



# A vision system to assist visually challenged people for face recognition using multi-task cascaded convolutional neural network (MTCNN) and local binary pattern (LBP)

A. Baskar<sup>1</sup> · T. Gireesh Kumar<sup>1</sup> · Sathishkumar Samiappan<sup>2</sup>

Received: 6 May 2022 / Accepted: 19 January 2023 / Published online: 12 February 2023  
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2023

## Abstract

Visually impaired people are socially disconnected in situations like face-to-face communication and recognize known individuals. Engaging freely with their sighted counterparts is still challenging and adequate attention is not given to non-verbal communication. This work proposed a compact wearable solution to recognize faces to aid the visually impaired in better social interaction. To address this, we develop a portable embedded device with face recognition capabilities, which facilitates a visually impaired person to recognize faces through the audio feedback system. In preprocessing a hybrid method is proposed for enhancing the visual quality of the face. This is based on LAB color space and Contrast Limited Adaptive Histogram Equalization (CLAHE) with gamma enhancement, accurately recognizing the faces irrespective of various illumination conditions. The efficiency of the proposed methodology is evaluated in a real-time scenario with the following parameters: Process CPU usage, process memory usage, Frame per Second (FPS), Model load analysis, and average CPU load analysis. Experimental results show The MTCNN based LBP uses optimal CPU utilization and improve the accuracy of real-time face recognition.

**Keywords** Face recognition · Visually impaired face recognition · Real-time face recognition · Face detection · Face enhancement · Visually challenged

## 1 Introduction

The term Visual Impairment refers to the loss of vision. Research statistics show that in the year 2015, 285 million people in the world have some form of Visual Impairment, and the numbers also have constantly increased every year. It can be either partial or complete loss of vision. Vision Loss Expert Group (VLEG) published wide estimates of the

number of people who are blind or have moderate to severe visual impairment (MSVI) in the global for the past, present, and future from the years 1990 to 2050 (Bourne et al. 2021, 2017). Figure 1 illustrates the estimated number of blind persons in a global number of all ages from 1990 up to 2050.

Independent living is an important problem faced by visually challenged people in daily life. The major problem faced by visually impaired people includes object recognition, navigation, mobility, difficulty in reading information from the surrounding, and recognizing the face of neighbor for conversation to live independently. Recent advanced technologies trying to reduce the difficulties encountered by visually challenged people through various assistive devices. Based on their functionalities, assistive devices are categorized into non-vision-based and vision-based assistive devices. Non-vision-based assistive devices address navigation and mobility issues, and vision-based assistive device use image sensors to address object, sign board, and face recognition issues. In general, these assistive devices take real-time input through various sensors including ultrasonic sensors, image sensors, etc. followed by computer vision

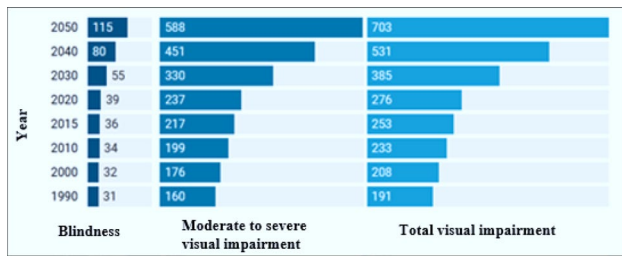
✉ A. Baskar  
a\_baskar@cb.amrita.edu

T. Gireesh Kumar  
t\_gireeshkumar@cb.amrita.edu

Sathishkumar Samiappan  
sathish@gri.msstate.edu

<sup>1</sup> Department of Computer Science and Engineering, Amrita School of Computing, Amrita Vishwa Vidyapeetham, Coimbatore, India

<sup>2</sup> Geosystems Research Institute, Mississippi State University, Starkville, MS 39762, USA



**Fig. 1** the estimated number of visually challenged persons in global numbers at all ages from 1990 up to 2050 (millions)

and machine learning techniques to process and extract useful features and finally provide feedback to users through various modes including auditory, vibratory, or both formats (Tapu et al. 2020).

In general, the challenge seen by visually impaired people is the incapacity to identify the face of sighted counterparts, it limits their social interactions in situations like face-to-face communication. For example, when the members of a group of people interact, they usually share their ideas and thoughts through nonverbal communication. They may use their gaze direction and eye contact to point out to whom the question is directed, and visually impaired people fail to participate during these interactions. Engaging freely with their sighted counterparts is a very important problem, but not adequate attention is given to the development of assistive devices that satisfy the need for access to non-verbal communication, which involves human-to-human interaction.

Human beings can recognize many faces through their visual interactions in a day to day life. Developing an intelligent system for automatic face recognition like the human perceptual system is an active research area. The advance of recent techniques in this domain improves visual quality. A major concern for visually challenged people is to recognize known individuals. Facial Recognition techniques can help them to overcome their disability. But there is not much research done on the Automatic recognition of faces for the visually impaired (Rabia et al. 2013). It has become a major challenge and it demands automatic face recognition systems for the visually impaired.

Face recognition, in general, gained popularity in defense to identify criminals, commercial, human–computer interaction, content-based retrieval, video surveillance, automatic authorization, and law. The automatic Face recognition model comprises four steps: pre-processing, face detection, feature extraction, and face recognition. Firstly, the pre-processing uses an automatic face enhancement technique which improves the quality of the face image in different conditions. Secondly, the automatic face detection technique locates the face image in a given video frame (Yang et al. 2002; Ding et al. 2016; Han et al. 2013; Sanath et al. 2021; Vamsi et al. 2020; Aakash Krishna et al. 2020; Baskar et al.

2018). The face detection system will address the challenges like; pose variations, different backgrounds, occlusion, and noise to improve the automatic face detection for further processing. It is followed by feature extraction techniques as said by Li et al (2018a, b); Zhou et al (2019); Li et al (2019), which extract the feature from the detected face image. The feature vector is generated in two ways: the first approach models the feature vector from the local region like eye, nose, and mouth details present in the face image. The second approach extracts the feature vector from the entire region of the detected face image as a global feature. In the final step, the automatic face recognition techniques use different classifiers, clusters, and deep learning techniques to model the feature vector to the recognition of the face image (Sun et al. 2018).

There was not much research done on the Automatic recognition of faces for the visually impaired. As the demand for face recognition systems for the visually impaired has increased, an effort has been put into developing models that are not only cost-effective but also efficient and user-friendly (Neto et al. 2016; Bhattacharya et al. 2017). The boom in Machine Learning research and Deep learning has become a blessing to impaired people. The algorithms and models used in these topics require more capable computer systems for functioning, but they have helped to increase the accuracy of detection. Several approaches have been proposed to aid people to recognize faces. In general, the Deep learning model as said by Zhang et al (2020) requires a massive dataset for training (Luo et al. 2018). It demands more training data to achieve better accuracy. But this approach demanded that the model be retrained every time there was a newer entry. A newer approach called One-Shot learning was developed. In this approach, training is performed with fewer images but doesn't require retraining as it relies on prior knowledge (Schroff et al. 2015). It relies on learning information about object classes based on a small amount of data. MTCNN as proposed by Zhang et al (2016) is a deep learning-based face detection algorithm. It detects faces and five facial landmarks more accurately under different illumination, sizes, and rotation invariants. The LBP as said by Rahim et al (2013) efficiently describes the texture characteristics of the object and its invariant, and computational efficiency demands various real-time image analytics applications. In general, a face can view as a structure of a micropattern. The LBP operator well describes this Micropattern texture of the face. With the increase in the power of computation, the possibility to include such systems in small wearable devices has increased.

This work proposed a compact wearable solution to recognize faces to aid the visually impaired in better social interaction. To address this, we develop a vision system and portable embedded device with face recognition capabilities, which facilitate a visually impaired person to recognize faces through

the audio feedback system. The objective is to design an efficient face recognition system that accurately recognizes faces irrespective of environmental changes and lighting conditions. The hardware efficiency of the proposed methodology is evaluated in a real-time scenario using the following quality parameters CPU percentage, CPU average load, Memory Usage, Process CPU percentage, Process Memory Usage, and Model Load Analysis and it provides constant feedback between users and a portable device.

The unique contributions of this work are summarized as follows:

- An intelligent prototype for the visually challenged to recognize a known face is established through visual sensors and a microcontroller along with an acoustic feedback system in real-time.
- The head-strap camera mount based model is implemented and analyzed the effect of feature representation using MTCNN and LBP in Raspberry Pi.
- CLAHE with gamma enhancement based method is proposed for enhancing the visual quality of the faces, under various illumination conditions.
- The efficiency of the proposed methodology is evaluated in a real-time scenario under the following parameters: CPU usage, memory usage, Frame per second (FPS), model load analysis, and average CPU load analysis.
- The system is evaluated in various illumination effects and different environmental conditions. The MTCNN based LPB uses optimal CPU utilization and improves the accuracy of real-time face recognition

## 2 Proposed work

This work proposed a compact wearable solution to recognize faces to aid the visually impaired in better social interaction. To address this, we develop a vision system and portable embedded device with face recognition capabilities, which facilitate a visually impaired person to recognize faces through the audio feedback system. The objective is to design an efficient face recognition system that accurately recognizes faces irrespective of environmental changes and lighting conditions. This research focuses on proposing an intelligent prototype for developing a portable embedded device, which is established through visual sensors and a microcontroller along with an acoustic feedback system. Followed by, the head-strap camera mount based model has implemented face enhancement, feature representation, and real-time face recognition through a Raspberry Pi and camera module using the deep learning paradigm.

The proposed work is composed of three modules. The first the most pivotal is the intelligent prototype for developing a portable embedded device, which acts as an intelligent eye for visually impaired people. The head-strap camera mount based model is established through visual sensors and a microcontroller along with an acoustic feedback system. The second automatic Face recognition module recognizes the known face to assist the visually impaired in better social interaction. It comprises four steps: pre-processing, face detection, feature extraction, and face recognition. The third and final module acoustic feedback system facilitates a visually impaired person to

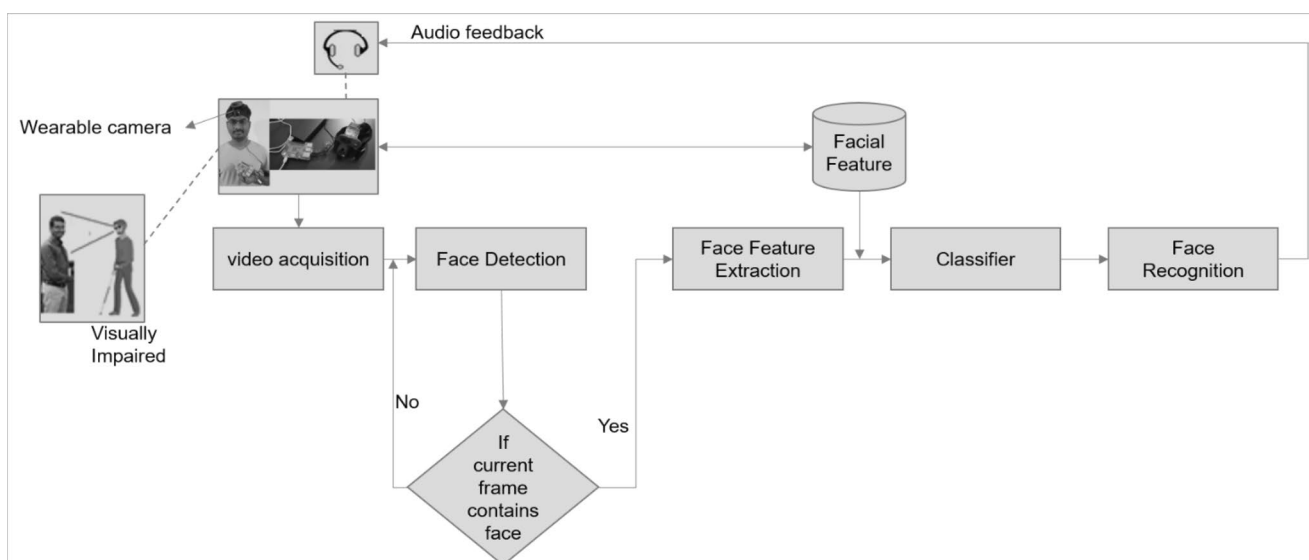


Fig. 2 Proposed System Architecture

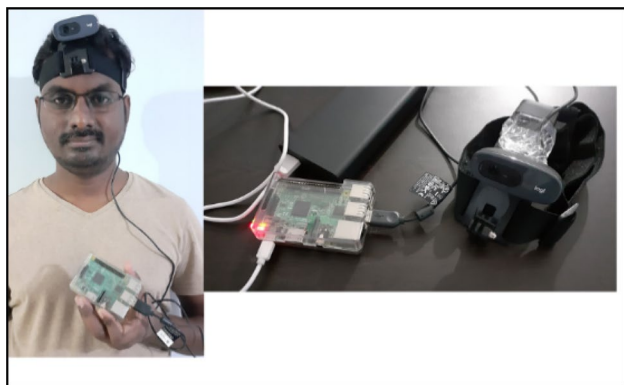


Fig. 3 A portable embedded device prototype

recognize faces through the audio feedback system. Figure 2 shows our proposed system architecture.

### 2.1 The head-strap camera mount based intelligent prototype for visually challenged

The proposed compact wearable is the head-strap camera mount based intelligent prototype, which acts as an intelligent eye for a visually challenged person to recognize the face of known sighted counterparts. Figure 3 shows the prototype of a portable embedded device based on a head-strap camera mount. The wearable head-strap camera mount based model is designed through a Raspberry Pi 3 B + and visual sensor along with an acoustic feedback system. The visual sensor is fixed in the middle of the head-strap mount and facing towards neighbors. In this work, the Raspberry Pi NoIR camera V2 and web camera are used as visual sensors. The camera captures real-time video from the surroundings and feeds it to the face recognition module to recognize the known persons. Successively, the headphone is connected to the raspberry pi through a 3.5 mm audio jack that serves as the acoustic feedback system, which facilitates a visually impaired person to get audio as an output of a recognized person’s name.

### 2.2 An automatic face recognition module

Recognizing the face of their known sighted counterparts for visually challenged people is a primary concern in a face to face communication and facial recognition techniques can help them to overcome the disability. In this work we proposed real-time automatic face recognition techniques for visually impaired people to assist to recognize the face, it comprises four steps: pre-processing, face detection, feature extraction, and face recognition.

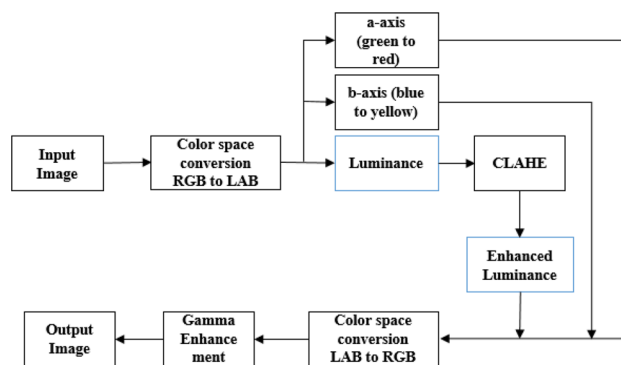


Fig.4 Outline of pre-processing method

### 2.3 Pre-processing

The change of illumination affects the visual quality of the face during image acquisition. It leads to inaccurate results in face detection and recognition techniques (Chen et al. 2020; Chang et al. 2015; Yuan et al. 2022; Yu et al. 2022; Yan et al. 2020). The change of illumination is one of the most challenging problems that largely influence facial detection techniques (Li et al. 2018a, b). Similar faces look dissimilar due to the change of illumination and lead to ineffective results in a facial recognition system. So, a method is needed to normalize the illumination in the face image and improve the visual quality of the face for further processing. In this work, a hybrid method for enhancing the visual quality of the face is proposed. This method is based on LAB color space and Contrast Limited Adaptive Histogram Equalization (CLAHE) with gamma enhancement. The advantage of LAB as said by Asmare et al (2009) color space works like the human eye sees things, and converts RGB color into a lightness component (L) and two-color components (a and b). The CLAHE adjusts the lightness contrast using the L component on small regions in the image and reduces the outliers by limiting the contrast amplification. The proposed pre-processing method is shown in Fig. 4.

In the first step, the input image  $I(x, y)$  (RGB) is converted into LAB color space. It converts red, green, and blue into

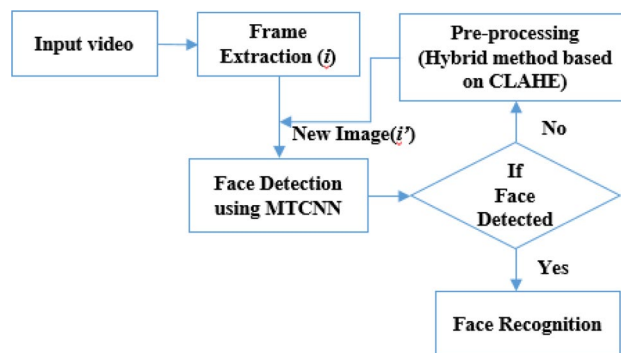


Fig.5 MTCNN based Face Detection

luminance  $I(x, y)'_l$  (black to white), a-axis (green to red)  $I(x, y)'_a$  and b-axis (blue to yellow)  $I(x, y)'_b$  in LAB color space. L-axis represents darkness to lightness and the a-axis, b-axis describe the color character. Successively in step two, CLAHE apply over the luminance part of the LAB color space and produce enhanced luminance  $I(x, y)'_{Enhanced\_l}$ , it improves the visual quality of the face for further process. In general, CLAHE is the variation of adaptive histogram equalization methods in which the image is divided into small regions and the mapping process is applied over tiles, and parameter contrast limiting is used to reduce the image noise amplification. In the third step, the LAB is converted into RGB color space, using the  $I(x, y)'_{Enhanced\_l}$ ,  $I(x, y)'_a$  and  $I(x, y)'_b$  of LAB color space and convert the image into enhanced RGB  $I(x, y)_{newrgb}$ . Finally, the gamma enhancement is applied over the image  $I(x, y)_{newrgb}$  and adjusts the overall brightness level of an image  $I(x, y)_{Enhanced\_rgb}$ . Equations 1–4 describe the entire process of enhancement. This pre-processing module is called if the face detection module gets fails to detect the face image.

(i) **Convert image RGB to LAB color space**

$$I(x, y)'_l, I(x, y)'_a, I(x, y)'_b = RGBtoLAB(I(x, y)) \quad (1)$$

Where,

$RGBtoLAB(I(x, y))$  Function converts RGB to LAB color space.

$I(x, y)'_l$  Luminance (black to white),

$I(x, y)'_a$  a-axis (green to red) and.

$I(x, y)'_b$  b-axis (blue to yellow).

(ii) **CLAHE applies over the luminance part of the LAB color space**

$$I(x, y)'_{Enhanced\_l} = CLAHE(I(x, y)'_l) \quad (2)$$

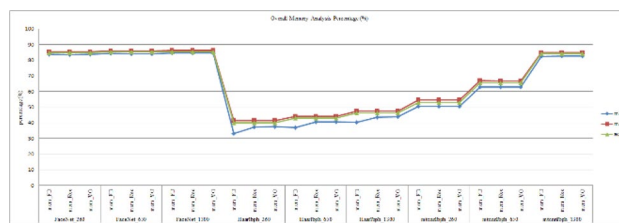


Fig.6 Overall system Memory utilization (%)

Where,

$CLAHE(I(x, y)'_l)$  CLAHE applied to Luminance of image.  
 $I(x, y)'_{Enhanced\_l}$  Enhanced luminance.

(iii) **LAB to RGB conversion**

$$I(x, y)_{newrgb} = LABtoRGB(I(x, y)'_{Enhanced\_l}, I(x, y)'_a, I(x, y)'_b) \quad (3)$$

Where,

$LABtoRGB(I(x, y)'_{Enhanced\_l}, I(x, y)'_a, I(x, y)'_b)$  LAB to RGB conversion in enhanced luminance.

$I(x, y)_{newrgb}$  New enhanced RGB image.

(iv) **Gamma enhancement process**

$$I(x, y)_{Enhanced\_rgb} = I(x, y)_{newrgb}^\gamma \quad (4)$$

Where,

$\gamma$  gamma value.

$I(x, y)_{Enhanced\_rgb}$  gamma enhanced image.

**2.4 Face detection**

The objective is to detect the face regions from the given background. This work uses Multi-Task Cascaded Convolutional Neural Network (MTCNN) for automatic face

Table 1 Quality parameter for the hardware performance study

Parameter	Description	Unit of Measure
CPU percentage	The amount of CPU utilization in a given interval is a percentage. Raspberry pi 3 B+ model had 4 CPU cores and measured the percentage of usage for each core in a given interval	Megabyte (MB) and percentage (%)
CPU average load	Calculates the average system load in the last 1, 5, and 15 min intervals	Percentage (%)
Memory usage	This parameter extracts the following overall information about the system. Total physical memory, Available memory, Memory used, Percentage of memory	Megabyte (MB) and percentage (%)
Process CPU percentage	This parameter extracts measuring the CPU usage of a specific process in percentage	Percentage (%)
Process memory usage	Memory usage of a specific process is analyzed. This parameter extracts the following information about a specific process. Total physical memory, Available memory, Memory used, Percentage of memory	Megabyte (MB) and percentage (%)
Model load analysis	This parameter analysis the amount of CPU utilization for the training vector model, Recognizer model, Audio output model	Percentage (%)



detection. The MTCNN has three convolutional networks (P-Net, R-Net, and O-Net) and it can outperform under different illumination conditions. The first network Proposal Network (P-Net) in MTCNN creates multiple scaled copies of the image and proposes candidate windows, which contain the face and bounding box coordinates for these windows. Non-Maximum Suppression is used to merge highly overlapped candidates. Followed by, all candidate windows from P-Net are fed to Refine Network(R-Net), which further rejects many false candidates. Non-Maximum Suppression is used to merge highly overlapped candidates. Finally, O-Net further reduces those bounding boxes that have low confidence scores and finds five facial landmarks and bounding boxes for face detection. Figure 5 illustrates MTCNN based face detection module.

### 2.5 Face recognition

The face recognition module gets the detected face from the previous step. In this work, we developed FaceNet and Local Binary Pattern (LBP) model for real-time face recognition. The output of the MTCNN based detected face is fed to FaceNet for face recognition. FaceNet is a one-shot model, which directly learns a mapping from face images to a compact Euclidean space where distances directly correspond to a measure of facial similarity. It uses a deep Convolutional Neural Network (CNN) that generates a 128-dimensional (128-D) embedding feature vector. A face embedding is a lower-dimensional feature vector extracted from the face that directly corresponds to a measure of face similarity finally. Followed by, a fully connected layer that uses a softmax classifier to recognize the face using the trained model.

In LBP, the face image is divided into various blocks (3 × 3 or 5 × 5) and analysis of the neighbor’s correlation. The LBP features are extracted from local regions. The textures of the facial regions are encoded for the local region by LBP. Subsequently, the binary strings (e.g. 10001101) are generated for each block. For each neighbor of the central value (threshold), we set a new binary value. We set 1 for values equal to or higher than the threshold and 0 for values lower than the threshold. It is followed by the feature vectors which are represented in histogram format. Finally, LBP concatenated the entire local region LBP into a single face

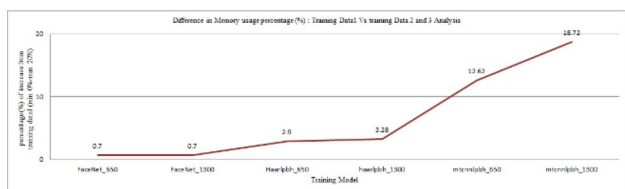


Fig. 7 Difference in memory usage (%): Training Data1 Vs. training data2 and 3

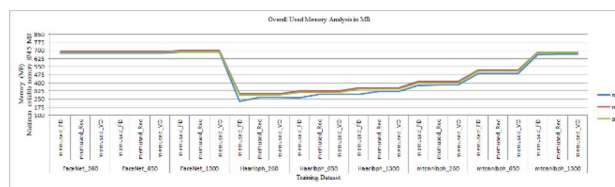


Fig. 8 Overall Memory usage analysis in MB

feature Local Binary Pattern Histogram (LBP). Extracted LBPH feed input to the classifier and it uses predefined facial feature classes, which come from a trained database and classify correct recognized faces.

## 3 Experimental results

This section presents the performance analysis of the proposed work on the head-strap camera mount based intelligent prototype. The following algorithms were used for experimental analysis: Real-time Face recognition in raspberry pi3 using MTCNN based LBP, MTCNN based FaceNet, and Haar based LBP. Subsequently, the quality parameters and datasets for the hardware performance are studied. Table 1 summarizes the type of quality parameter, description, and units of measure. Finally, evaluate the performance of various algorithms with quality parameters, and datasets on different scales. The experimental study captures data under the following scenarios: 1. Response time in real time, 2. consider both indoor and outdoor environments, 3. camera setup through Raspberry Pi NoIR camera and USB camera, 4. Various lighting conditions, 5. feedback through an acoustic feedback system.

### 3.1 Dataset

The dataset consists of 26 classes and 1300 facial images, comprised of different age groups, size of images, view, and illumination. Further, a dataset is fragmented into three training vectors with various scales. The purpose of the different scales is to analyze the hardware load and efficiency

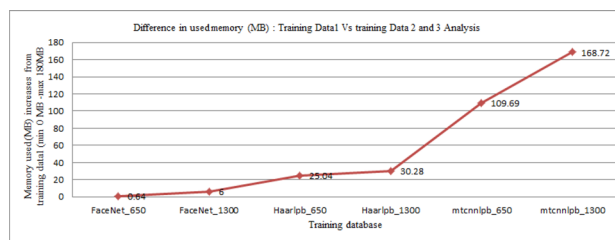


Fig. 9 Difference in used memory (MB): Training Data1 Vs. training data2 and 3

of the proposed algorithms. Training vector1 consists of 26 classes and 10 images per class, training vector 2 consists of 26 classes and 25 images per class, and training vector3 consists of 26 classes and 50 images per class.

### 4 Performance analysis

The experimental setup captures data in two phases. Primarily, real-time data is captured through face detection, face recognition, and audio feedback during execution. Subsequently, CPU utilization is captured during MTCNN, FaceNet, LPB, and audio feedback model loads at the beginning. Followed by quality parameters, evaluate the performance of proposed algorithms in hardware with three training vectors.

#### 4.1 Case 1: memory analysis

This section describes the performance analysis of system memory usage for overall memory and specific process memory. The following statistics are captured during execution: Total physical memory (MB), Available memory (MB), Memory used MB and percentage. The overall memory captures the statistics of overall system memory usage, including algorithm and other system processes, and the specific process memory captures the statistics of memory usage of a proposed algorithm in real-time execution.

#### 4.2 Overall system memory utilization

The experiment considers minimum, maximum, and average memory utilization for performance analysis. Figure 6 describes the percentage of memory utilization of Haar based LBP is low compared to MTCNN based FaceNet and MTCNN based LBP, but it fails to recognize the face in various illumination conditions and poses. MTCNN based LBP and MTCNN based FaceNet perform better in illumination and pose conditions. The horizontal axis describes the proposed algorithm and is evaluated using three training vectors and mem\_FD, mem\_Rec, and mem\_VO is memory

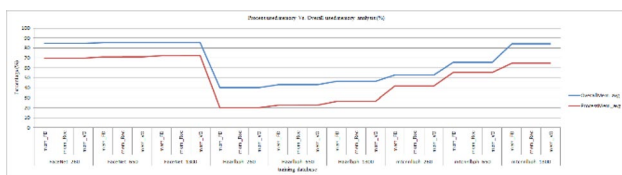


Fig. 10 Overall system memory vs. specific process memory utilization (%)

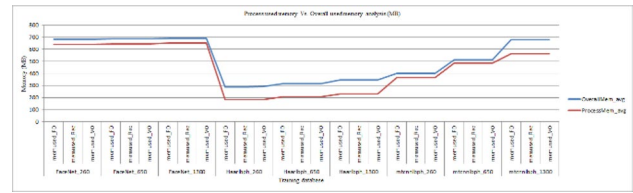


Fig. 11 Overall system memory vs. specific process memory utilization (MB)

utilization captured during face detection, recognition, and audio feedback respectively.

Figure 7 describes at what ratio the percentage of memory utilization varies between training vectors and Eq. 5 defines the calculation of the difference in memory utilization. The percentage of changes for MTCNN based Face Net and Haar based LBP is 1% and 3%-4% respectively and MTCNN based LBP is 13%-19% between training vectors. The ratio of memory utilization varies less for Haar based LBP compare with MTCNN based LBP, but fails to recognize the face in most scenarios.

$$m_{p_{diff}} = (m_m - m_{m-1}) \tag{5}$$

where,

$m_{p_{diff}}$  Memory difference between two training vectors in percentage (%).

$m_{m-1}$  Previous training vector.

$m_m$  the current training vector is arranged in increasing order as per no. of images.

tn = 2,3

Figure 8 illustrates the amount of memory used in MB, we consider minimum, maximum, and average used memory for performance analysis. The embedded board assigns total memory of 874.5 MB out of 1024 MB for executing all the processes. The average memory usage of Haar based LBP differs from 290 to 345 MB, MTCNN based LBP differs from 401 to 679 MB and MTCNN based FaceNet differs from 684 to 691 MB for various training vectors. The horizontal axis describes the proposed algorithm and is evaluated using three training vectors and memused\_FD,

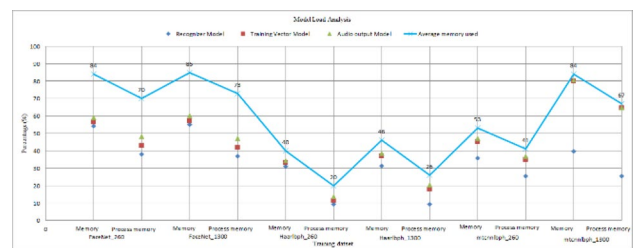
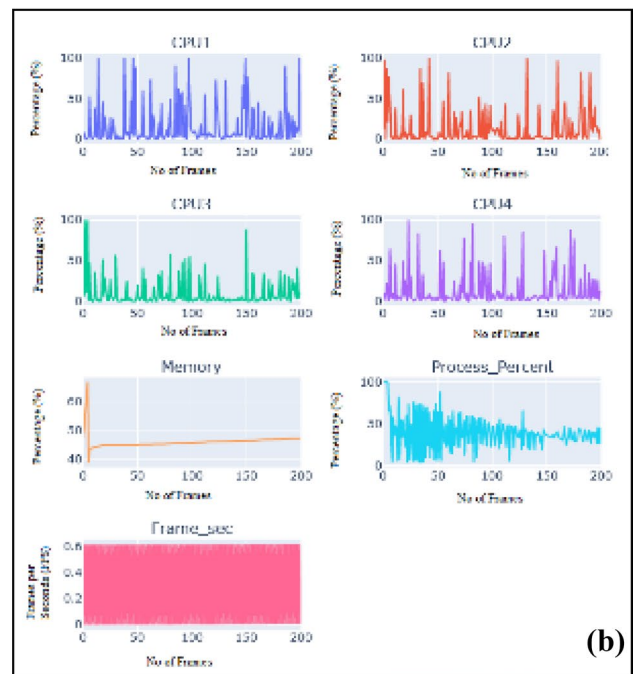
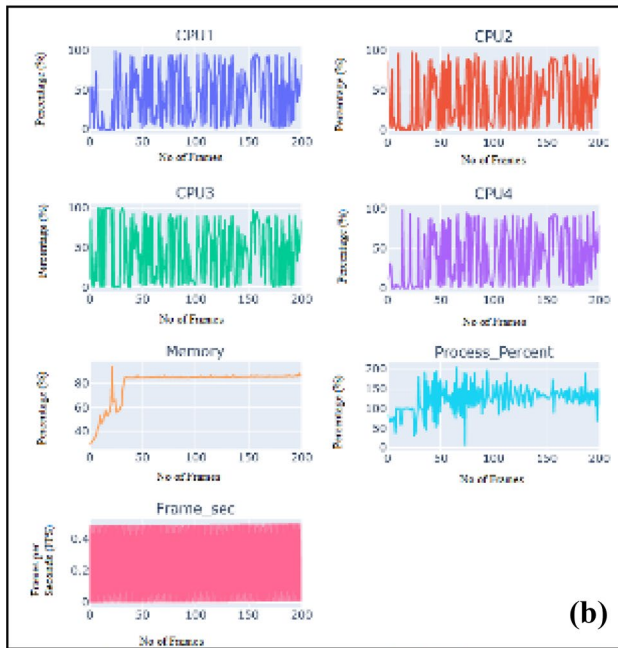
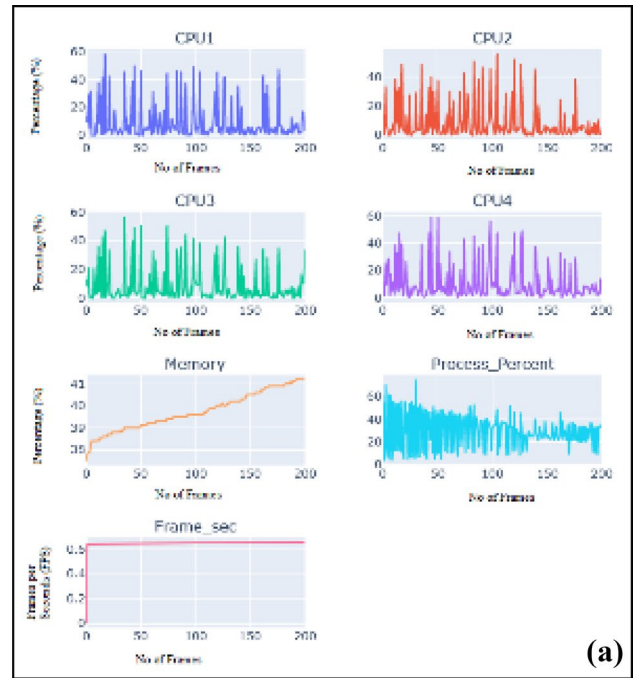
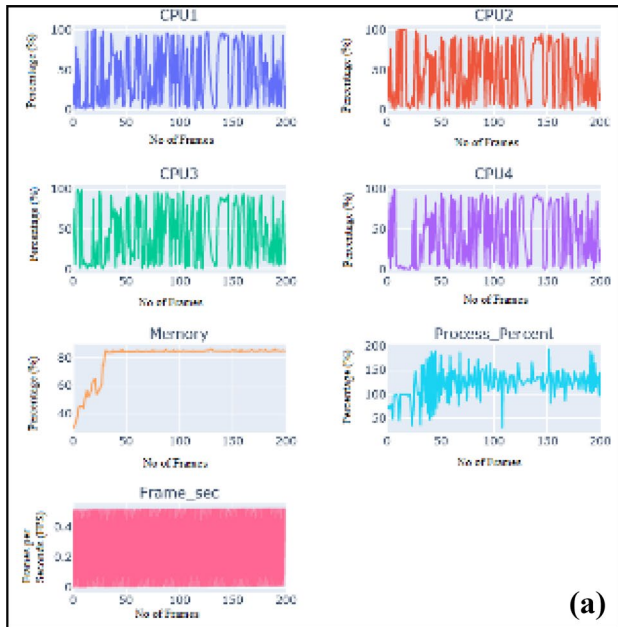


Fig. 12 Average CPU utilization of different Models in execution at the beginning



**Fig.13** **a** The MTCNN based FaceNet average system load analysis for the training vector1 **b** The MTCNN based FaceNet average system load analysis for the training vector3

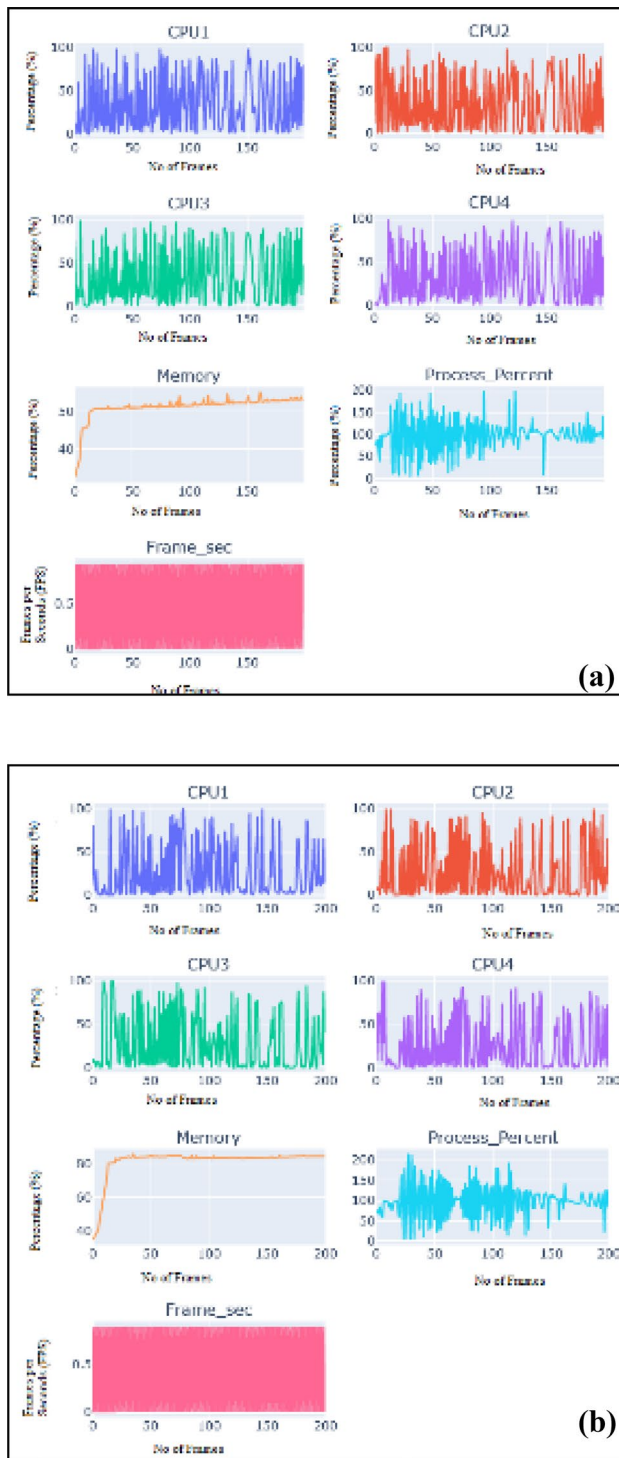
**Fig.14** **a** The Haar based LBP average system load analysis for the training vector1 **b** The Haar based LBP average system load analysis for the training vector3

memused\_Rec, and memused\_VO is memory utilization captured during face detection, recognition, and audio feedback respectively.

Figure 9 describes at what ratio the amount of memory utilization varies between training vectors, measured in MB, and Eq. 6 defines the calculation of the difference in memory utilization. The ratio of memory utilization varies for MTCNN based Face Net and Haar based LBP is

1–6 MB and 25–30 MB respectively and MTCNN based LBP is 109–168 MB between training vectors. The ratio of memory utilization varies less for Haar based LBP compare with MTCNN based LBP, but fails to recognize the face in most scenarios.





**Fig.15** a The MTCNN based LBP average system load analysis for the training vector1 b The MTCNN based LBP average system load analysis for the training vector3

$$m_{\text{used\_diff}} = ((m_m - m_{m-1}) \div m_{\text{total}}) * 100 \tag{6}$$

where,

$m_{\text{used\_diff}}$  Used memory difference between two training vectors in percentage (%).

$m_{m-1}$  Previous training vector.

$m_m$  the current training vector is arranged in increasing order as per no. of images.

$m_{\text{total}}$  Total available memory (MB).

$tn = 2,3$

The Haar based LBP uses less memory utilization compared to others but, it fails in recognizing faces under various illumination and poses. MTCNN based FaceNet works effectively with fewer images in the dataset; in general, FaceNet requires only a few images to train the model. The significant increase in images reduces the accuracy of this model. Both MTCNN based FaceNet with a lesser dataset and MTCNN based LBP performs better in different condition.

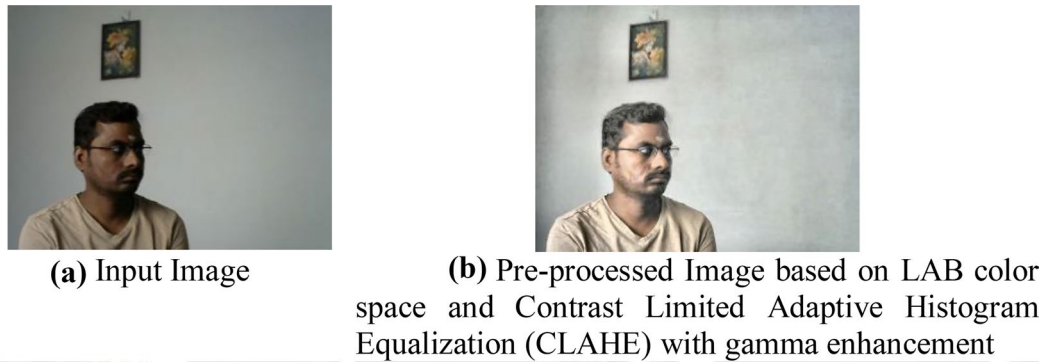
### 4.3 Specific process memory utilization

Memory utilization of a particular algorithm excluding system process in real-time execution is analyzed. Figure 10 compares overall system memory and specific process memory utilization. The graph shows, the MTCNN based FaceNet, Haar based LBP and MTCNN based LBP uses an average of 14%, 22%, and 13% less than overall system memory utilization respectively. Haar based LBP utilizes less memory compared with the other two algorithms and fails to perform better in various illumination conditions.

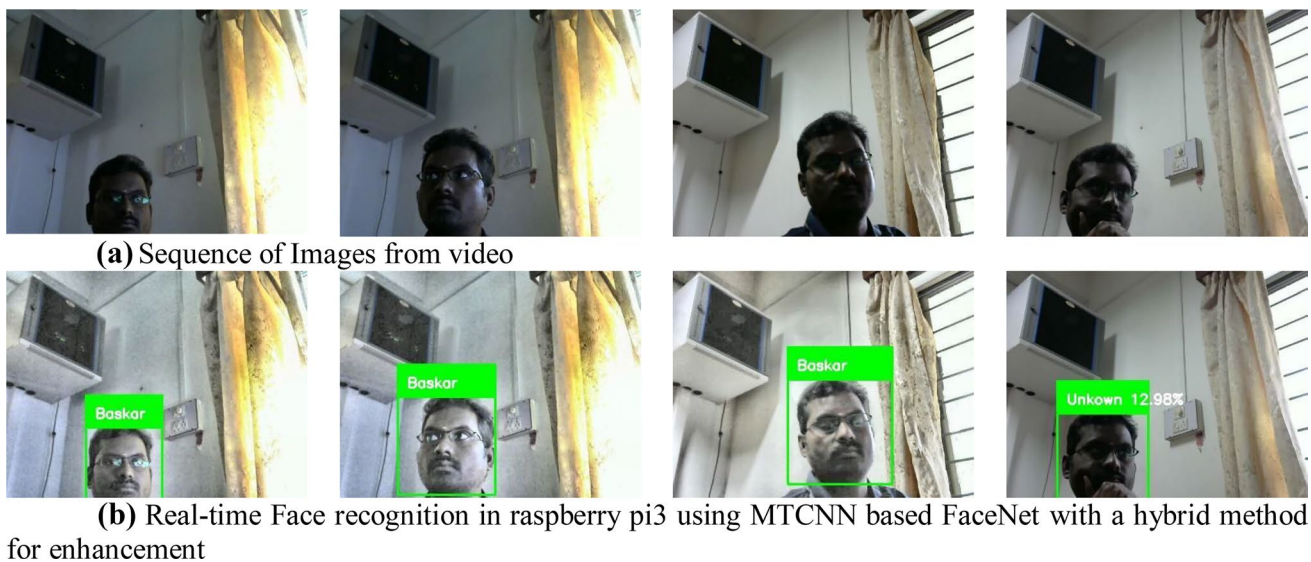
Figure 11 illustrates overall system memory and specific process memory utilization in MB. The MTCNN based FaceNet, Haar based LBP and MTCNN based LBP uses an average of 41 MB, 109 MB, and 59 MB less than overall system memory utilization respectively.

### 4.4 Case 2: average CPU utilization of different models in execution at the beginning

In this experiment at the beginning, the average CPU utilization is captured in three stages of execution: the face detection model (MTCNN, Haar Cascade), face recognition model (FaceNet, LBP), and audio feedback model (text-to-speech). Figure 12 describes the average CPU utilization of model load analysis vs. the average memory utilization of overall and specific process memory. The term *memory* and *process memory* in the horizontal axis describe overall system memory and specific process memory respectively. The experiment evaluated the performance using minimum load (the training vector1 contains minimum no. of images per class) and maximum load (the training vector3 contains maximum no. of images per class). The experimental result shows the face detection, audio feedback model occupies more CPU in execution compared to the recognizer model for MTCNN based approach since it is based on a



**Fig. 16** a Input Image b Pre-processed Image based on LAB color space and Contrast Limited Adaptive Histogram Equalization (CLAHE) with gamma enhancement



**Fig. 17** a Sequence of Images from video. b Real-time Face recognition in raspberry pi3 using MTCNN based FaceNet with a hybrid method for enhancement

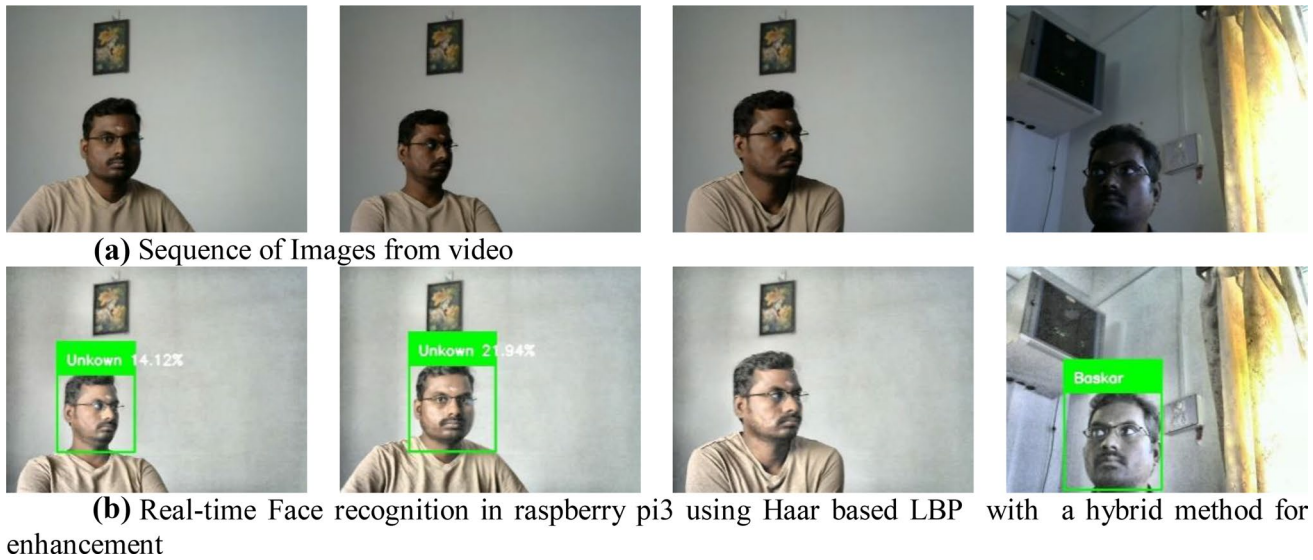
deep learning model and needs to load many libraries during execution. The minimum to maximum load analysis of MTCNN based LBP illustrates differences that are increased by approximately more than 48% from minimum load CPU utilization. The less CPU utilization of the recognizer model benefits recognizing faces quickly in the sequence of frames.

#### 4.5 Case 3: real-time average system load analysis

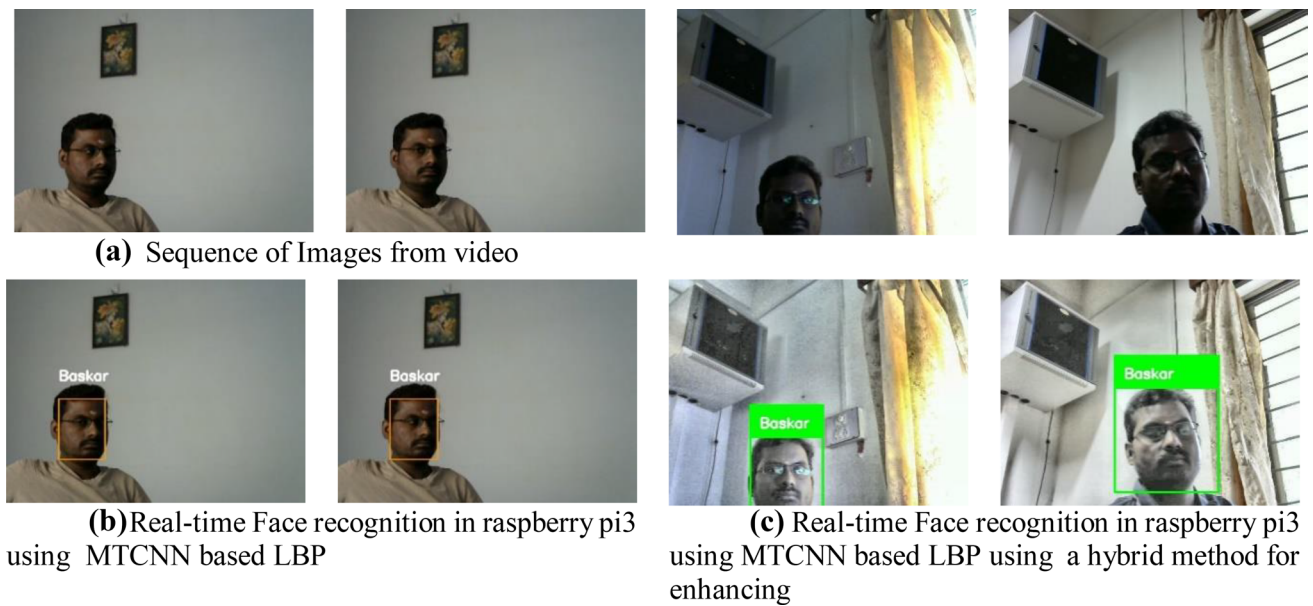
The experimental setup uses the raspberry pi 3 b+ model, having the ARM Cortex-A5, 1.4 GHz processor, and 4 cores. The average system load in 1, 5, 15 min intervals are captured for the processes that are in a runnable state, and the following statistics are estimated in real-time: Individual CPU core utilization, overall system memory utilization, specific process memory utilization, and frames per seconds. The

experiment evaluates the performance between the minimum load (the training vector1 contains minimum no. of images per class) and maximum load (the training vector3 contains maximum no. of images per class). Figure 13a illustrates the average system load analysis for the training vector1 and (b) for training vector3 of MTCNN based FaceNet algorithm. The graph shows, memory utilization is approximately the same for both case1 results and the average system load analysis. The algorithm uses an average of 0.52 FPS to process the frame.

Figure 14a illustrates the Haar based LBP average system load analysis for training vector1 and (b) for training vector3. The graph shows, memory utilization is approximately the same for both case1 results and the average system load analysis. The algorithm uses an average of 0.61 FPS to process the frame.



**Fig. 18** a Sequence of Images from video. b Real-time Face recognition in raspberry pi3 using Haar based LBP with a hybrid method for enhancement



**Fig.19** a Sequence of Images from video

Figure 15a Illustrates the MTCNN based LBP average system load analysis for training vector1 and (b) for training vector3. The graph shows, memory utilization is approximately the same for both case1 results and the average system load analysis. The algorithm uses an average of 0.82 FPS to process the frame.

Figure 16a shows the input image captured at low illumination and Fig. 16b illustrates Pre-processed Image based on LAB color space and Contrast Limited Adaptive Histogram Equalization (CLAHE) with gamma enhancement. Figure 17

(a) and (b) illustrate a sequence of input images and real-time Face recognition in raspberry pi3 using MTCNN based FaceNet with a hybrid method for enhancement. Figure 18a, b sequence of images and real-time Face recognition in raspberry pi3 using Haar based LBP with a hybrid method for enhancement. Figure 19a–c Real-time Face recognition in raspberry pi3 using MTCNN based LBP. The results show MTCNN based LBP performs better in different illumination. The proposed pre-processing technique improves the



results in different illuminations, but Haar based LBP with pre-processing fails in different illumination.

## 5 Discussion

The result demonstrates the proposed intelligent prototype for the visually challenged performs successfully and recognizes the known individual in real-time. The proposed algorithms were trained, tested, and validated using the aforesaid training vector and parameters. It was observed that the Haar based LBP utilizes less memory and better frame per second, but fails in various illumination and pose conditions. MTCNN based FaceNet uses an average of 84% -85% memory for all training vectors, and the MTCNN based LBP uses an average of 52% for training vector1. If increase the no of images per class, progressively increases the average memory utilization. In this case, the average memory utilization of training vector 3 is 84%. The proposed preprocessing technique further improves the results of MTCNN based LBP with training vector2 and the average memory utilization is 62%, which uses optimal memory utilization. In the beginning, MTCNN and FaceNet model spends more CPU time on the loading model, since it is based on a deep learning model and need to load more library for processing. After the model load and the recognizer improve the FPS in a sequence of frames, it was observed that the proposed algorithm's FPS various between 0.6 and 0.8 and is needed to improve further processing frames in real-time.

## 6 Conclusion

In this paper, an intelligent prototype for the visually challenged to recognize a known face was developed through a visual sensor and a microcontroller along with an audio feedback system in real-time. The head-strap camera mount-based model was implemented and analyzed the effect of feature representation through a Raspberry Pi and camera module using the deep learning paradigm. A hybrid method based on LAB color space and Contrast Limited Adaptive Histogram Equalization (CLAHE) with gamma enhancement was developed for enhancing the visual quality of the faces. The efficiency of the proposed methodology was evaluated in a real-time scenario with the following parameters: Process CPU usage, process memory usage, Frame per Second (FPS), Model load analysis, and average CPU load analysis. Experimental results show The MTCNN based LBP uses optimal CPU utilization and improve the accuracy for real-time face recognition in various illuminations and pose conditions.

The proposed system needs improvement in the following scenarios, the person walking with a wearable device and capturing the real time data, optimizing the frames per second to improve the speed.

## References

- Aakash Krishna GS, Pon VN, Rai S, Baskar A (2020) Vision system with 3D audio feedback to assist navigation for visually impaired. *Proc Comput Sci* 167:235–243
- Asmare MH, Asirvadani VS, Iznita L (2009) Color space selection for color image enhancement applications. *International conference on signal acquisition and processing*. IEEE, pp 208–212
- Baskar A, Gireesh Kumar T (2018) Facial expression classification using machine learning approach: a review. *Data Eng Intell Comput* 542:337–345
- Bhattacharya J, Marsi S, Carrato S, Frey H, Ramponi G (2017) Feeding a DNN for face verification in video data acquired by a visually impaired user. *40th international convention on information and communication technology*. Electronics and Microelectronics (MIPRO), pp 1084–1089
- Bourne RR, Flaxman SR, Braithwaite T, Cicinelli MV, Das A, Jonas JB, Keeffe J, Kempen JH, Leasher J, Limburg H, Naidoo K (2017) Vision loss expert group magnitude, temporal trends, and projections of the global prevalence of blindness and distance and near vision impairment: a systematic review and meta-analysis. *Lancet Glob Health-Elsevier* 5(9):e888–e897. [https://doi.org/10.1016/S2214-109X\(17\)30293-0](https://doi.org/10.1016/S2214-109X(17)30293-0)
- Bourne R, Steinmetz JD, Flaxman S, Briant PS, Taylor HR, Resnikoff S, Casson RJ, Abdoli A, Abu-Gharbieh E, Afshin A, Ahmadi H (2021) Trends in prevalence of blindness and distance and near vision impairment over 30 years: an analysis for the global burden of disease study. *Lancet Glob Health* 9(2):e130–e143. [https://doi.org/10.1016/S2214-109X\(20\)30425-3](https://doi.org/10.1016/S2214-109X(20)30425-3)
- Chang X, Nie F, Wang S, Yang Y, Zhou X, Zhang C (2015) Compound rank-k projections for bilinear analysis. *IEEE Trans Neural Netw Learn Syst* 27(7):1502–1513
- Chen K, Yao L, Zhang D, Wang X, Chang X, Nie F (2020) A semi-supervised recurrent convolutional attention model for human activity recognition. *IEEE Trans Neural Netw Learn Syst* 31(5):1747–1756
- Ding C, Tao D (2016) A comprehensive survey on pose-invariant face recognition. *ACM Trans Intell Syst Technol* 7(3):1–40. <https://doi.org/10.1145/2845089>
- Han H, Shan S, Chen X, Gao W (2013) A comparative study on illumination preprocessing in face recognition. *Pattern Recogn* 46(6):1691–1699. <https://doi.org/10.1016/j.patcog.2012.11.022>
- Li Z, Nie F, Chang X, Nie L, Zhang H, Yang Y (2018a) Rank-constrained spectral clustering with flexible embedding. *IEEE Trans Neural Netw Learn Syst* 29(12):6073–6082
- Li Z, Nie F, Chang X, Yang Y, Zhang C, Sebe N (2018b) Dynamic affinity graph construction for spectral clustering using multiple features. *IEEE Trans Neural Netw Learn Syst* 29(12):6323–6332
- Li Z, Yao L, Chang X, Zhan K, Sun J, Zhang H (2019) Zero-shot event detection via event-adaptive concept relevance mining. *Pattern Recogn* 88:595–603
- Luo M, Chang X, Nie L, Yang Y, Hauptmann AG, Zheng Q (2018) An adaptive semi supervised feature analysis for video semantic recognition. *IEEE Trans Cybern* 48(2):648–660
- Neto LB, Grijalva F, Maike VR, Martini LC, Florencio D, Baranauskas MC, Rocha A, Goldenstein S (2016) A Kinect-based wearable face recognition system to aid visually impaired users. *IEEE Trans Human-Mach Syst* 47(1):52–64
- Rabia J, Ali SA, Arabnia HR (2013) Face recognition for the visually impaired. In: *Proceedings of the international conference on information and knowledge engineering (IKE)*. The steering committee of the world congress in computer science, Computer engineering and applied computing (WorldComp). IEEE, pp 1–7

- Rahim MA, Azam MS, Hossain N, Islam MR (2013) Face recognition using local binary patterns (LBP). *Glob J Comp Sci Technol* 13(4):1–8
- Sanath K, Meenakshi K, Rajan M, Balamurugan V, Harikumar ME (2021) RFID and face recognition based smart attendance system. 5th international conference on computing methodologies and communication (ICCMC). ICCMC, pp 492–499
- Schroff F, Kalenichenko D, Philbin J (2015) Facenet: a unified embedding for face recognition and clustering. *Proceedings of the IEEE conference on computer vision and pattern recognition*. IEEE, pp 815–823
- Sun X, Wu P, Hoi SC (2018) Face detection using deep learning: an improved faster RCNN approach. *Neurocomputing* 299:42–50
- Tapu R, Mocanu B, Zaharia T (2020) Wearable assistive devices for visually impaired: a state of the art survey. *Pattern Recogn Lett* 137:37–52
- Vamsi M, Soman KP, Guruvayurappan K (2020) Automatic seat adjustment using face recognition. *International conference on inventive computation technologies (ICICT)*. ICICT, pp 449–453
- Yan C, Chang X, Luo M, Zheng Q, Zhang X, Li Z, Nie F (2020) Self-weighted robust LDA for multiclass classification with edge classes. *ACM Trans Intell Syst Technol (TIST)* 12(1):1–19. <https://doi.org/10.1145/3418284>
- Yang M-H, Kriegman DJ, Ahuja N (2002) Detecting faces in images: a survey. *IEEE Trans Pattern Anal Mach Intell* 24(1):34–58
- Yu E, Ma J, Sun J, Chang X, Zhang H, Hauptmann AG (2022) Deep discrete cross-modal hashing with multiple supervision. *Neurocomputing* 486:215–224
- Yuan D, Chang X, Li Z, He Z (2022) Learning adaptive spatial-temporal context-aware correlation filters for UAV tracking. *ACM Trans Multimedia Comput Commun Appl (TOMM)* 18(3):1–18. <https://doi.org/10.1145/3486678>
- Zhang K, Zhang Z, Li Z, Qiao Y (2016) Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Process Lett* 23(10):1499–1503
- Zhang D, Yao L, Chen K, Wang S, Chang X, Liu Y (2020) Making sense of spatio-temporal preserving representations for EEG-based human intention recognition. *IEEE Trans Cybern* 50(7):3033–3044
- Zhou R, Chang X, Shi L, Shen YD, Yang Y, Nie F (2019) Person re-identification via multi-feature fusion with adaptive graph learning. *IEEE Trans Neural Netw Learn Syst* 31(5):1592–1601

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

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.