scientific reports

Check for updates

OPEN

Real-time facial recognition via multitask learning on raspberry Pi

Abdulatif Ahmed Ali Aboluhom[1]✉ & Ismet Kandilli[2]

This paper investigates the feasibility of multi-task learning (MTL) for facial recognition on the Raspberry Pi, a low-cost single-board computer, demonstrating its ability to perform complex deep learning tasks in real time. Using MobileNet, MobileNetV2, and InceptionV3 as base models, we trained MTL models on a custom database derived from the VGGFace2 dataset, focusing on three tasks: person identification, age estimation, and ethnicity prediction. MobileNet achieved the highest accuracy, with 99% in person identification, 99.3% in age estimation, and 99.5% in ethnicity prediction. Compared to previous studies, which primarily relied on high-end hardware for MTL in facial recognition, this work uniquely demonstrates the successful deployment of efficient MTL models on resource-constrained devices like the Raspberry Pi. This advancement significantly reduces computational load and energy consumption while maintaining high accuracy, making facial recognition systems more accessible and practical for real-world applications such as security, personalized customer experiences, and demographic analytics. This study opens new avenues for innovation in resource-efficient deep learning systems.

Keywords Multi-task learning, Raspberry Pi, Deep learning, Face recognition, Real-time

Facial recognition and identification of individuals are crucial research topics. Various approaches have been developed to address this challenge. Face recognition is a significant aspect of image recognition research. Humans constantly recognize visual descriptions, identify visual data with their eyes, and process this information with their brains. In contrast, computers evaluate images or videos as matrices of pixels. The aim is to determine which group of pixels represents which concept or object in the dataset. Essentially, it is an image classification task. In face recognition, it is essential to accurately determine the identity of individuals from facial images. The identification process includes face detection, face positioning, and face verification. In the face detection method, the system detects the coordinates of faces in an image or video. This process requires a detailed scan of the entire image, resulting in an output that encloses the human face within a square or rectangle. The location of facial features defines the position of the human face within the coordinate system obtained during face detection. Deep learning techniques are often used in face positioning methods. The face recognition process consists of two stages: identifying faces and verifying these identified faces by comparing them with a database. Various methods are used in face detection and recognition studies. The facial recognition process can be performed using an appearance-based approach or a feature-based approach focusing on geometric features Selvi et al.[1]. Although significant progress has been made in facial recognition studies, further research is required for practical applications. Recent studies have begun tackling major challenges in facial recognition accuracy, especially those caused by low quality data captured by surveillance and similar camera systems[2,3].

In face recognition studies, this technology is predominantly used in access control, finance and security, retail, logistics, transportation, education, real estate, and smartphone and network information security, among other fields[4–6]. This technology serves as an integral component in public safety and surveillance systems, enabling real-time monitoring of airports and public spaces to track individuals of interest[7–9]. The banking and financial sector utilizes facial recognition technology in ATMs as well as remote customer verification processes, enhancing the security of financial transactions[10,11]. The advent of artificial intelligence has increased the demand for more accurate, flexible, and faster recognition technologies. New deep learning methods have achieved high accuracy in identifying individuals using digital imaging. Facial recognition using software algorithms is crucial for research because these algorithms enable the recognition of individuals through digital images and the processing of large amounts of digital data. Deep neural networks have enabled the recognition of faces in digital photography through experimental applications. Raspberry Pi-based solutions are an example of these applications by Bajrami et al.[12] and Human Face Recognition for Video Surveillance and Automation using Raspberry Pi is introduced by Mustakim et al.[13]. In recent years, Convolutional Neural Networks (CNNs) and deep neural networks have shown excellent performance in face recognition tasks. Face recognition systems

[1]Engineering Faculty, Electronics Department, Ibb University, Ibb, Yemen. [2]Electronics and Automation Department, Kocaeli University, Izmit, Turkey. ✉email: abdullatif1995.11@gmail.com

can be developed using the CNN model, which detects the human face with a camera and accurately identifies the individual using a CNN model by Wang & Guo[14]. The evolution of deep learning, driven by progress in neural network architectures such as Convolutional Neural Networks (CNNs), has revolutionized facial recognition systems, enhancing their accuracy and reliability to unprecedented levels[15–19]. In recent studies, multi-task learning has been used for face recognition, decomposing facial features into person- and age-related components Huang, Z et al.[20]. The approach by Foggia et al.[21] employed a multitask CNN to simultaneously recognize gender, ethnicity, age, and emotion from facial images, addressing the need for multitask recognition systems. While Foggia et al.[21] and Wang et al.[22] have explored multitask recognition systems, their approaches are not optimized for low-power, resource-constrained environments, and platforms such as Raspberry Pi.

In parallel, other studies have explored related challenges in facial analysis and image recognition. For example, Kumar et al.[23] utilized the Caffe-MobileNetV2 model for feature extraction and classification of masked and unmasked faces, focusing on visible areas like the eyes and forehead. Similarly, Dahri et al.[24] proposed a CRNN model with BiGRU and transfer learning using VGG16 for image captioning, while Dahri et al.[25] developed a specialized deep CNN model to enhance indoor scene recognition. In the domain of Deepfake detection, Javed et al.[26] proposed a hybrid model combining MesoNet4 and ResNet101 for real-time applications. For medical imaging, Shazia et al.[27] introduced a hybrid model combining Inception V3 and VGG16 for automated diabetic retinopathy detection. Additionally, Dahri et al.[28] proposed a YOLOv5-based approach for face mask detection, and Kumar et al.[29] introduced the CMNV2 model, combining CAFFE and a modified MobileNetV2, for masked face age and gender identification. Alhanaee et al.[30] introduced facial recognition attendance system based on deep learning CNNs. These studies highlight the versatility of deep learning models in addressing diverse challenges.

According to Wang et al.[22], multitask networks are implemented using two main approaches: soft parameter sharing and hard parameter sharing. Hard parameter sharing uses the same set of weights for the initial layers across all task-specific networks. Consequently, these shared layers compute outputs only once, which are then reused for each task. On the other hand, soft parameter sharing involves constructing individual networks for each task. However, during the training phase, constraints are applied to the initial layers of these networks to ensure that they converge toward similar values, often through specialized additional terms in the loss function, as noted by Crawshaw[31].

Great success has been achieved in deep learning and face recognition. However, the small amount of labeled data poses a problem in the face recognition process. Despite significant advances in the field, this limitation makes transfer learning particularly useful when dealing with large amounts of labeled data. In the traditional learning approach, independent models are developed on the basis of specific datasets for various tasks. Using deep learning techniques, one of the biggest challenges of face recognition—enabling the system to detect and recognize faces in real-world conditions—has been addressed. Such conditions include varying poses, lighting, aging, masking, facial expressions, plastic surgery, and low-resolution Oloyede et al.[32].

This study aims to address these limitations by optimizing multitask facial recognition for low-power, resource-constrained environments using the Raspberry Pi 4, enabling efficient, real-time performance for person, age, and ethnicity estimation. The Raspberry Pi 4, with its quad-core processor, 4GB RAM, and GPU support, offers a unique platform for deploying deep learning models in resource-constrained settings. However, its limited computational power necessitates innovative optimization strategies, such as transfer learning, and lightweight architectures like MobileNet and MobileNetV2, to balance real-time performance and accuracy. By leveraging pre-trained models and transfer learning, this study reduces training time and computational requirements, making it feasible to deploy advanced multitask recognition systems on the Raspberry Pi 4.

The significance of this research lies in its ability to overcome the technical and logistical challenges of multitask facial recognition on resource-constrained devices. Unlike prior studies, which primarily focused on high-performance computing environments, this work explores how the Raspberry Pi 4 can handle multitask learning effectively, offering advantages such as reduced energy consumption, improved real-time performance, and scalability for real-world applications. Furthermore, this study critically evaluates the limitations of existing approaches and proposes a methodology that addresses these challenges, such as balancing computational constraints with real-time performance and leveraging natural correlations between tasks (e.g., person recognition, age, and ethnicity estimation) to improve overall accuracy.

This research advances the field of facial recognition by demonstrating how multitask learning can be optimized for resource-constrained platforms like the Raspberry Pi 4. By evaluating the underlying principles, methodologies, datasets, and evaluation metrics for each task, key challenges and opportunities for future research are identified.

Methods

The facial recognition system we developed is designed to be robust, fast, and lightweight. This system is ideal for integration into devices such as Raspberry Pi 4. The system consists of four primary stages: face detection, alignment, feature extraction, and classification.

In face detection and alignment, it effectively performs precise localization of facial areas using the multitask cascaded convolutional networks (MTCNN) algorithm. To adapt to the hardware limitations of the Raspberry Pi 4, deep CNN models such as MobileNetV2, MobileNet, and InceptionV3 selected for feature extraction were optimized, thereby establishing a balance between computational efficiency and representation ability.

The extracted features were classified using multitask learning model to classify multiple tasks, using softmax layers to recognize individuals, estimate their age, and detect ethnicity. In this study, development and implementation process, our focus is on resource utilization and optimization to ensure real-time performance on Raspberry Pi 4. This offers a practical solution for face recognition using various models.

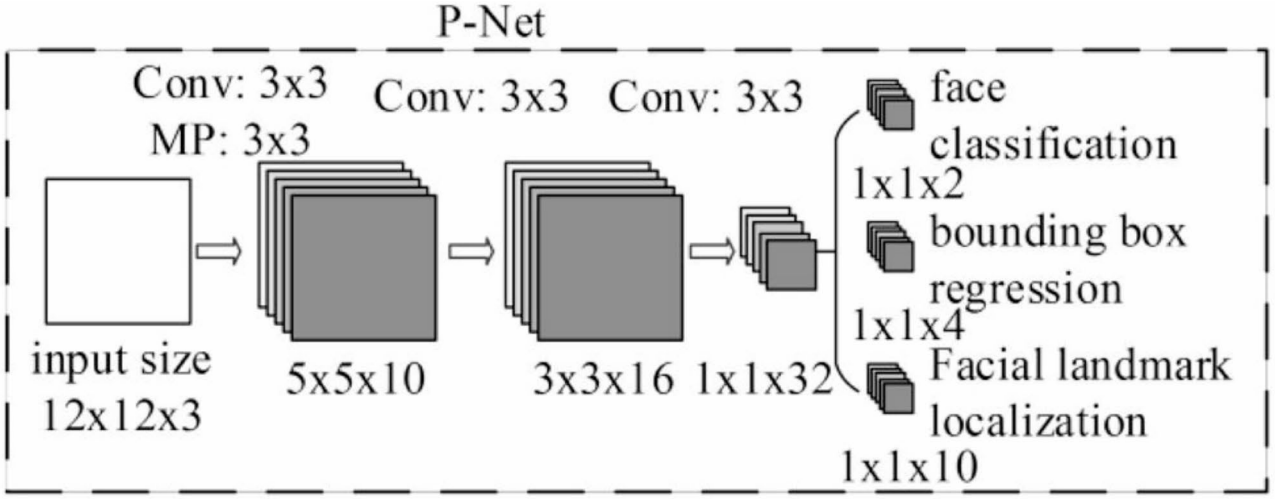

Fig. 1. P-Net[33].

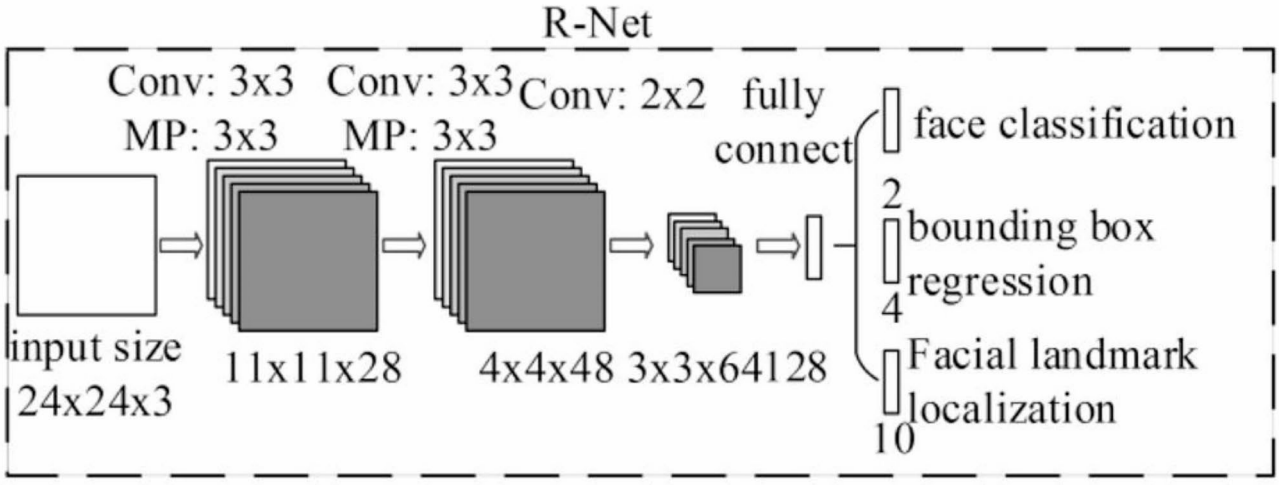

Fig. 2. R-Net[33].

Face detection

The MTCNN is the procedure of face detection using the CNN algorithm and then aligning the detected faces with facial landmarks. The purpose of face detection is to recognize and locate human faces in an image. Facial alignment involves accurately defining certain facial features, such as the eyes, nose, and mouth. MTCNN is a powerful algorithm designed for face detection and alignment and is known for its fast and accurate performance. It excels at handling challenges that can hinder other situations, such as changing image conditions and facial changes.

MTCNN, a face detection technique developed by Zhang et al.[33], employs a CNN-based approach. It comprises three networks: P-Net, R-Net, and O-Net. P-Net is fully convolutional, whereas R-Net and O-Net are conventional CNNs. MTCNN accommodates input images of varying sizes. Usually, when an image is encountered, it is resized to various scales to create an image pyramid for the next three-stage cascade frame. MTCNN is widely used because of its outstanding accuracy and effectiveness in face detection. It works through three different stages:

P-Net: P-Net uses convolutional networks to generate potential face regions by generating candidate bounding boxes. It scans the image at various scales to identify these regions. Figure 1 shows the proposal network P-Net.

R-Net: R-Net refines the candidate bounding boxes obtained in the first stage, eliminates false positives, and locates faces precisely. This stage further increases the accuracy of face detection. Figure 2 shows the refine network R-Net.

O-Net: O-Net uses O-Net as the final stage where facial features are fine-tuned and facial landmarks are detected within bounding boxes. In addition to providing the coordinates of facial landmarks, O-Net provides additional information about the detected faces. Figure 3 shows the Output network (O-Net).

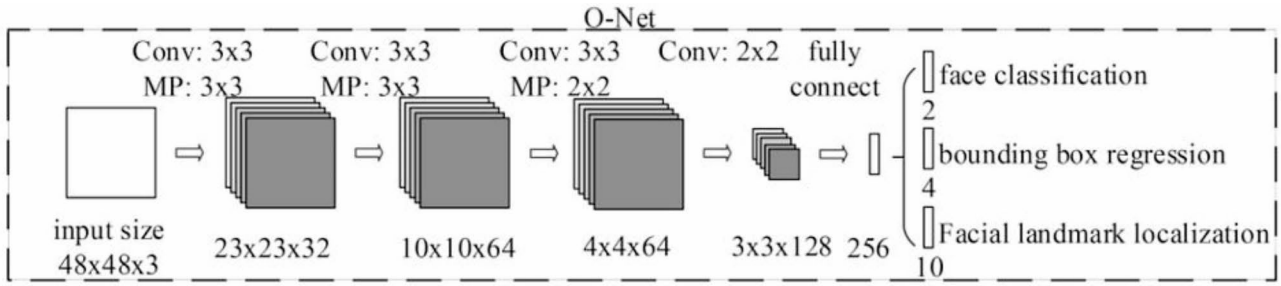

Fig. 3. O-Net[33].

To train the networks, three tasks must be completed: face classification, facial landmark localization, and bounding box regression. The loss function employed for face classification is the cross-entropy loss, as depicted in Eq. (1); where p_i represents the network's probability for a given face and $y_i^{det} \in \{0,1\}$ is the actual label.

$$L_i^{det} = -(y_i^{det} \log(p_i) + (1 - y_i^{det})(1 - \log(p_i))) \tag{1}$$

During the training phase, it is crucial to minimize the distance between the bounding box and the nearest ground truth, especially when the bounding box encompasses the face in the image. Euclidean loss is used for the bounding box, as illustrated in Eq. (2); where $\hat{y}_i^{box}$ represents the network-derived bounding box results and y_i^{box} denotes the proximate ground truth.

$$L_i^{box} = \left\| \hat{y}_i^{box} - y_i^{box} \right\|_2^2 \tag{2}$$

The size of the bounding box is four and includes the coordinates of the upper left corner, width, and height. Likewise, for facial point regression, the Euclidean loss is applied, illustrated in Eq. (3); $\hat{y}_i^{landmark}$ represents the point coordinate on the face from the mesh, and $y_i^{landmark}$ denotes the ground truth coordinate.

$$L_i^{landmark} = \left\| \hat{y}_i^{landmark} - y_i^{landmark} \right\|_2^2 \tag{3}$$

Face alignment

Once faces are detected using the MTCNN algorithm, the next step is to align the identified faces with specific facial landmarks. These landmarks, which include landmarks on the face such as the eye corners, nose tip, and mouth corners, are essential to ensure consistent facial orientation and improve subsequent tasks such as facial recognition. The alignment procedure typically involves several steps. First, the MTCNN algorithm provides coordinates representing facial landmarks (such as eyes, nose, and mouth) within each detected face bounding box. Geometric transformations such as translation, rotation, and scaling are then applied based on these landmarks to position consistent facial features across various faces. The normalization process ensures consistency across bookmarks. Finally, the aligned face images are based on calculated transformations to correct distortions introduced during alignment, thus ensuring consistent relationships between facial landmarks. Face detection and face alignment techniques make our methodology uniquely suited for deploying multitask facial recognition systems in low-power, real-world applications.

Feature extraction

CNN models widely used in facial recognition are InceptionV3, MobileNet, and MobileNetV2. These designs are widely recognized for their effectiveness, precision, and adaptability in various applications such as feature extraction and facial recognition. They are suitable for devices with low computing power such as Raspberry Pi. For these reasons, CNN models such as InceptionV3, MobileNetV2, and MobileNet are preferred, especially in face recognition fields and for classifying images according to age, person and ethnicity. Our method is based on efficiently extracting features from deep and complex architectures of models such as MobileNetV2, InceptionV3, and MobileNet. In practical applications, we use these models primarily as fixed feature extractors, using their upper layers and convolutional bases. This strategy allows us to take advantage of the complex feature representations learned by these models and adapt to new challenges with minimal additional training.

Table 1 summarizes the training setup for three different convolutional neural network models: MobileNet, MobileNetV2, and InceptionV3. Each model is configured with a dense top layer of 128 units. The number of batches for training was set to 32, the same for all models; This means that 32 images will be processed before the model's internal parameters are updated. Finally, the input size was set to 128×128 pixels with three color channels (RGB), which is the expected size of the input images for all models. The choice of hyperparameters was carefully selected to balance performance and computational efficiency on the Raspberry Pi 4. An input size of 128×128 was chosen to reduce memory usage while retaining sufficient detail for accurate face detection and alignment. A batch size of 32 was selected to optimize GPU utilization and training stability without exceeding the Raspberry Pi 4's memory constraints. The 100 epochs were chosen to ensure model convergence, as preliminary experiments showed that this number provided a good trade-off between training time and

The based Model	Top Layer	Batch size	Input size
MobileNetV2	Dense (128)	32	(128,128,3)
InceptionV3	Dense (128)	32	(128,128,3)
MobileNet	Dense (128)	32	(128,128,3)

Table 1. Training setup of the CNN models.

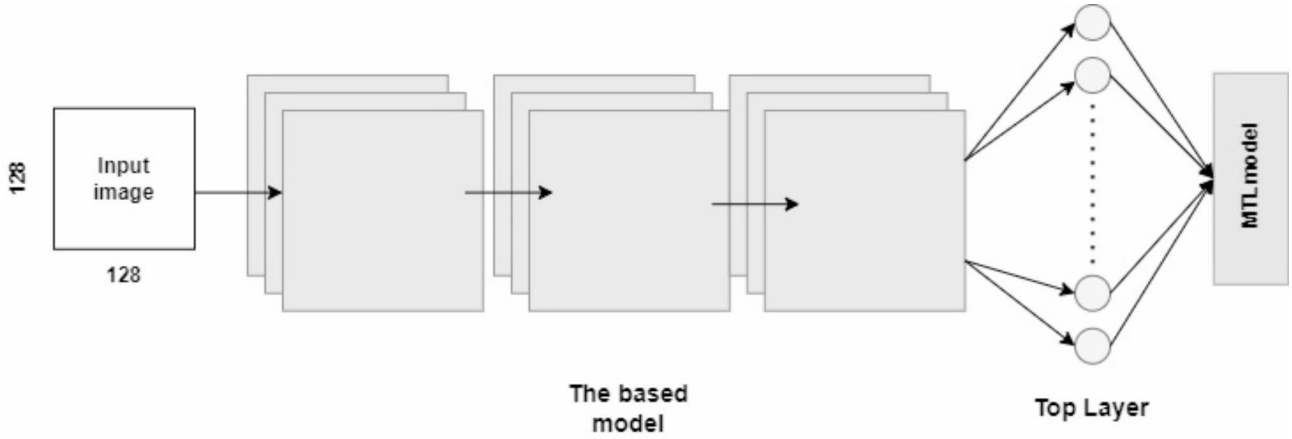

Fig. 4. Facial feature extraction model architecture.

accuracy. These values were validated through iterative testing to achieve optimal results within the platform's limitations. This setup is used for pre-trained models on the dataset and uses a dense layer to adapt to the specific output requirements of our tasks. Figure 4 shows the Facial feature extraction model architecture used in our study.

The feature extraction process is a critical component of our methodology, enabling the system to identify and encode key facial attributes. As illustrated in Fig. 4, the architecture consists of multiple layers designed to extract hierarchical features from input images. The input image, resized to 128×128 pixels, is passed through a series of convolutional layers, each followed by pooling operations to reduce dimensionality while preserving essential features.

The top layer of the base model, as shown in Fig. 4, is responsible for capturing high-level features such as facial contours and landmarks. These features are then fed into subsequent layers for further processing and classification. By referencing Fig. 4, readers can better visualize the flow of data through the network and understand how each layer contributes to the overall feature extraction process.

Proposed MTL models

In multi-task learning (MTL), hard parameter sharing and soft parameter sharing differ significantly in architecture, scalability, performance, and applicability. Hard parameter sharing divides parameters into shared (common across tasks) and task-specific (unique to each task) components, typically using a shared encoder followed by task-specific heads, as seen in UberNet Kokkinos[34]. This approach is highly scalable, efficient to train, and reduces overfitting but risks negative transfer if tasks are not closely related. In contrast, soft parameter sharing assigns each task its own parameters and uses feature-sharing mechanisms, such as linear combinations of activations, as in Cross-Stitch Networks Misra et al.[35]. While more flexible and robust to task dissimilarity, soft parameter sharing is less scalable, harder to train, and more prone to overfitting. Hard parameter sharing is best suited for closely related tasks, such as vision tasks, while soft parameter sharing is more effective for diverse or loosely related tasks. The choice between the two depends on task relationships, problem domain, and computational resources.

It uses a multi-task learning model to classify people with tasks to determine their age and ethnicity. Our method uses hard parameter sharing to efficiently use computational resources and reveal commonalities between different tasks. This method primarily involves sharing the sublayers of a network among all tasks and having several task-specific output layers. Following the feature extraction phase, where models such as MobileNetV2, InceptionV3, and MobileNet are used to extract deep and complex features from data, these extracted features are used as shared layers in our MTL model. These shared layers provide a common and robust set of features, serving as a unified feature extraction base that benefits multiple learning tasks.

The architecture of our MTL models starts with the shared convolutional base derived from one of the three selected CNN models: MobileNetV2, InceptionV3, or MobileNet. On top of this common base, we add multiple task-specific headers tailored to each of our targeted tasks: identifying the person, estimating age, and classifying ethnicity. Each of these tasks has dedicated output layers that are independently optimized; This allows the model to specialize in the intricacies of each task while taking advantage of the generalized feature representation

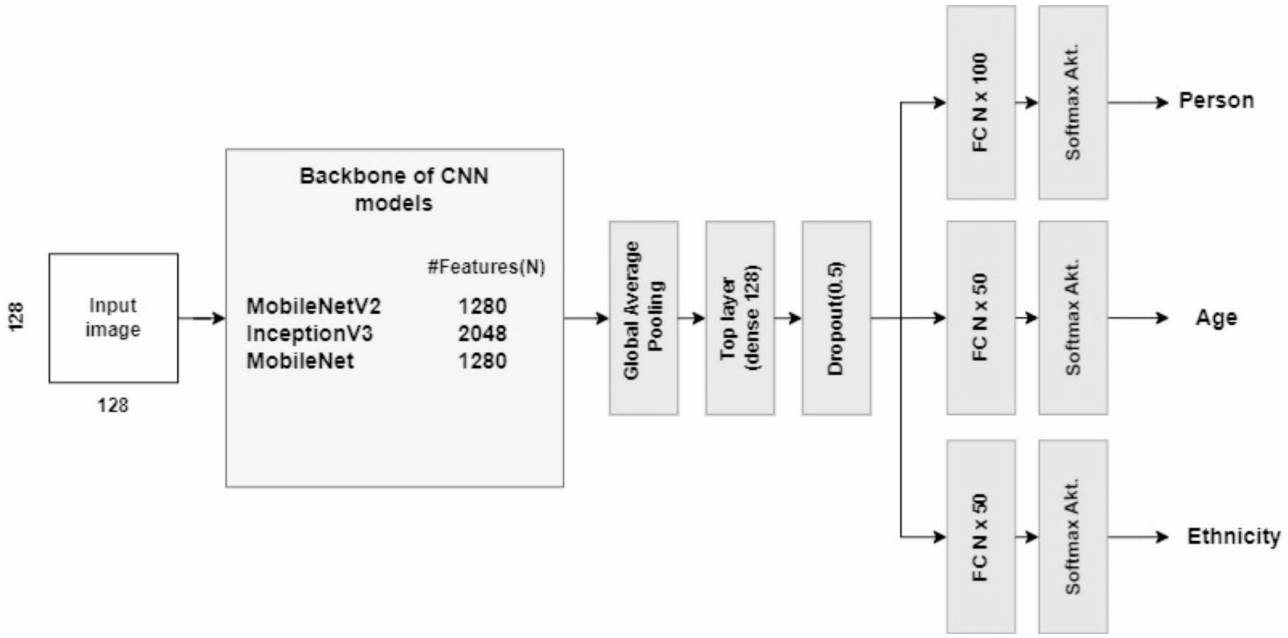

Fig. 5. Proposed MTL models.

Task Type	Number of Classes	Images Per Class	Total Images
Person	9	150	1350
Age	3	450	1350
Ethnicity	3	450	1350

Table 2. Summary of the dataset.

provided by shared layers. Our method integrates complex feature extraction using CNN architectures with a multitask learning model that leverages hard parameter sharing. This work not only enables efficient use of computational resources but also improves the model's capacity to generalize across diverse but interrelated tasks. It solves multiple tasks simultaneously using a single network. For this reason, more advanced backbone architectures such as MobileNetV2, InceptionV3, and MobileNet have been integrated. Figure 5 shows the network architectures used in this study in detail.

Figure 5 shows the MTL framework using CNN models such as MobileNetV2, InceptionV3, and MobileNet as feature extractors. CNNs process 128×128 input images and output 128-dimensional feature vectors. These features are fed into a shared layer, followed by task-specific, fully connected layers. The system predicts multiple characteristics, including person, age, and ethnicity simultaneously. Each task-specific branch consists of a fully connected layer followed by softmax activation to output prediction probabilities for each task.

Face dataset

When training a multitask model, we need datasets that provide labeled data for each task the network should perform. This allows the network to learn to perform all tasks simultaneously. Our MTL models used labeled data from the VGGFace2 dataset to train and develop a comprehensive framework that can simultaneously perform tasks related to person identification, age estimation, and ethnicity classification. Cao et al.[36] created the VGGFace2 dataset, a widely recognized resource in the field of computer vision specifically designed for face recognition tasks.

For this study, a subset of the VGGFace2 dataset consisting of 1350 images was used. These images are organized according to three main tasks: person, age, and ethnicity. Person categorization consists of nine classes with 150 images in each class. Similarly, the age and ethnicity classifications each contain three classes with 450 images in each class. Table 2 provides a summary of the dataset composition.

Table 3 details the distribution of the VGGFace2 dataset used in this study, including the total number of images and the separation between training and testing subsets. We used 70% of the images for training and 30% for testing subsets, ensuring a balanced approach to model development and evaluation.

Raspberry Pi board

The Raspberry Pi 4 stands as a flexible single-board computer (SBC) developed by the Raspberry Pi platform. It is the latest version of the popular Raspberry Pi series and offers significant improvements in performance and features compared with its predecessors. Raspberry Pi 4 is a highly durable device that can operate continuously

Task Type	Training Set (70%)	Testing Set (30%)
Person	945	405
Age	945	405
Ethnicity	945	405

Table 3. Distribution of the VGGFace2 data set used in this study.

Equipment	Features
Processor	Broadcom BCM2711, featuring a Quad-core Cortex-A72 (ARMv8) 64-bit SoC running at 1.5 GHz.
Memory (RAM)	4 GB
GPU	VideoCore VI Graphics Processor
Video Outputs	2 × micro-HDMI ports (supporting resolutions up to 4Kp60)
Connection	Gigabit Ethernet, 2.4 GHz and 5 GHz Wi-Fi, Bluetooth 5.0
USB Ports	2×USB 2.0 and 2×USB 3.0
GPIO	40-pin GPIO
Storage	The MicroSD card slot serves as the port for both the operating system and data storage.
Power	The USB-C port as the power supply input point.

Table 4. Raspberry Pi 4 technical specifications.

as a server-level machine. It has very low power consumption, and the heat generated by its CPU is minimal. It finds suitability for a wide range of applications that do not require high processing power. Some common uses of Raspberry Pi include home security systems, robot controllers, desktop computers, media centers, and web servers. Table 4 shows the different components and specifications of Raspberry Pi 4.

Raspberry Pi 4 sets a new standard in miniature computing. With its advanced processing capacity, expanded memory options, increased connectivity options, and dual 4 K display support, Raspberry Pi 4 offers unlimited possibilities for creating innovative projects.

A webcam represents a digital camera capable of connecting to a computer, allowing the user to send images directly to different parts of the world over the Internet. These cameras can be connected to a computer using different methods such as USB ports or Wi-Fi. An Everest SC-HD03 model webcam with 1080p resolution and excellent image quality was used for this project.

Results

In this section, experiments were conducted using the proposed face recognition method based on the MTL model procedure to evaluate the performance of the system created using the proposed technique. MTL models trained on the Vggface2 dataset were used in this work. We used the MobileNet, MobileNetV2, and InceptionV3 models as shared layers in our MTL architecture. Then, new layers were added after the CNN models and before the main task layers, followed by classification layers for each task. After the MTL models were trained and their features were learned, they were recorded with their weights for application.

First, dataset images were entered into the MTCNN model on a class basis. MTCNN looked for facial features such as nose, mouth, and eyes in the images, recorded their positions, and then calculated the rotation angle to rotate the image so that the two eyes were on the same y-axis and the nose and mouth were the same. The face was then centered, and the indices of the bounding box that outlined the detected face were rotated. The face was then cropped and stored in a new folder with the same name but in a different directory.

The next step is to define the structure of the MTL models. Our work examined facial recognition and used a combinatorial multi-task learning approach in three different models: MobileNet, InceptionV3, and MobileNetV2. These base models are frozen so that they cannot be trained. New trainable layers are appended as the top layer on top of the frozen base model. This upper layer consists of a dense layer with ReLU activation followed by a dropout layer. Subsequently, The MTL is added with dense layers followed by a softmax activation layer for each task. The softmax activation layers of the task were added as output, with 9 units corresponding to person identification, 3 units corresponding to age recognition, and 3 units corresponding to ethnicity recognition. The batch size hyperparameter represents the number of iterations required to complete an epoch and is the number of partitions. It is critical to choose a batch size that is not too small to avoid overlearning. The larger the batch size, the faster the training process, but the lower the accuracy. The batch size for our MTL models was determined to be 32. Dividing the total number of training images by batch size gives the number of images in each iteration. Therefore, 30 images are migrated in each iteration, and all images in the dataset need to be migrated in each epoch. To train the MTL models, the epoch was set to 100. The ADAM optimizer was used with a learning rate set to 0.0001.

Configuring parameters and metrics in our face recognition MTL models is crucial for optimal performance. When setting up the model for training, we need to adjust the loss, accuracy, and optimizer functions. The loss function helps us understand how far the model's predictions are from the desired outcomes, aiming to

minimize errors. Since we are dealing with a model that classifies objects with multiple categories, the sparse_categorical_crossentropy loss function has been preferred.

By using the accuracy function, we observe how well the model performs. This function simply indicates how often the model's predictions are correct, providing a clear picture of the model's performance. Due to its efficiency in handling datasets and non-stationary targets, the ADAM optimizer has been chosen.

Additionally, to combat overfitting, the 'early stopping' callback has been implemented. This mechanism continuously monitors the model's accuracy and triggers the 'early stopping' procedure if no improvement is seen over 3 consecutive epochs. For further refinement, the training process was initiated with a relatively high learning rate of 0.0001. However, to dynamically fine-tune the learning process, the ReduceLROnPlateau function has been used. This function automatically reduces the learning rate by a factor of 0.5 when the model's accuracy plateaus, ensuring smoother convergence and preventing overshooting the target. This way, our MTL models achieve strong performance while reducing the risk of overfitting.

To save our trained model and its weights, the ModelCheckpoint command has been used. This allows us to easily retrieve them for the testing phase later. By using ModelCheckpoint, we gain the flexibility to save the best-performing weights obtained during training. This ensures that we can preserve the best model weights throughout the training process, enhancing the model's performance and allowing access to these optimal weights even after any interruptions during training.

The MTL models were trained with the epoch set to 100. This includes MTL MobileNet, MTL MobileNetV2, and MTL InceptionV3. The training process for the MTL MobileNet model was stopped after 62 epochs. For the identity task, the test loss was 0.0465 and the test accuracy was 0.9901; for the age task, the test loss was 0.0200 and the test accuracy was 0.9926; and for the ethnicity task, the test loss was 0.0212 and the test accuracy was 0.9951. The best results were achieved at the 59th epoch, with a test loss of 0.0453 and test accuracy of 0.9901 for the identity task, a test loss of 0.0189 and test accuracy of 0.9926 for the age task, and a test loss of 0.0221 and test accuracy of 0.9951 for the ethnicity task. The model spent approximately 48 min on the Raspberry Pi.

The training process for the MTL MobileNetV2 model was stopped after 69 epochs. For the identity task, the test loss was 0.0681 and the test accuracy was 0.9827; for the age task, the test loss was 0.0664 and the test accuracy was 0.9728; and for the ethnicity task, the test loss was 0.0337 and the test accuracy was 0.9901. The best results were achieved at the 66th epoch, with a test loss of 0.0682 and test accuracy of 0.9753 for the identity task, a test loss of 0.0649 and test accuracy of 0.9728 for the age task, and a test loss of 0.0343 and test accuracy of 0.9926 for the ethnicity task. The MTL MobileNetV2 spent approximately 62 min on the Raspberry Pi.

Finally, the training process for the MTL InceptionV3 model was stopped after 41 epochs. For the identity task, the test loss was 0.2120 and the test accuracy was 0.9333; for the age task, the test loss was 0.1206 and the test accuracy was 0.9556; and for the ethnicity task, the test loss was 0.1209 and the test accuracy was 0.9753. The best results were achieved at the 38th epoch, with a test loss of 0.1910 and test accuracy of 0.9397 for the identity task, a test loss of 0.0962 and test accuracy of 0.9693 for the age task, and a test loss of 0.1142 and test accuracy of 0.9545 for the ethnicity task. The results demonstrate that the MobileNet model strongly supports the performance of our MTL model. The MTL InceptionV3 spent approximately 69 min on the Raspberry Pi.

Table 5 provides a clear representation of the person identification, age estimation, and ethnicity recognition results of the three MTL models. These results include models InceptionV3, MobileNet, and MobileNetV2 as Sharing layers.

The evaluation of the trained model's performance on the test dataset relies on metrics such as precision, recall, F1 score, and accuracy, as shown in Eqs. 4,5,6, and 7:

$$Precision = \frac{True\ Positives + False\ Positives}{True\ Positives} \tag{4}$$

$$Recall = \frac{True\ Positives + False\ Negatives}{True\ Positives} \tag{5}$$

$$F1 = 2 \times \frac{Precision + Recall}{Precision \times\ Recall} \tag{6}$$

$$Accuracy = \frac{True\ Positives + True\ Negatives}{True\ Positives + True\ Negatives + False\ Positives + False\ Negatives} \tag{7}$$

Table 6 shows the precision, recall, and F1 score of the MTL models.

Table 6 summarizes the precision, recall, and F1 scores for person identification, age estimation, and ethnicity recognition across three models. In person identification, MTL-InceptionV3 achieved 93.6% precision, 93.3% recall, and 93.4% F1 score, while MTL-MobileNet outperformed with 99% across all metrics. MTL-MobileNetV2 followed closely with 98.4% precision, 98.3% recall, and 98.3% F1 score. For age estimation, MTL-

	Person				Age				Ethnicity				
Model	Training accuracy	Training loss	Test accuracy	Test loss	Training accuracy	Training loss	Test accuracy	Test loss	Training accuracy	Training loss	Test accuracy	Test loss	Time Spent (Seconds)
MTL-InceptionV3	0.9460	0.1686	0.9333	0.2120	0.9757	0.0783	0.9556	0.1206	0.9712	0.0932	0.9753	0.1209	4098.19
MTL-MobileNet	0.9905	0.0686	0.9901	0.0465	0.9884	0.0411	0.9926	0.0200	0.9915	0.0369	0.9951	0.0212	2867.94
MTL-MobileNetV2	0.9852	0.0669	0.9827	0.0681	0.9873	0.0411	0.9728	0.0664	0.9873	0.0456	0.9901	0.0337	3712.28

Table 5. Accuracy, loss results and time taken to train the MTL models.

Model	Person			Age			Ethnicity		
	Precision (%)	Recall (%)	F1 score (%)	Precision (%)	Recall (%)	F1 score (%)	Precision (%)	Recall (%)	F1 score (%)
MTL-InceptionV3	93.6	93.3	93.4	95.6	95.6	95.6	97.6	97.5	97.5
MTL-MobileNet	99	99	99	99.3	99.26	99.26	99.5	99.5	99.5
MTL-MobileNetV2	98.4	98.3	98.3	97.3	97.3	97.3	99	99	99

Table 6. Performance metrics of the MTL models.

Age Classification Report:

	precision	recall	f1-score	support
Young	0.99	0.99	0.99	135
Adult	0.94	0.96	0.95	135
Old	0.96	0.93	0.95	135
accuracy			0.96	405
macro avg	0.96	0.96	0.96	405
weighted avg	0.96	0.96	0.96	405

Ethnicity Classification Report:

	precision	recall	f1-score	support
Asian	0.97	0.99	0.98	135
White	0.97	0.98	0.97	135
Black	1.00	0.98	0.99	135
accuracy			0.98	405
macro avg	0.98	0.98	0.98	405
weighted avg	0.98	0.98	0.98	405

Person Classification Report:

	precision	recall	f1-score	support
Dalai Lama	0.89	0.93	0.91	45
Dina Lohan	1.00	0.98	0.99	45
Airi Suzuki	0.85	0.89	0.87	45
Ilkka Kanerva	0.98	0.96	0.97	45
Kyousake Hamao	0.89	0.87	0.88	45
Denis Mukwege	0.91	0.93	0.92	45
Kendrick Perkins	0.91	0.91	0.91	45
Vivica Fox	1.00	0.98	0.99	45
Vaclav klaus	0.95	0.93	0.94	45
accuracy			0.93	405
macro avg	0.93	0.93	0.93	405
weighted avg	0.93	0.93	0.93	405

Fig. 6. MTL InceptionV3 model classification report.

Age Classification Report:

	precision	recall	f1-score	support
Young	1.00	1.00	1.00	135
Adult	0.98	1.00	0.99	135
Old	1.00	0.98	0.99	135
accuracy			0.99	405
macro avg	0.99	0.99	0.99	405
weighted avg	0.99	0.99	0.99	405

Ethnicity Classification Report:

	precision	recall	f1-score	support
Asian	0.99	1.00	1.00	135
White	1.00	0.99	0.99	135
Black	0.99	1.00	1.00	135
accuracy			1.00	405
macro avg	1.00	1.00	1.00	405
weighted avg	1.00	1.00	1.00	405

Person Classification Report:

	precision	recall	f1-score	support
Dalai Lama	0.98	1.00	0.99	45
Dina Lohan	1.00	1.00	1.00	45
Airi Suzuki	1.00	1.00	1.00	45
Ilkka Kanerva	1.00	1.00	1.00	45
Kyousake Hamao	1.00	1.00	1.00	45
Denis Mukwege	0.98	0.96	0.97	45
Kendrick Perkins	0.96	1.00	0.98	45
Vivica Fox	1.00	1.00	1.00	45
Vaclav klaus	1.00	0.96	0.98	45
accuracy			0.99	405
macro avg	0.99	0.99	0.99	405
weighted avg	0.99	0.99	0.99	405

Fig. 7. MTL MobileNet model classification report.

InceptionV3 scored 95.6% in all metrics, MTL-MobileNet achieved 99.3% precision and 99.26% recall, and MTL-MobileNetV2 scored 97.3% across the board. In ethnicity recognition, MTL-InceptionV3 achieved 97.6% precision, 97.5% recall, and 97.5% F1 score, while MTL-MobileNet excelled with 99.5% in all metrics. MTL-MobileNetV2 also performed well with 99% across all metrics. Overall, MTL-MobileNet demonstrated the most balanced and robust performance across all tasks. Overall, the results demonstrate the effectiveness of all three models on various classification tasks. The MTL-MobileNet sharing layer provides balanced performance in all tasks. Figures 6 and 7, and 8 show the classification results of the MTL models in our face recognition system according to person, age, and ethnicity.

The confusion matrix is a table representing the counted number of correct and incorrect predictions and separates them by class. A summary of the prediction results in the MTL model is displayed in the confusion matrix, as shown in Figs. 9 and 10, and 11.

Age Classification Report:

	precision	recall	f1-score	support
Young	0.99	1.00	0.99	135
Adult	0.96	0.96	0.96	135
Old	0.97	0.96	0.97	135
accuracy			0.97	405
macro avg	0.97	0.97	0.97	405
weighted avg	0.97	0.97	0.97	405

Ethnicity Classification Report:

	precision	recall	f1-score	support
Asian	0.98	0.99	0.98	135
White	0.99	0.98	0.98	135
Black	0.99	0.99	0.99	135
accuracy			0.99	405
macro avg	0.99	0.99	0.99	405
weighted avg	0.99	0.99	0.99	405

Person Classification Report:

	precision	recall	f1-score	support
Dalai Lama	0.96	1.00	0.98	45
Dina Lohan	1.00	0.98	0.99	45
Airi Suzuki	0.90	1.00	0.95	45
Ilkka Kanerva	1.00	0.98	0.99	45
Kyousake Hamao	0.98	0.91	0.94	45
Denis Mukwege	0.98	1.00	0.99	45
Kendrick Perkins	1.00	0.98	0.99	45
Vivica Fox	1.00	0.98	0.99	45
Vaclav klaus	0.98	0.96	0.97	45
accuracy			0.98	405
macro avg	0.98	0.98	0.98	405
weighted avg	0.98	0.98	0.98	405

Fig. 8. MTL MobileNetV2 model classification report.

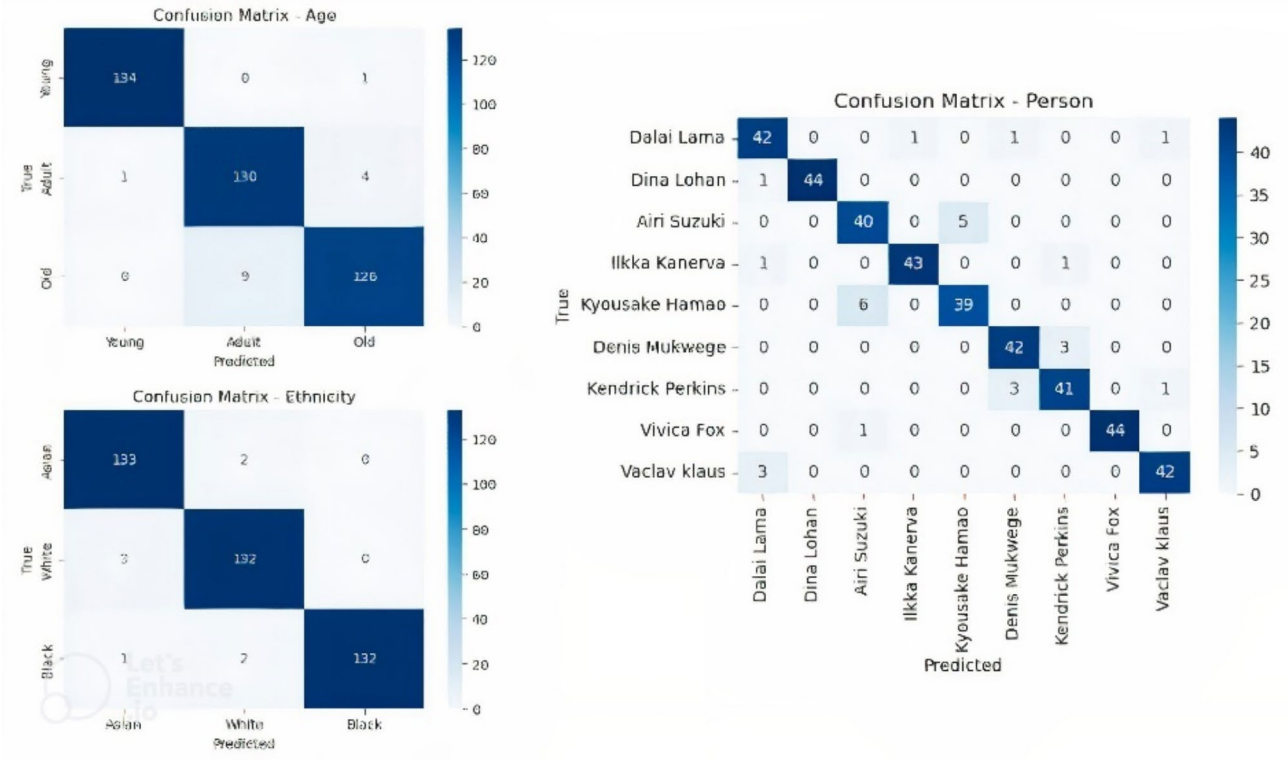

Fig. 9. MTL-InceptionV3 model confusion matrix.

Figure 9 shows the performance of the MTL-InceptionV3 model and the classes in which it made errors are shown. In ethnicity classification, the accuracy is high for Asian, White, and Black classes; there are 131 correct predictions for the White and Black classes and 133 for the Asian class. In age group classification, the accuracy is high for Young, Adult, and Elderly classes; there are 133 correct predictions for the Young class, 126 for the Adult class, and 128 for the Elderly class. In the person identification model, the accuracy is high for classes labeled with the names of different individuals; there are 44 correct predictions for the Dalai Lama, Dina Lohan, and Vivica Fox classes, and 39 for the Kyousake Hamao class.

The MTL-MobileNet model's performance and the classes where it makes errors are shown in Fig. 10. In ethnicity classification, the accuracies are high for the Asian, White, and Black classes. There are 135 correct predictions for the Asian and Black classes, while the White class has 133 correct predictions. In age group classification, the accuracies are high for the Young, Adult, and Elderly classes. There are 135 correct predictions for the Young and Adult classes, while the Elderly class has 132 correct predictions. In identity recognition, the

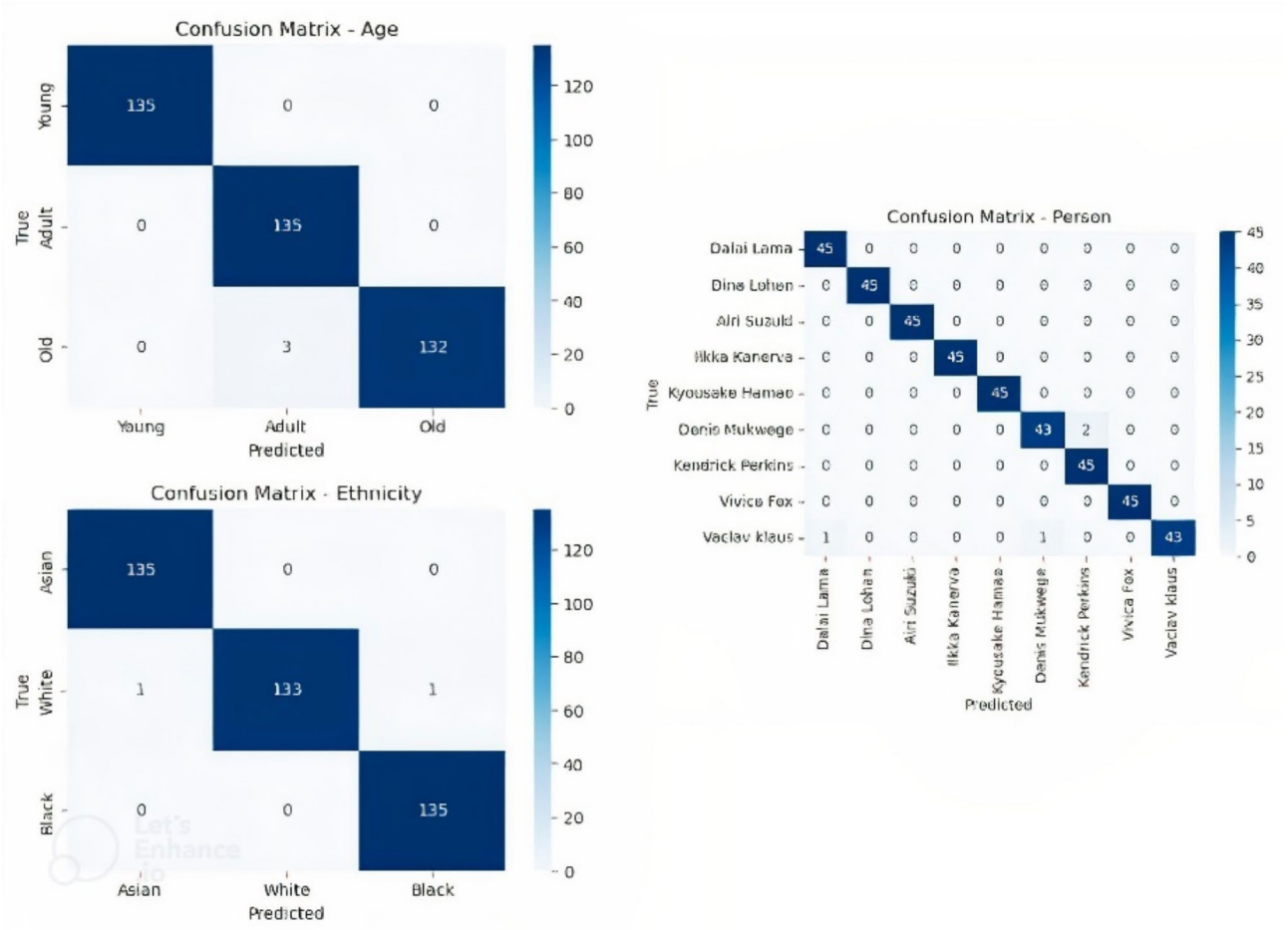

Fig. 10. MTL-MobileNet model confusion matrix.

model achieves high accuracy for different individuals identified by their names. The Dalai Lama, Dina Lohan, Airi Suzuki, Ilkka Kanerva, Kyousuke Hamao, Kendrick Perkins, and Vivica Fox classes each have 45 correct predictions, while the Denis Mukwege class has 43 correct predictions.

Figure 11 shows he performance of the MTL-MobileNetV2 model and the errors it makes in certain classes are shown. In ethnic classification, the accuracies for the Asian, White, and Black classes are high; for example, there are 135 correct predictions in the Asian class. In age group classification, the accuracies for the Young, Adult, and Elderly classes are also high; there are 135 correct predictions in the Young class. In the identity recognition model, the accuracies for the classes labeled with the names of different individuals are high; there are 45 correct predictions in the Dalai Lama and Airi Suzuki classes.

This real-time test proved the model's remarkable performance and potential in various face recognition tasks. The system was tested on live videos and achieved an average of 4.21 fps. Figure 12 shows the test results, demonstrating excellent precision in classifying images from the VGGFace2 database (available at: https://www.kaggle.com/datasets/dimarodionov/vggface2) and live-captured video frames under controlled conditions.

The models are trained on some images for each task. Data augmentation techniques applied to augment the data significantly increased the number of images in the dataset, which significantly improved the accuracy of the system, thereby avoiding the overlearning problem. To evaluate the effectiveness of our MTL facial recognition system, it was compared with models trained using the same dataset and same setup. These comparisons allow us to measure the performance and accuracy of our system.

The original multi-task learning models by Saroop et al.[37] and Foggia et al.[21] were designed to perform four tasks: gender, age, ethnicity, and emotion recognition. We extended these models by incorporating an additional task head for person classification to enhance the models' capabilities and provide a more comprehensive evaluation. The shared layers of the models remained unchanged, and a new dense layer with a softmax activation function was added to predict the person. Similarly, the MT-Vgg16 and MT-Xception models by Vidyarthi et al.[38] and the MTL model by Lee et al.[39] were originally designed for two tasks: gender and age recognition. To facilitate a comprehensive comparison with our proposed three-task MTL models, we extended the MT-Vgg16 and MT-Xception models to include additional task heads for person and ethnicity prediction. Again, the shared layers of the models were unchanged, and new dense layers with softmax activation functions were added to predict person and ethnicity. These modifications have been added to make the comparison fairer.

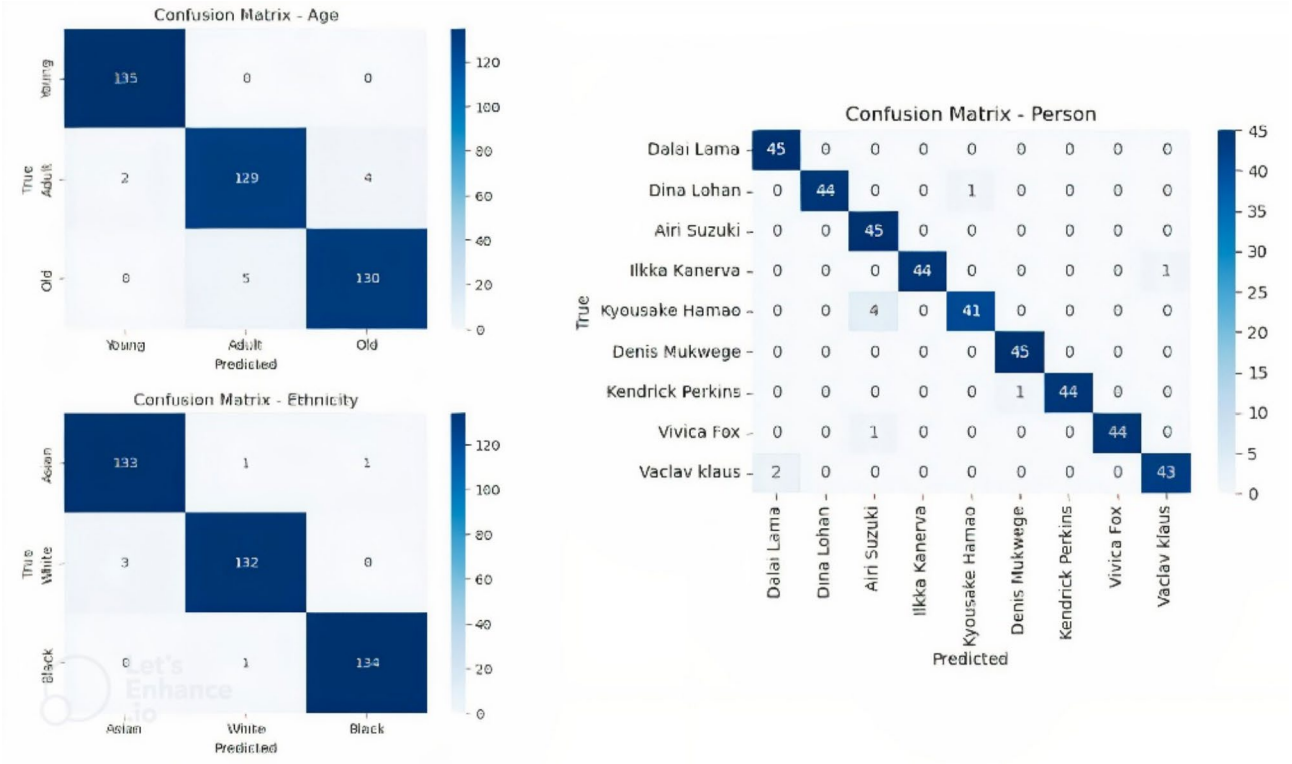

Fig. 11. MTL-MobileNetV2 model confusion matrix.

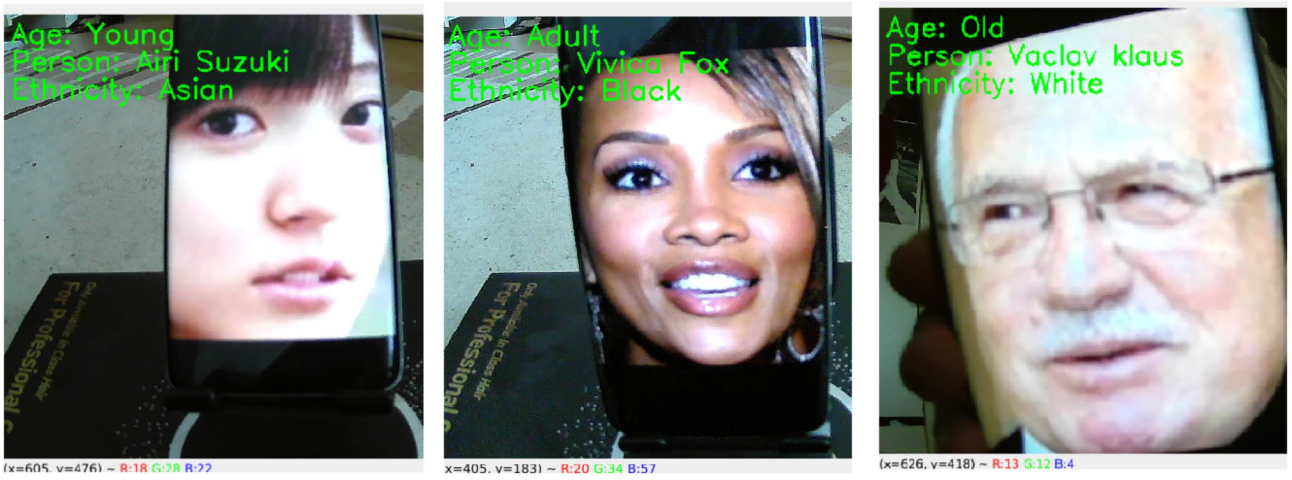

Fig. 12. Real-Time MTL test results.

Table 7 shows the results of the MTL models. Our MTL MobileNet model achieved higher accuracy than all other models.

The MTL-MobileNet model achieved the highest accuracy on every task, showcasing its multitask learning capabilities and the efficiency of the MobileNet architecture. This superior performance highlights the model's overall accuracy and proficiency across various tasks, proving that it can operate accurately. The comparison of results with other multitask learning (MTL) models highlights the superior performance of the proposed MTL-MobileNet and MTL-MobileNetV2 models, which achieve 99% and 98.3% accuracy in person recognition, respectively, outperforming models like MT-SENET-B, MT-Vgg16, and MT-Xception. This performance gap can be attributed to key architectural differences. MobileNet-based models utilize depthwise separable convolutions, which significantly reduce computational complexity and memory usage compared to the traditional convolutional layers in Vgg16 and Xception. This lightweight design enables MobileNet to achieve high accuracy while maintaining efficiency, making it particularly suitable for resource-constrained

Model	Person accuracy (%)	Age accuracy (%)	Ethnicity accuracy (%)
MT-SENET-B[21]	88.9	91.1	94.1
[37]	91.6	96.1	97.5
MT-Vgg16[38]	94.8	96.5	98.7
MT-Xception[38]	96.5	96.1	97.3
[39]	92.1	97.3	97.5
MTL-InceptionV3	93.3	95.6	97.5
MTL-MobileNet	99.0	99.3	99.5
MTL-MobileNetV2	98.3	97.3	99

Table 7. Results of the MTL models comparison.

environments like the Raspberry Pi 4. In contrast, models like Vgg16 and Xception, though powerful, are more computationally intensive and less optimized for low-power devices. Additionally, the MTL-InceptionV3 model, while competitive, falls short of MobileNet's performance due to its higher parameter count and computational demands. These architectural advantages explain why the proposed models excel in multitask learning scenarios, balancing accuracy and efficiency effectively.

Discussion

The multitask learning (MTL) approach significantly enhances model performance on the Raspberry Pi 4, with the MTL-MobileNet model achieving the highest accuracy: 99% in person recognition, 99.3% in age estimation, and 99.5% in ethnicity classification. MobileNet's lightweight design, using depthwise separable convolutions, makes it ideal for resource-constrained environments, outperforming more computationally intensive models like InceptionV3. Practical applications include attendance tracking, smart kiosks, and security systems. However, the Raspberry Pi 4's limited RAM and computational power pose challenges. Future work could address these through hardware acceleration (e.g., TPUs), optimized libraries (e.g., TensorFlow Lite), and techniques like quantization and pruning. While MTL offers advantages like resource sharing, it also involves trade-offs, such as task interference and increased complexity. By addressing these limitations, the system can be refined for broader real-world applications. However, constraints due to the Raspberry Pi 4's limitation computing power were observed with the InceptionV3 model exhibited significantly longer training times and higher memory consumption compared to lightweight architectures like MobileNet and MobileNetV2. This is due to its complex structure (e.g., larger parameter count and deeper layers), which strained the Raspberry Pi 4's limited computational resources. While MobileNet variants leverage depthwise separable convolutions to reduce computational overhead, InceptionV3's resource-intensive design made it less practical for real-time applications or scenarios requiring frequent retraining on edge devices.

Facial recognition technology raises significant ethical concerns, including privacy violations, as it enables mass surveillance without consent. Bias and discrimination are also major issues, with many models showing racial and gender biases that lead to misidentifications, particularly for minority groups. Balancing innovation with ethical responsibility is crucial to ensure fairness and prevent harm.

Conclusion

When we look at the results of our MTL face recognition models, it is seen that our MTL-MobileNet model reaches the highest accuracy in every task. In this study, a dataset consisting of 1350 images was used for three different tasks. In the data set, nine classes were found for the person recognition task, three classes for the age recognition task, and three classes for the ethnicity recognition task.

The MTL-InceptionV3 model performed 93.3% in person accuracy, 95.6% in age accuracy, and 97.5% in ethnicity accuracy. The MTL-MobileNet model reached the highest accuracy of 99% in person accuracy, 99.3% in age accuracy, and 99.5% in ethnicity accuracy. The MTL-MobileNetV2 model showed a performance of 98.3% in person accuracy, 97.3% in age accuracy, and 99% in ethnicity accuracy. These results prove that our MTL-MobileNet model achieves high accuracy rates faster than other models and shows superior performance in all tasks. This study clearly demonstrates the effectiveness of the MTL MobileNet model for facial recognition applications.

The future of multitask facial recognition for Raspberry Pi will focus on improving accuracy, speed, and security through a structured approach. First, we will expand the dataset by adding new classes and more images per class, enabling the model to learn diverse features across various conditions. This will involve collecting and annotating a diverse dataset, ensuring representation across demographics, lighting, and environments. We will also optimize multitask learning (MTL) models for Raspberry Pi, streamline data streaming, and refine power-efficient training techniques for IoT applications. Additionally, we plan to integrate emotion detection and face anti-spoofing (FAS) as new tasks. Emotion detection will identify emotional states from facial expressions, while FAS will enhance security by detecting fraudulent attempts like photos, videos, or masks. Challenges such as increased computational load, real-time performance on resource-constrained devices, and maintaining high accuracy across diverse datasets will be addressed through techniques like model pruning, quantization, and edge computing optimizations. A practical application of this enhanced system is employee attendance tracking, providing organizations with accurate and reliable attendance records. By addressing these improvements

and challenges in a structured manner, the study will achieve stronger performance and broader real-world applicability.

Data availability

The datasets analysed during the current study available from the corresponding author on reasonable request.

Received: 21 October 2024; Accepted: 4 April 2025
Published online: 04 August 2025

References

1. Selvi, K. S., Chitrakala, P. & Jenitha, A. A. Face recognition based attendance marking system. *Int. J. Comput. Sci. Mob. Comput.* **3** (2), 337–342 (2014).
2. Li, P., Prieto, L., Mery, D. & Flynn, P. J. On low-resolution face recognition in the wild: comparisons and new techniques. *IEEE Trans. Inf. Forensics Secur.* **14** (8), 2000–2012 (2019).
3. Li, P., Prieto, M. L., Flynn, P. J. & Mery, D. Learning face similarity for re-identification from real surveillance video: A deep metric solution. In *2017 IEEE International Joint Conference on Biometrics (IJCB)* (pp. 243–252). IEEE. (2017), October.
4. Hu, Y. et al. The development status and prospects on the face recognition. In *2010 4th International Conference on Bioinformatics and Biomedical Engineering* (pp. 1–4). IEEE. (2010), June.
5. Lander, K., Bruce, V. & Bindemann, M. Use-inspired basic research on individual differences in face identification: Implications for criminal investigation and security. *Cogn. Research: Principles Implications.* **3**, 1–13 (2018).
6. Manjula, V. S. & Baboo, L. D. S. S. Face detection identification and tracking by PRDIT algorithm using image database for crime investigation. *Int. J. Comput. Appl.* **38** (10), 40–46 (2012).
7. Introna, L. D. & Nissenbaum, H. Facial Recognition Technology: A Survey of Policy and Implementation Issues; Center for Catastrophe Preparedness and Response, New York University: New York, NY, USA, ; Volume 74, pp. 1–36. (2009).
8. Kostka, G., Steinacker, L. & Meckel, M. Between security and convenience: Facial recognition technology in the eyes of citizens in China, Germany, the united Kingdom, and the united States. *Public. Underst. Sci.* **30**, 644–659 (2021).
9. Bennett, C. J., Surveillance Society, D. & Buckingham and Philadelphia: Open University Press, xii + 189 pp. $27.95 (paper). ISBN 0-33520546-1. *The Inf. Soc.* 2003, *19*, 335–336. (2001).
10. Mohanraj, K. C., Ramya, S. & Sandhiya, R. Face Recognition-Based banking system using machine learning. *Int. J. Health Sci.* **6**, 19724–19730 (2022).
11. Manonmani, S. P., Abirami, G. & Sri, V. N. Spoof prevention for E-Banking using live face recognition. *Int. Res. J. Mod. Eng. Technol. Sci.* **5**, 4600–4603 (2023).
12. Bajrami, X. & Gashi, B. Face recognition with raspberry Pi using deep neural networks. *Int. J. Comput. Vis. Rob.* **12** (2), 177–193 (2022).
13. Mustakim, N. et al. Face recognition system based on raspberry Pi platform. In *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)* (pp. 1–4). IEEE. (2019), May.
14. Wang, H. & Guo, L. Research on face recognition based on deep learning. In *2021 3rd International Conference on Artificial Intelligence and Advanced Manufacture (AIAM)* (pp. 540–546). IEEE. (2021), October.
15. Chen, Y. & Zhang, M. Research on face emotion recognition algorithm based on deep learning neural network. *Appl. Math. Nonlinear Sci.* **9**, 1–16 (2024).
16. Medjdoubi, A., Meddeber, M. & Yahyaoui, K. Smart City surveillance: Edge technology face recognition robot deep learning based. *Int. J. Eng. Trans. Basics*. **37**, 3 (2024).
17. Guo, G. & Zhang, N. A. Survey on deep learning based face recognition. *Comput. Vis. Image Underst.* **189**, 102805 (2019).
18. Kasim, N. A. B. M., Rahman, N. H. B. A., Ibrahim, Z. & Mangshor, N. N. A. Celebrity face recognition using deep learning. *Indones J. Electr. Eng. Comput. Sci.* **12**, 476–481 (2018).
19. Alzu'bi, A., Albalas, F., Al-Hadhrami, T., Younis, L. B. & Bashayreh, A. Masked face recognition using deep learning: A review. *Electronics* **10**, 2666 (2021).
20. Huang, Z., Zhang, J. & Shan, H. *When Age-invariant Face Recognition Meets Face Age Synthesis: a multi-task Learning Framework and a New Benchmark* (IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022).
21. Foggia, P., Greco, A., Saggese, A. & Vento, M. Multi-task learning on the edge for effective gender, age, ethnicity and emotion recognition. *Eng. Appl. Artif. Intell.* **118**, 105651 (2023).
22. Wang, S., Wang, Q. & Gong, M. Multi-task learning based network embedding. *Front. NeuroSci.* **13**, 487803 (2020).
23. Kumar, B. A. & Bansal, M. Face mask detection on photo and real-time video images using Caffe-MobileNetV2 transfer learning. *Appl. Sci.* **13** (2), 935 (2023).
24. Hussain Dahri, F., Ali Chandio, A., Ahmed Dahri, N. & Soomro, M. A. Image Caption Generator Using Convolutional Recurrent Neural Network Feature Fusion, J. Xi'an Shiyou Univ. Nat. Sci. Ed., vol. 19, no. 3, pp. 1088–1095, [Online]. (2023). Available: http://xisdxjxsu.asia
25. Dahri, F. H., Abro, G. E. M., Dahri, N. A., Laghari, A. A. & Ali, Z. A. Advancing robotic automation with custom sequential deep CNN-Based indoor scene recognition. *IECE Trans. Intell. Syst.* **2** (1), 14–26 (2024).
26. Javed, M., Zhang, Z., Dahri, F. H. & Laghari, A. A. Real-Time deepfake video detection using eye movement analysis with a hybrid deep learning approach. *Electron*. https://doi.org/10.3390/electronics13152947 (2024).
27. Shazia, A. et al. Automated early diabetic retinopathy detection using a deep hybrid model. *IECE Trans. Emerg. Top. Artif. Intell.* **1** (1), 71–83 (2024).
28. Dahri, F. H., Mustafa, G. & Dahri, U. Automatic face mask detection and recognition using deep learning. *Int. J. Sci. Eng. Res.* **13** (11), 433–447 (2022).
29. Kumar, B. A. & Misra, N. K. Masked face age and gender identification using CAFFE-modified MobileNetV2 on photo and real-time video images by transfer learning and deep learning techniques. *Expert Syst. Appl.* **246**, 123179 (2024).
30. Alhanaee, K., Alhammadi, M., Almenhali, N. & Shatnawi, M. Face recognition smart attendance system using deep transfer learning. *Procedia Comput. Sci.* **192**, 4093–4102 (2021).
31. Crawshaw, M. Multi-task learning with deep neural networks: A survey. (2020). https://doi.org/10.48550/arXiv.2009.09796
32. Oloyede, M. O., Hancke, G. P. & Myburgh, H. C. A review on face recognition systems: recent approaches and challenges. *Multimedia Tools Appl.* **79** (37), 27891–27922 (2020).
33. Zhang, K., Zhang, Z., Li, Z. & Qiao, Y. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE. Signal. Process. Lett.* **23** (10), 1499–1503 (2016).
34. Kokkinos, I. Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, Honolulu, HI, USA, 21–26 July 2017. (2017).
35. Misra, I., Shrivastava, A., Gupta, A. & Hebert, M. Cross-stitch networks for multi-task learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, Las Vegas, NV, USA, 27–30 June 2016. (2016).

36. Cao, Q., Shen, L., Xie, W., Parkhi, O. M. & Zisserman, A. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018)* (pp. 67–74). IEEE. (2018), May.
37. Saroop, A., Ghugare, P., Mathamsetty, S. & Vasani, V. Facial emotion recognition: A multi-task approach using deep learning. *ArXiv Preprint arXiv* :211015028. (2021).
38. Vidyarthi, P., Dhavale, S. & Kumar, S. Gender and age estimation using transfer learning with multi-tasking approach. In *2022 2nd Asian Conference on Innovation in Technology (ASIANCON)* (pp. 1–5). IEEE. (2022), August.
39. Lee, J. H., Chan, Y. M., Chen, T. Y. & Chen, C. S. Joint estimation of age and gender from unconstrained face images using lightweight multi-task cnn for mobile applications. In *2018 IEEE conference on multimedia information processing and retrieval (MIPR)* (pp. 162–165). IEEE. (2018), April.

Acknowledgements
We are grateful to Ayhan Küçükmanisa for his invaluable guidance and support.

Author contributions
A.A. and I.K. were responsible for drafting the primary text of the manuscript, contributing significantly to the conceptualization and development of the research narrative. A.A. played a key role in preparing all figures and all tables, ensuring they accurately represented the findings and were clear for presentation. Throughout the preparation of the manuscript, all authors actively participated in discussions, providing critical feedback and suggestions. The final version of the manuscript was thoroughly reviewed and approved by all authors before submission.

Funding
This study was conducted without financial support from any public, commercial, or non-profit funding agency.

Declarations

Competing interest
The authors declare no competing interests.

Additional information
Correspondence and requests for materials should be addressed to A.A.A.A.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.