CSIT (December 2016) 4(2–4):165–171
DOI 10.1007/s40012-016-0111-2

SPECIAL ISSUE REDSET 2016 OF CSIT

Triah: an intelligent guiding system for the visually impaired

Vishal Thanvantri Vasudevan[1] · Srinidhi Sridharan[2] · Srividhya Swaminathan[2] · Sri Vishnu Kumar Karlapati[2] · Swathika Rengasamy[2]

Published online: 5 December 2016

Abstract Computer vision can be deemed as an emerging field which processes real world images in order to display a variety of numerical information about them. To improve the quality of the lifestyle that the visually impaired possess, we propose an assistive model which combines the various aspects of computer vision. Our proposed model aims at detecting the number of faces by using haar cascade classifiers and integrating it with Raspberry Pi. The processed images are run through the classifier and sum total is alerted through an auditory output via headphones. This model further boasts an average accuracy of 82.5%. Our model also notifies the visually impaired of the spatial orientation of the people surrounding them.

Keywords Computer vision · Haar features · Cascade classifiers · AdaBoost learning · Raspberry Pi

✉ Vishal Thanvantri Vasudevan
vishal13125@cse.ssn.edu.in

Srinidhi Sridharan
srinidhis@it.ssn.edu.in

Srividhya Swaminathan
srividhyas@it.ssn.edu.in

Sri Vishnu Kumar Karlapati
srivishnukumar@it.ssn.edu.in

Swathika Rengasamy
swathikar@ssn.edu.i

[1] Department of Computer Science and Engineering, SSN College of Engineering, Kalavakkam, Tamil Nadu, India

[2] Department of Information Technology, SSN College of Engineering, Kalavakkam, Tamil Nadu, India

1 Introduction

Humans are called Homo-Sapiens (man the wise) because our mental capabilities are extremely important to function effectively in our day to day lives. Computing devices, on the other hand, are discrete hardware systems with no such ability to think logically. Presently, the interaction between humans and computing devices has advanced to an extent where it has become a necessity. The technology is embedded with our daily schedules as it helps us work, communicate, manage time and, lay out business plans with ease. The popular notion prevails that the existing techniques in the domain of computing and display technologies sometimes serve as a bottleneck in effective utilization of the available information flow. To efficiently use them, the machines need to be taught like toddlers. This is where the concept of artificial intelligence comes into play. Artificial Intelligence is a science which trains the computers to exhibit intelligent behavior. This reduces human labor to a great extent and assists them in creating several paradigms. One major limiting factor to creating an intelligent system is modeling it to navigate through the dynamic environments. Hence, the field of computer vision provides this kind of information in an optimized manner and serves as a vision sensor. Computer vision is a methodology of acquiring, processing and analyzing the real world images to display some numeric information about them.

Computer vision has its applications in multiple disciplines and one such is the enhancement of the lives of visually impaired. It is immensely disheartening to see the visually impaired undergo a variety of challenges in their day to day activities. Each day in a visually impaired person's life is arduous without the use of technology. One such major challenge is the obvious fact that they find it hard to identify people present in their near surroundings.

A normal human being can perform face detection with minimal efforts and will also be able to distinguish the person standing near him and far away from him. In order to mimic this, we built an intelligent system which helps the visually impaired to navigate around publics spaces.

The rest of the paper is organized as follows. Literature Survey is presented in Sect. 2. A formal presentation of the proposed system is done in Sect. 3. Experimental Setup is discussed in Sect. 4. Section 5 presents details of the datasets, results and performance discussion. The conclusionis given in Sect. 6.

2 Literature survey

Face detection and recognition has been a subject of study and research for a long time. Over the years, there have been many different approaches to solving this problem with each newer algorithm trying to achieve better accuracy and real-time detection.

Cohn et al. [1] proposed an automated face analysis system using feature point tracking. Hierarchical algorithm for estimating optical flow was used to automatically track the facial features in digitized image sequences and the measurements were normalized for variations in position, scale, and orientation. They compared their results with manual FACS (Facial Action Coding System) and found that they had high concurrent validity. Sirovich and Kirby [2] used the concept of principal component analysis for characterization of human faces. Any face can be represented in terms of best coordinate system that are termed as eigen pictures. These are the eigen functions of the averaged covariance of the ensemble of faces. Turk and Pentland [3] achieved a major milestone by providing a near real-time system for face detection that can locate and track a subject's head. They have considered the face recognition problem as a 2-D recognition problem, taking advantage of the fact that faces are normally upright and can be described as a set of 2-D characteristic views. They projected the images over a features space which scanned the image for significant variations. The significant features are called "eigenface" and the system characterized an image as a face by calculating the weighted sum of eigenface features. These weights are then compared with known individuals.

Etemad and Chellappa [4] used Linear Discriminant Analysis of human faces in spatial and wavelet domains to evaluate significant visual information in different parts of the face to recognize a person. They were able to achieve very good accuracy for amidsized database with only four features. Viola and Jones [5] proposed a robust real-time face detection system using integral image for image representation, adboost learning for selecting the features and training the classifier and attentional cascades to eliminate non-face regions immediately. This is a benchmark algorithm and has been used by many people for different applications. This algorithm has a slow learning time but can do very fast real-time feature computations for detecting faces.

There have been many intelligent applications that have been developed with the help of raspberry pi. One such system used raspberry pi, for detecting vehicles and extracting its license plate numbers which were compared against existing database and based on the result authorized people to enter a gate [6]. They used canny edge detectors fro extracting license plate numbers. A vision-based hand gesture recognition system was proposed which used haar features and AdaBoost learning for posture recognition and context-free grammar-based syntactic analysis for gesture recognition [7].

3 Proposed system

In this paper, we propose a system that detects the faces of the people and gives out the number of people and where they are located spatially as the output. Drawing from Viola-Jones face detection framework, we have used Haar features, Integral images, AdaBoost learning and Cascade classifiers. Further, we have written the algorithm to give the spatial location of the people in the frame. The output is read out to the user using text-to-speech converter.

3.1 Haar features

Viola-Jones [5] used features that were reminiscent of the haar basis function which is a mathematical function that produces a square wave output. So, they are just rectangular patterns in data. The original Viola–jones implementation didn't have many different features. A scale, say 24×24 is picked, features of that scale are slid across the image. Then, average pixel value under black and white areas are computed. If the difference between the two areas is greater than some threshold then the features match. Black and white features don't indicate that one is greater than the other. It's kept to show the difference between the two. The same feature can match to abunch of other things in an image. But, it has been found that a feature based system works much faster than a pixel based system. Features can encode data that is usually difficult to learn.

3.2 Integral image

This is an image representation technique that was used to compute the features rapidly. It is similar to the Summed Area Table in Computer Graphics. An integral Image pixel is the sum of the pixel values above and to the left of (x, y).

$$ii(x, y) = \sum_{x' < x, y' < y} i(x', y'). \quad (1)$$

where ii(x, y) is the integral image and i(x, y) is the original image. The recurrences

$$s(x, y) = s(x - 1, y) + i(x, y). \quad (2)$$

$$ii(x, y) = ii(x, y - 1) + s(x, y). \quad (3)$$

are used to compute the integral image from the original image in one pass. Here, s(x, y) is the cumulative row sum. So, with 3 operations, the integral image can be computed for every pixel in the image. Thus, integral image calculation happens in linear time. It is just like integrating over 2-dimensional space.

3.3 AdaBoost learning

Viola-Jones [5] variant of AdaBoost learning algorithm which was used to select the feature as well as train the classifier. For a 24 × 24 detector there are more than 160,000 features, and checking all of them for all images is not possible. So AdaBoost selects a set of important features called weak classifiers and combines them linearly to form a strong classifier. These weak classifiers must work better than a random guess, i.e. they should at least be a little more than 50% correct. We train the classifier by feeding it a set of positive and negative images. Initially, all images are given same weight and features are slid across them and computed. After this, a number of misclassified images is taken into account and they are added to the error. Thus, the weighted error for each feature is calculated. The one with the least error is chosen as the best feature and for the next time the weights of the misclassified images are increased, and the weights of the correctly classified images are decreased and the features are again computed to find another weak classifier. So we try to reduce the number of misclassifications by focusing on them. This process goes on till the number of features to build the strong classifier is obtained. Every time the weights are normalized so as to sum up to 1. The final strong classifier is computed as the linear combination of these classifiers where the weight of each classifier is directly proportional to its accuracy. The strong classifier is given by,

$$c(x) = \begin{cases} 1, & \sum_{t=1}^{T} \alpha_t h_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0, & otherwise \end{cases}. \quad (4)$$

where α_t is the weight in which is $\alpha_t = \log \frac{1}{\beta_t}$, where $\beta_t = \frac{\in_t}{1 - \in_t}$ and $\in_t$ is the weighted error.

3.4 Cascading

To immediately eliminate non-face region and focus only on the face like regions cascade classifiers are used. So it is called attentional cascades. Combining increasing complex classifiers will increase performance drastically. A key measure is the false positive rate of the attentional process. Filtering out 50% of the image and preserving 99% of the faces is what the face detection attentional operator tries to do.

A single strong classifier formed by the linear combination of all the best features is not very efficient because it would lead to a very high computation cost. A cascade classifier has a number of stages each comprising of strong classifiers. So all the features were grouped into a number of stages and each stage had a certain number of feature. Each stage decides if the given sub-window is a face or not a face. To design a cascade, we must consider,

- Number of stages in the cascade
- Number of features in each strong classifier
- Thresholding of each strong classifier (Fig. 1).

Each stage discards the images that do not have faces and passes to the next stage only those sub-windows that might have a face.

3.5 Spatial detection

To find the spatial location of the detected face, the entire video is divided into a 3 × 3 grid of equal widths and equal heights. When the face is detected, the height and width of the face and the beginning coordinates of the face are found from the Haar Cascade Classifiers. The middle of the face is

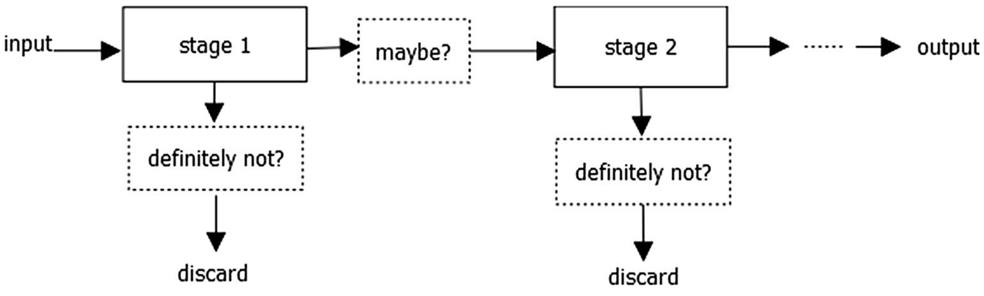


Fig. 1 Cascade classifier structure

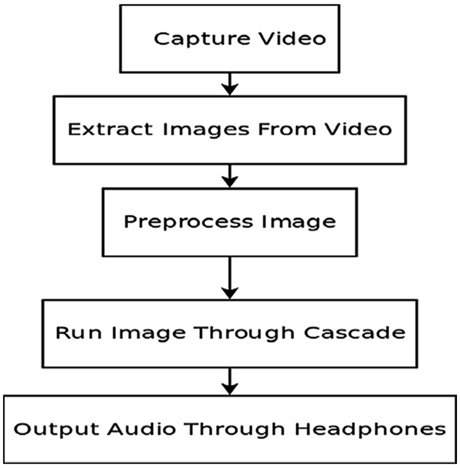

Capture Video
Extract Images From Video
Preprocess Image
Run Image Through Cascade
Output Audio Through Headphones

Fig. 2 System flow

found, by taking the mean of the start and end points of the width and height respectively. The location of this point in the grid is found. Based on the grid in which the point lies, its spatial location is found. This spatial location is told the person via headphones, using the Text-to-Speech Convertor.

4 Experimental setup

The Setup of the proposed system and its description is as follows. The device is developed using components that capture video, extracts an image from video, process the image and provide output in the form of audio to the user.

The components that are used in the system is as follows (Fig. 2):

- Raspberry Pi 3
- Raspberry Pi camera module
- Power source
- Headphones

4.1 Raspberry Pi 3

The Raspberry Pi is a series of credit card-sized single board computers which include ARM compatible CPU and anon-chip GPU. Secure Digital SD cards are used to store the operating system. The models support a Linux based operating system and promote python as the main programming language.

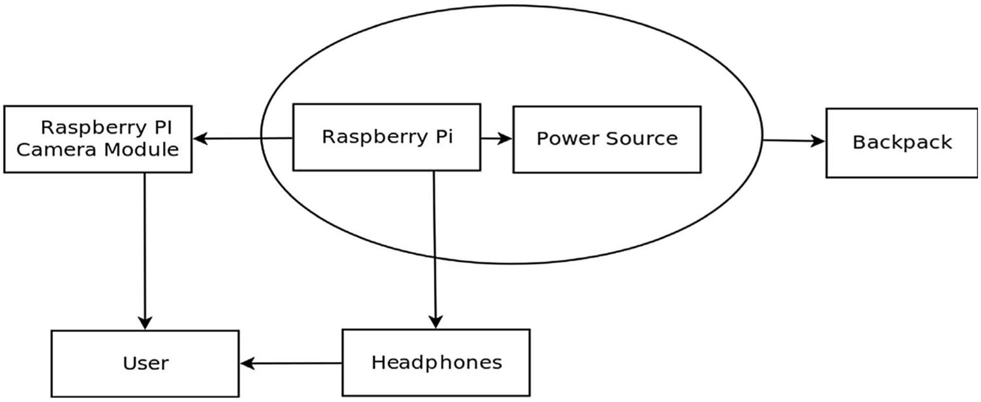
4.2 Raspberry Pi camera module

The Raspberry Pi camera module can be used to take high definition video as well as still photographs. It can be interfaced with the Raspberry Pi using a 15 cm ribbon cable and connecting it to the CSI port on the board. The camera resolution is 5 MP and supports 1080p30, 720p60 and 640 × 480p60/90 video modes as well. The picture formats supported are JPEG, JPEG + RAW, GIF, BMP, PNG, YUV420, RGB888. It supports raw h.264 video format.

4.3 Setup

The Raspberry Pi camera module is attached to the middle of a spectacle frame. The camera module is interfaced with the board. The Raspberry Pi board is placed inside a backpack worn by the user. The SD card on the board is used to hold the images and the operating system needed to process the data and produce the output. The Battery pack is connected to the board at all times and is placed along with it in the backpack. The headphones are connected to the audio port on the Raspberry Pi and extend from the backpack to the user's ears. The camera is configured to constantly take video. The Headphones provide constant output to the user as and when new images are captured (Fig. 3).

Fig. 3 System architecture



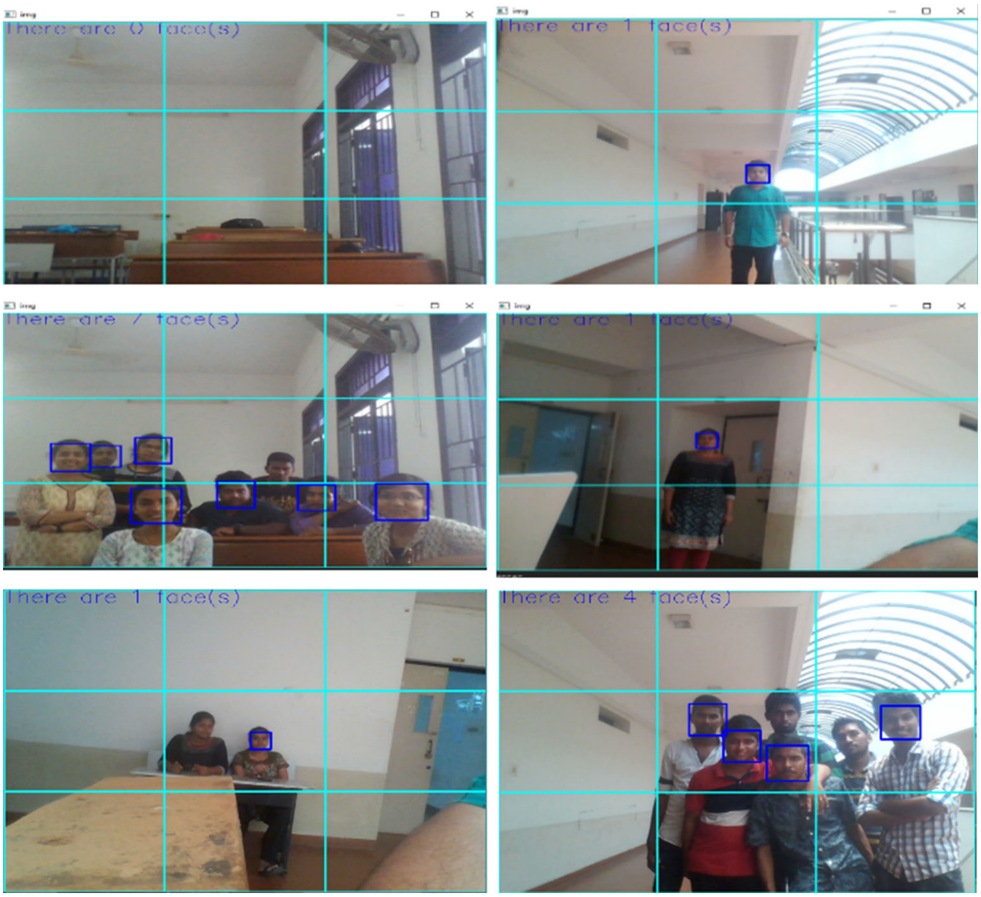


Fig. 4 Raw images (From the top *left*, 9.png, 14.png, 8.png, 15.png, 16.png, 13.png)

Table 1 Experiment results

Image name	No. of people in the image	No. of people detected	The distance of the first person from the camera in cm	No. of spatially correct detections	Orientation	Lighting
13	6	4	180	4	Straight	Bright
14	1	1	340	1	Straight	Bright
15	1	1	290	1	Straight	Dull
16	2	1	340	1	Straight	Dull
8	8	7	160	7	Straight	Varied
9	0	0	500	0	Nil	Varied

5 Results and analysis

The following results were generated using the proposed system. The results were taken at multiple locations, with bright, dull and varied lighting. The number of people in the video were also varied. Screenshots of the live video have been provided below for the reader's reference (Fig. 4; Table 1).

It can be seen from the table and the plots that as the distance of the person from the camera increases the accuracy of detection decreases. We found that, for a range of up to 3.5 m, the faces were correctly detected, at points further away, accuracy decreased drastically. Accuracy of face detection is calculated as follows (Figs. 5, 6, 7, 8):

$$\textit{Accuracy of face detection} = \frac{\textit{Number of people correctly detected} \times 100}{\textit{Number of people in the video at that point of time}} \quad (5)$$

It is also observed that if the person in less than 0.75 m away from the camera, the face isn't detected. However, this isn't a problem, as most people are more than 0.75 m away during day to day interaction. It can be seen that

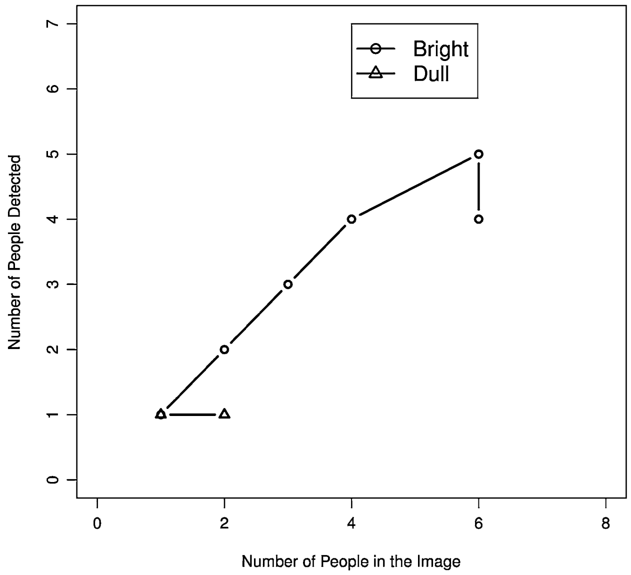


Fig. 5 No. of people (vs.) no. detected (bright and dull lighting)

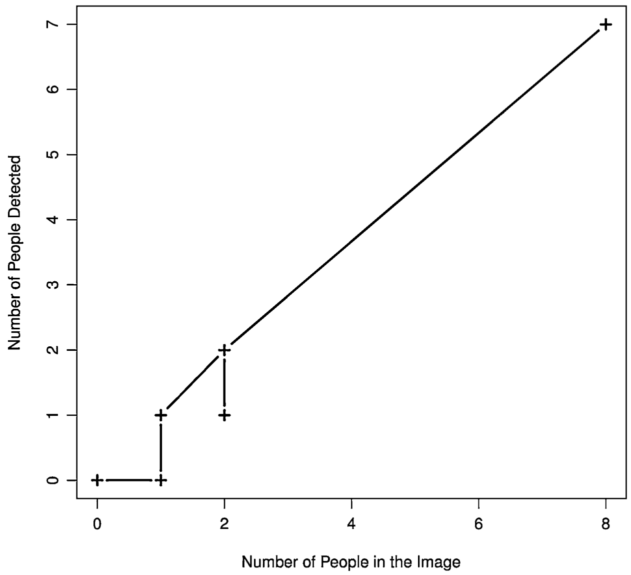


Fig. 6 No. of people (vs.) no. detected (varied lighting)

under dull lighting, the accuracy decreases even if the people are within the prescribed range. It is seen that under bright lighting people are detected with high accuracy. Even under varied lighting the same trend is observed. It can be seen from the table that, the orientation of the face matter while detection. If the face of the person in the video is turned to the side, then, it is not detected, however, faces are detected with a small turn radius, as long as the entire face is still shown to the camera. It is seen that the spatial locations of the detected faces in the video have been found with 100% accuracy. Accuracy of spatial detection is calculated as follows:

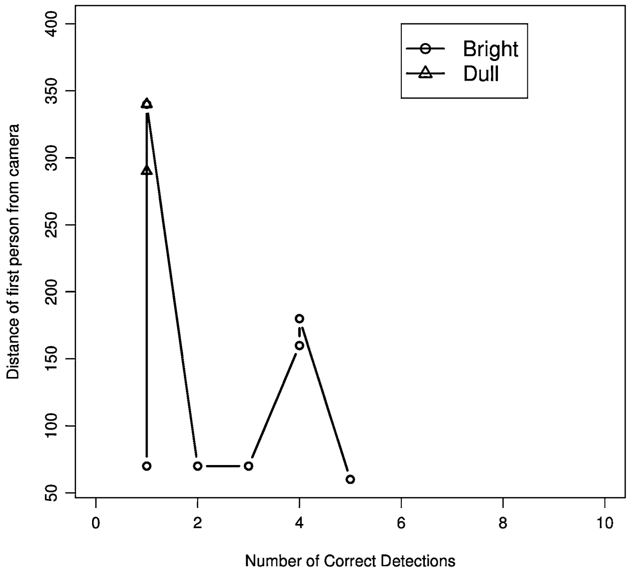


Fig. 7 No. of correct detections (vs.) distance of the first person from the camera (bright and dull lighting)

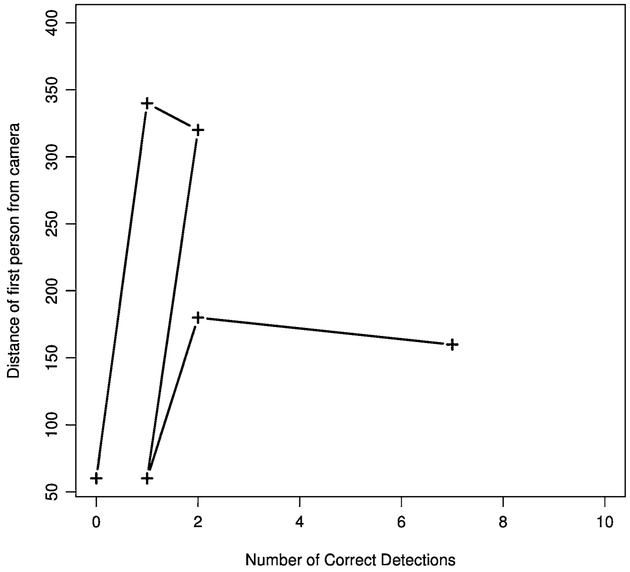


Fig. 8 No. of correct detections (vs.) distance of the first person from the camera (varied lighting)

$$\begin{aligned}&\mathit{Accuracy\ of\ spatial\ detection}\ (\%)\\&\quad=\frac{\mathit{Number\ of\ spatially\ correct\ detections}\ \times\ 100}{\mathit{Number\ of\ people\ correctly\ detected}}\end{aligned}\tag{6}$$

6 Conclusion

The developed intelligent system is capable of successfully identifying faces with a fairly good accuracy. From our experiments, we observe that the system achieves an average accuracy of 82.5%. Moreover, it achieves an accuracy of 92.75% under bright light, 75% under dull light and around 89.58% under varied light. Every day, new algorithms are discovered to solve various problems of computer vision. The proposed system can be improved to accommodate more features such as gender detection and face recognition. The robustness and performance of the system can be improved by using Convolutional Neural Networks instead of Cascades to achieve better results. The system can be made more portable by using wireless components for capturing the images and also playing the audio output.

References

1. Cohn J, Zlochower A, Lien J, Kanade T (1999) Automated face analysis by feature point tracking has high concurrent validity with manual FACS coding. Psychophysiology 36:35–43

2. Sirovich L, Kirby M (1987) Low-dimensional procedure for the characterization of human faces. J Opt Soc Am A 4:519

3 Turk A, Pentland A (1991) Eigenfaces for recognition. J Cognit Neurosci 3:71

4 Etemad K, Chellappa R (1997) Discriminant analysis for recognition of human face images. J Opt Soc Am A 14:1724

5 Viola P, Jones M (2004) Robust real-time face detection. Int J Comput Vision 57:137–154

6 Sasi Kala KS, Ahmed S (2016) Implementation of number plate extraction for security system using raspberry Pi processor. IJERT. 5:03

7 Chen Q et al (2007) Real-time vision-based hand gesture recognition using Haar-like features. In: 2007 IEEE instrumentation and measurement technology conference IMTC 2007