2017; pp. 249–255.
73. Rajeshkumar, G.; Braveen, M.; Venkatesh, R.; Shermila, P.J.; Prabu, B.G.; Veerasamy, B.; Bharathi, B.; Jeyam, A. Smart office automation via faster R-CNN based face recognition and internet of things. *Meas. Sens.* **2023**, *27*, 100719. [[CrossRef](#)]
74. Nadafa, R.A.; Hatturea, S.; Bonala, V.M.; Naikb, S.P. Home security against human intrusion using Raspberry Pi. *Procedia Comput. Sci.* **2020**, *167*, 1811–1820. [[CrossRef](#)]
75. Mukto, M.M.; Hasan, M.; Al Mahmud, M.M.; Haque, I.; Ahmed, M.A.; Jabid, T.; Ali, M.S.; Rashid, M.R.A.; Islam, M.M.; Islam, M. Design of a Real-Time Crime Monitoring System Using Deep Learning Techniques. *Intell. Syst. Appl.* **2023**, *21*, 200311. [[CrossRef](#)]
76. Hangaragi, S.; Singh, T.; Neelima, N. Face detection and Recognition using Face Mesh and deep neural network. *Procedia Comput. Sci.* **2023**, *218*, 741–749. [[CrossRef](#)]
77. Budiman, A.; Fabian; Yaputera, R.A.; Achmad, S.; Kurniawan, A. Student attendance with face recognition (LBPH or CNN): Systematic literature review. *Procedia Comput. Sci.* **2023**, *216*, 31–38. [[CrossRef](#)]
78. Chaudhry, A.; Elgazzar, H. Design and implementation of a hybrid face recognition technique. In Proceedings of the 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Vegas, NV, USA, 7–9 January 2019; pp. 0384–0391.
79. Pratama, Y.; Ginting, L.M.; Nainggolan, E.H.L.; Rismanda, A.E. Face recognition for presence system by using residual networks-50 architecture. *Int. J. Electr. Comput. Eng.* **2021**, *11*, 5488. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.