EXPERIMENTAL ANALYSIS OF CAMERA CALIBRATION TECHNIQUES USED FOR EYE TRACKING

Zeenat AlKassim and Qurban Memon
EE Department, College of Engineering, UAE University
Al-Ain, United Arab Emirates
201370466@uaeu.ac.ae; qurban.memon@uaeu.ac.ae

ABSTRACT. This paper discusses the calibration phase required prior to working of any gesture recognition or eye tracking devices. The calibration phase is an starting phase of eye tracking system. Prior to tracking phase, calibration (for eye detection) is required. Four methods have been tried in this paper, namely: using a box, multiple frames, laser and neural networks. The results of implementing each of these methods are compared to conclude the optimum solution to lead to eye detection and tracking.

Keywords: camera calibration, eye tracking, neural network, human-computer interaction

INTRODUCTION

Human detection and tracking is gaining more importance each day. This importance is huge mainly because of wide variety of applications that can be built. One such problem is building ways to interact with devices for the people with physical difficulties. Eye detection and tracking can be used to build such systems that operate by detecting and tracking eye movements without the need to use hands. Typically, building an eye tracking system consists of several consecutive stages or phases. Calibration phase is the first phase of designing an eye tracking device. This phase consists of training the system to recognize the eyes in a face prior to tracking them. This is the case in many devices like touch screens, smart boards and body tracking, in which a calibration stage is required first before the devices can work efficiently.

In literature, many techniques can be found that deal with detecting the face and then the eyes, while a few others detect eyes directly without the need to detect the face. As in [1], different methods exist for face detection like knowledge-based methods, feature invariant methods, template matching methods and appearance based methods. The first three depend on certain selected features in a human face for it detection. The appearance based method deals with learning how a human face looks like and the possible variability in faces. Thus, in (Daniel, et al, 2011), face is detected first and then the eyes are detected. The main technique implemented is skin colour segmentation with the help of a look up table for skin colour pixel values.

While skin colour segmentation approaches are easy to implement, a major drawback in using such a technique is the presence of the same skin colour objects in the surroundings. Designing a calibration phase that depends solely on skin colour segmentation has limitation of detecting any object in the skin color range as a face.

The authors (Zia, et al, 2014), skin colour segmentation has also been implemented for face detection. However, the colour space used was 'Lab' - *L* in the 'Lab' colour space resembles Lightness, while *a* and *b* resemble colour components of the face image. The *L* effect is removed to counteract the effect of lightness in detection of skin colour. a and b components are used for detecting the face. Here, a look up table is not used. Instead, that pixel colour region with the largest area in the image is regarded as the face. After the skin colour pixels are recognised and the face is detected, circular Hough transform is implemented for direct detection of the eye's iris. While the methods used in [2] are easy to implement, once again skin colour segmentation techniques cannot be fully efficient due to the possibility of existence of other skin similar colours in the background. Also, each person's eyes are different in size and shape. Circular Hough Transform may fail to detect the circular iris in some cases.

In another work (Praglin, 2014), a different method for eye detection is implemented in MATLAB, which trains a MAP detector to detect the iris based on pixel colours. The threshold set for the MAP detector was made low since there are different colours of the iris. After eye detection, face was detected by computing the largest centred area with pixel colour values close to the mean of that area. Thus, this approach combines colour segmentation with morphological operations for detection of the eyes and the face.

The works discussed above present novel methods for detection of face and eyes, unlike other works that implement known concepts like Viola Jones' face detection method and Haar-like features terminology.

CALIBRATION APPROACHES

Calibration phase is very important for proper synchronization between a user and the system. The more accurate the calibration phase, the better the working of the system and tracking done by it. While in some devices calibration is required whenever device is powered on, in other devices only a single calibration at first-time-use is required.

As discussed in the previous section, some techniques involve detection of face first for eye detection, while other techniques detect the eyes directly without the need for face detection. Following four approaches are introduced. All those methods require the user to calibrate his/her eyes only one time. This was purposely intended since the aim is to design systems for people with physical disabilities. Devices designed for users with disabilities are usually used by a single person, unlike those designed for public use like the computers in universities and other public places.

Capturing Multiple Frames: Here, the idea is to detect the eyes directly without the need to detect the face. A person blinks her/his eyes naturally. This quick movement of the eyes is utilized to detect the user, assuming that the background is stationary. Since the system is designed for indoor use and specifically for users with physical difficulties, the surrounding environment will most probably be stationary. Even if there is a movement, the system focuses on the movements in the middle centre of the screen (where the user is most probably seated).

This method involves the user looking at the camera and blink once, during which multiple frames are captured. By comparing multiple captured frames, the system will be able to detect the eyes of the user which are under movement with the help of simple image processing techniques. From the detected eyes, certain features can be extracted. And finally, using these features, the entire face of the user is detected. Thus, this approach involves detecting the eyes directly in one step.

Bounding Box: This approach also detects the eyes directly without the need to detect the face prior to eye detection. The aim of the calibration is to detect the user's eyes accurately and locate its position. As such, in this approach, bounding box visually appears on the system screen in front of the user. The user is required to position him/her in such a way that the two eyes are positioned inside this box. This, this approach is named 'bounding' box. This method is very simple and works efficiently. The user is only required to blink once while positioning the eyes inside the bounding box that appears on the system screen in front of her/him. Since the position and size of the bounding box is already known, the position of the user's eyes can be known and extracted. This method is flexible and the size of the bounding box can be chosen such that it allows the user to use the system from a range of distances measured from the system screen.

A couple of eye tracking systems implement calibration by presenting a number of successive dots at different locations on the screen and the user needs to stare at each dot. This helps the system to detect and track the eyes by visualising the reflection of light on the cornea of the user's eyes when viewing dots at different locations. Such methods have been implemented in UATP, 2013, MyGaze, 2013.

Line Laser: Laser has been implemented in a number of systems because of its unique features, one of which is its brightness on reflection. This feature is exploited in laser keyboards, as in (Celluon, 2012). In laser keyboards, the reflection of the line laser onto the typing finger is captured by the camera and indicates that a key has been pressed. Similarly, in this calibration method, a line laser (a point laser can be an alternative) helps in detection of the user in front of the system. This method depends on the fact that a user is the nearest object to the system screen. Thus, when a line laser is directed horizontally from the screen and projected forwards in front direction, the reflection of the line laser is expected to be brightest on the nearest object, which is the user. Therefore, this reflection onto the user's face is captured by the camera, after which the user's face can be identified.

Unlike the previous two methods, this method involves recognition of the user's face and then the eyes. This method is similar to the physical methods implemented for eye and gesture detection. This is because the main working of this technique is physical (through the use of laser). Many physical approaches have proved successful in human computer interaction like the use of electrodes for detecting eye movements, the use of hand gloves to detect hand movements and the use of coloured markers on hand fingers in the Sixth Sense Technology (Pranav, 2009).

Artificial Neural Networks: Artificial Neural Networks (ANN) is a concept inspired from central nervous system of a human body that consists of neurons in a similar structure to a network. Inputs in the network are multiplied by weights and a mathematical another function computes the output of the neuron (Gerhenson, 2013). Weights can be adjusted to obtain wanted output for specific inputs. The weights are usually adjusted by learning or training.

The challenging aspect is how to choose the training image samples, the training parameters as well as choosing the best architecture of the neural network. However, once trained, the system can directly detect faces based on certain facial features it has been trained to detect. This method is different from the previous methods since the training of the system happens prior to using the system. However, other methods involve detection of the user on-the-spot without prior training.

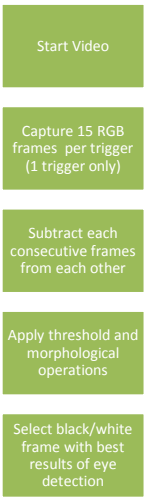
A couple of different types of networks with different number of hidden neurons can be trained in order to find the network with best results. Both data fitting and pattern recognition networks can be trained. In data fitting networks, the aim is to train the network such that it can output the exact position of the eyes once the user is in front of the screen. Pattern recog-

nition networks can also be trained such that it recognises the pattern of the face by recognising dark spots in the face and outputting a 'yes'. Though this method of implementing neural networks is longer and time-consuming during training, it is time saving for implementation after being trained (Memon, 2001; Memon, 2003). This is because while other methods involve processing the input image for every new user in order to detect the eyes, neural networks are trained only once and can then be used for detection of multiple users without the need to re-train. In the next section, all these approaches are built and tested for calibration purposes.

BUILDING AND TESTING

This section explains each of the four approaches discussed in previous section for eye detection; and presents testing results. In all methods, images were captured by a Logitech webcam C110, with a capacity of 30 frames per second, at a resolution of 640 by 480 pixels.

Capturing Multiple Frames Method: This method is software implemented, where the user is made to stare at the screen and blink naturally for around 0.5 s. As seen in Fig. 1(a), this method mainly involves subtraction between frames, processing the frames and selection of the best subtraction results. Best subtraction results mean that the subtraction took place between one completely closed eye frame and one completely opened eye frame. As such, the subtraction results show the eyes shape and size correctly. The final result is a black and white image with the subtraction results (eyes) in white pixels and the remaining picture in black pixels. The Figure 1(b) shows sample images after processing and how the best frame is selected. The best frame selected is frame 10 with maximum white pixels representing the eyes accurately. Frame 7 shows the interpretation of extra white pixels as the eyes.



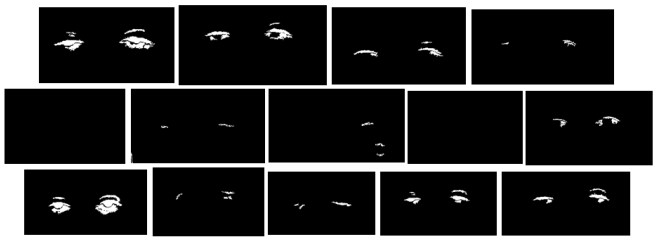


Figure 1(a). Calibration By Multiple Frames **Figure 1(b). Captured Frames After Processing (Frames 1-14 From Top Right)**

Results of this method are quite satisfactory. However, the existence of multiple frames is cumbersome. Testing was done multiple times on the same individual. On testing, results showed that sometimes system fails to select the best image. The existence of maximum number of white pixels in some images might not indicate the position of the eyes in the image, as is the case in frame 7 of Fig. 1(b). On the other hand, when the best frame is correctly selected, the results are accurate and exact position of the eyes in the image is located.

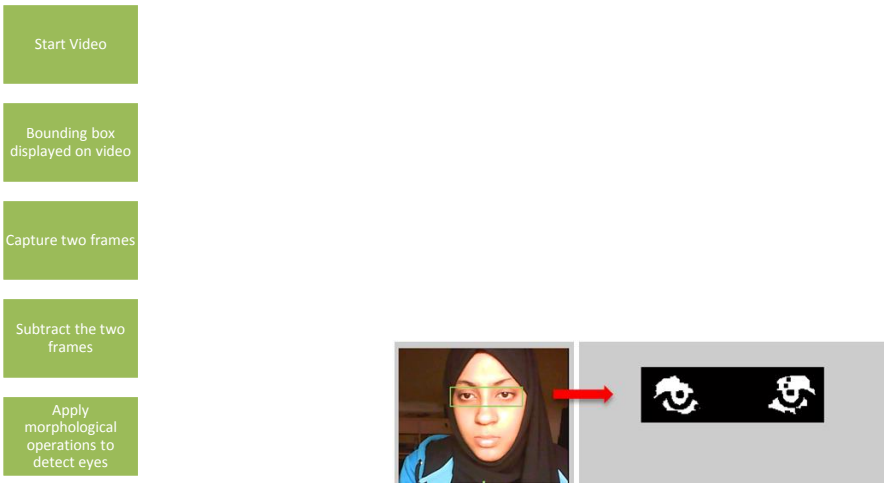


Figure 2(a). Bounding Box Calibration (b) Bounding Box Implementation (First and Final Images)

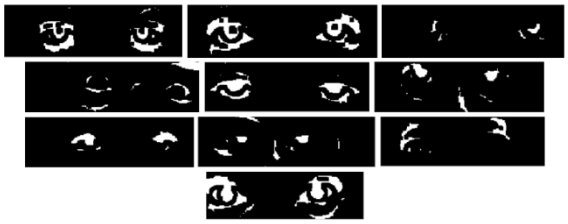


Figure 3. Bounding Box Testing Results

Bounding Box: The bounding box method is very easy in implementation when compared to the multiple frames. It is a straight forward technique for eye calibration that displays a box in front of the user, requires the user to position herself/himself such that the eyes fall inside that bounding box and instructs the user to perform two actions: 'open your eyes' and 'close your eyes'. With two actions, the eyes action can be precisely extracted since the location and size of the bounding box are already known. Thus, since the concentration is only on the bounding box area, there are no mistakes of other white pixels that are mistaken as eyes as is the case in method 1. Another main difference of this method over the previous one is that it does not deal with multiple frames, only two frames. The steps of this approach are as in Fig. 2(a). For synchronisation, the user is directed by the system when to open and close eyes. A pause of 5 seconds is added before each frame is captured to allow user to adjust face position with respect to the bounding box and respond at ease to sound file played before frame capturing. This will ensures accurate results, as in Fig. 2(b).

The dimensions of the bounding box have been decided to be of width 213 pixels and height 63. This size of the box has been chosen by trial-and-error method in such a way that will provide freedom to the user to be seated in any distance from 23.75 inches (60 cm) to 71.25 inches (180 cm) from the computer screen. A distance of less than 23.75 inches might be so close that the eyes fall outside the bounding box, while beyond 71.25 inches from the computer screen might be so far in such a way that the closing and opening of the eyes are not properly detected by the program. Both cases might lead to improper detection of the eyes. According to Apple Inc., a distance of 18 – 24 inches is the comfortable zone to place the computer screen away from the eyes. This means that a distance of 23.75 – 71.25 inches is a good distance to avoid eye discomfort of the user.

Testing was done on 10 different individuals, both male and female, adults and kids. Sample results shown in Fig. 3 proved efficiency of the method and ease of implementation. In one case, the eyes were not perfectly detected due to uneven lighting inside the room. Overall, testing results were successful by 96 % (24 successful attempts out of 25). One attempt failed to detect the eyes because of poor quality images taken by camera under very dim light.

Line Laser: In this method, hardware setup is required. A line laser has been built powered on a board and placed under the webcam at a horizontal position to the user. In this set up, it is projected at the lower part of the user's face. For calibration, the laser is switched on and calibration starts. The laser line spans horizontally the objects in front of the screen. Since the user is the closest object to the screen, the reflection of the laser on the user's face is the brightest. Thus, the brightest reflection is recognised by the camera. For this method, only one image is taken by the camera. A few processing steps are done, and the region on which the laser reflection is brightest is detected as the face. As in Fig. 4(a)., few steps are involved in this method.

While this method requires an additional hardware set up (setting the line laser), it is advantageous in the fact that it requires only one frame. From this one frame, the face can be directly detected after a few processing of the image. Fig. 4(b) shows the implementation of this method. Testing on one sample showed good results as long as the user is the nearest object to the screen and the laser is mounted in a proper horizontal position and distance from the user. As noticed, this method involves face detection prior to eye detection unlike previous method where eyes were detected directly without the need for face detection.

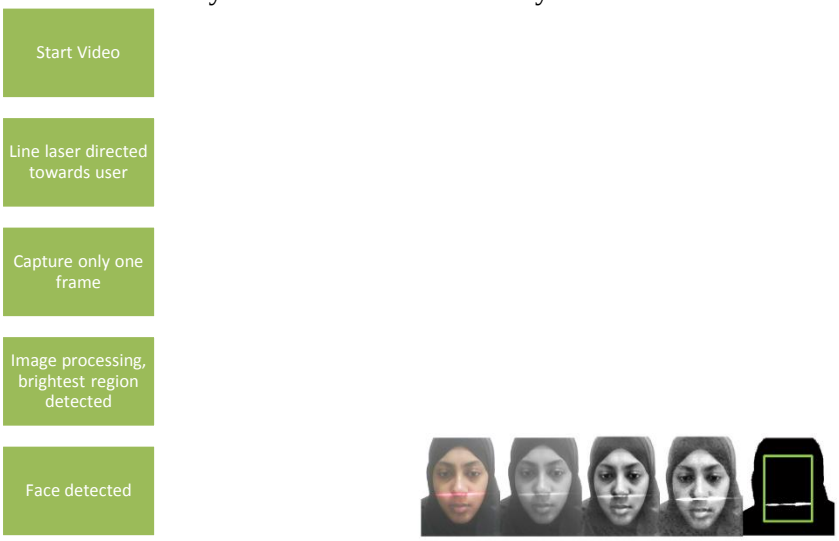


Figure 4(a). Line Laser Implementation **(B) Brightest Laser Reflection in Green Box is Detected as the User's Face**

Artificial Neural Networks: In this method, a data fitting neural network is trained to automatically detect faces without the appearance of a bounding box on the recording video in front of the user during calibration. A neural network is trained to detect a face based on common dark spots in any human face. The advantage of this method over the previous bounding box method is that once the neural network is trained, it can automatically detect any face sitting in front of the camera. Thus, training is required only once.

First, processing of training image samples is done. Images of the people's faces (training samples) are captured in RGB form at a resolution of 640-by-480 pixels. Next step is to convert the RGB images into grey scale. Third step is to increase the contrast of the sample face images by adaptive histogram equalization of all channels (Fig. 5a).

Last step in processing images before inserting them as training input into the neural networks is to resize images. The images are first cropped to a size of 368-by-336. By cropping, the shape of the images is changed from a rectangle 640-by-480 to a square 368-by-336. With square images (width and height are almost equal, with a ratio of 368/336 = 1.09), width and height of the images can be rescaled to the same number without distorting the images. The images are then rescaled to 25-by-25 resolution for simplicity. Thus, prior to training the neural network, each image undergoes a set of steps. Each processed image is inserted into the network as a training sample input. Thus, for each 25-by-25 image, 625 values of the grey scale image are inserted row wise as shown in Figure 5(c).

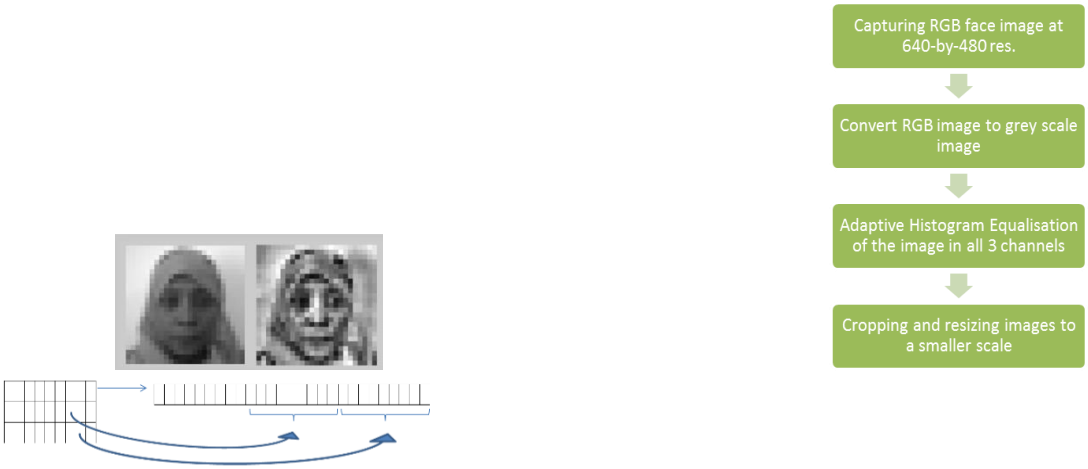


Figure 5. (a) Processing of Input Images (b) Processing Steps of Input Training Images (c) Row Wise Insertion of Image Greyscale Pixel Values

The greyscale pixel values are normalized by division with 255. The training samples consisted of faces of both male and female, adults (aged 20-50) and children (aged 5-8). Thus, the total input file for training the network is a 34×625 matrix consisting of 34 samples of 625 elements each. Participants were made to close and open eyes for capturing two images of each participant. The network is trained by supervision known as supervised training. As such, the desired output will be provided to the network. The desired output values for each input image are four values that indicate the position of the nose with respect to the eyes (marked as red boxes in Fig. 6(a).

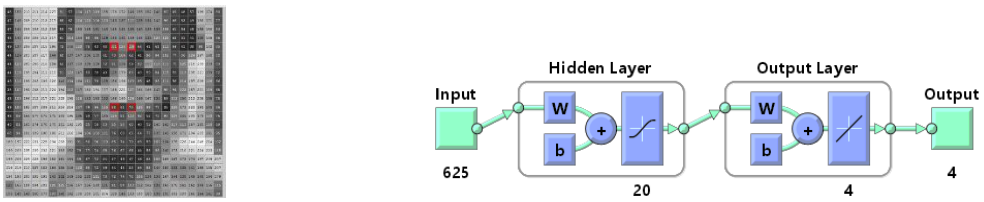


Figure 6(a). Desired Pixel Positions (Desired Output) (b) Neural Network Architecture

Since the aim is to extract the position of the pixels and not the value, the output has been converted from grey scale value to the pixel position by the following formula:

$$Output\ value = [(row\ number - 1) \times 25] + column\ number;$$

Finally, the output values are normalised by division with 625 (highest output position possible is the last row number). Thus, the output file is 34×4 matrix consisting of 34 samples of 4 elements each. The neural network used is a two layer (one hidden layer and one output layer) feed-forward network trained with the Levenberg-Marquardt and scaled conjugate gradient back propagation algorithm. While 34 samples were inserted for training, 27 (80%) samples were used for training and 5 (15%) used for validation and 2 (5%) used for testing. The hidden layer consists of 20 neurons. The hidden layer works with a sigmoid function and the output layer implements a linear function, as shown in Fig. 6(b).

Training the network was done for 1030 iterations, after which a minimum mean square error (MSE) of 0.00004632702398146551 was reached. The output values produced by the trained network are very close to the desired output values with very small error values. On testing the network by a random test image captured by webcam, it was found that while an error of only 0.00224493 was produced, and the output pixel positions are close to the desired output, the level of accuracy is significantly enough. Thus, the network outputs quite accurate row numbers to the desired pixel positions.

Table 1. ANN Testing Results

Comparison (MSE=0.00224493)							
Desired Output				Network Output			
0.2224	0.2256	0.5424	0.5456	0.273832	0.276808	0.571432	0.574408

COMPARING DIFFERENT APPROACHES

Different methods of first time eye detection were tried in order to come up with the best approach by comparing the different methods.

Method 1: This method captures multiple frames and chooses best image based on number of white pixels.

Pros: No hardware set up is required. It is easy method for the user since user is only required to stare at screen or camera and blink naturally. The user might stare at any other location as well as long as the head is not tilted to a big degree. Thus, it is practical to the user (here, focus is on disabled person). If best image is picked out of all other images, accurate position of the eyes can be extracted.

Cons: System might fail to pick out the best image showing exact eyes. This is because other white pixels might be left out after image processing and interpreted as the eyes. In this case, wrong eyes will be detected. Also, this method involves many frames to loop over to select the best frame, and consumes more system memory.

Method 2: The second approach is the bounding box.

Pros: No hardware is required, only software. It is a very simple and straight forward approach that captures only two frames, one of closed eyes and the other of opened eyes, from which the eyes position is extracted. Also, eyes position is extracted accurately with no minimum errors on testing.

Cons: The user needs to stare at camera; slight movement might produce wrong eyes position.

Method 3: *This approach is the use of line laser for eyes detection.*

Pros: This method is so fast in implementation. Only one frame is captured to detect the face. Also, user can be positioned at any angle and the head can be freely tilted since this method depends on concept of distance from the line laser and reflection from the closest object. There are no complications of capturing multiple frames.

Cons: This method requires software and hardware setup, thus not practical in terms of set up. The line laser needs to be horizontally positioned or the face will be detected but measurements will be wrong. Under extremely bright environments, the line laser might not be completely visible. Other objects placed close to the screen (and hence laser) might be wrongly detected as the face; face is detected prior to the eyes unlike previous methods in which eyes were detected directly - line laser cannot be directed directly on the user's eyes.

Method 4: *This approach is the use of Neural Networks for face detection.*

Pros: While other approaches require calibration whenever a new user is introduced, this method involves training the network only once. After training, the network can be used for any new user without the need to re-train. Thus, it is practical and time saving after training.

Cons: Training the network, choosing the network architecture and training parameters is more complicated and time consuming compared to the processing in previous methods. With 34 face samples for training and 1030 training iterations, the output produced was accurate enough to locate the exact eyes position on the face.

These comparative findings were tabulated and are shown in Table 2.

Table 2. Comparing the Calibration Approaches

Feature	Calibration Methods			
	1	2	3	4
Speed	15.35628 seconds	19.674946 seconds	2.536435 seconds	1030 iterations of training
Software req.	Yes	Yes	Yes	Yes
Hardware req.	No	No	Yes	No
No. of Frames	Multiple (15)	Two	One	34 training images
No. of times running code	For every new user	For every new user	For every new user	Only once during training
Accuracy of Results	Accurate when best frame selected	Very accurate	Accurate if user is closest object to screen	Row number accurate; Column number not very accurate

CONCLUSION

Based on results, it is shown that each method has some advantages and disadvantages over the other. However, better results could be produced using neural network if the training samples are increased or if the training samples are further cropped so that only the face shows in the training samples. The comparison also shows how different implementations in both software and hardware can be conducted to come up with best results for first-time eye

calibration. Other approaches can also be explored such as support vector machines (SVM), and compared with methods tested in this paper

REFERENCES

Celluon (2012). Magic Cube. Retrieved from http://www.celluon.com/products_mc_technology.php

Daniel, P., Cristian, F., Avila, M., & Felix, M. (2011, October). Algorithm for face and eye detection using colour segmentation and invariant features. *Proceedings of the 34th Telecommunications and Signal Processing*, 564-569.

Gershenson, C. (2003). *Artificial neural networks for beginners*. arXiv preprint cs/0308031.

Memon, Q, (n.d). Camera calibration and three-dimensional world reconstruction of stereo-vision using neural networks, *International Journal of Systems Science*, 32 (9), 1155-1159.

Memon, Q, & Shuja, M. (2003). Crime investigation and analysis using neural nets, *IEEE Multi Topic Conference*, 346-350

Praglin, M., & Tan, B. (2014). Eye Detection and Gaze Estimation. Eye, 1.

Pranav Mistry (2009). *SixthSense. Fluid Interfaces Group*, MIT Media Lab. http://www.pranavmistry.com/projects/sixthsense/

The Gamers' Charity (2013). The €500 'MyGaze' Gaze Tracker. Retrieved online January 2015 from http://www.specialeffect.org.uk/

Utah Assistive Technology Program (2013). Tobii ATI eye-tracking technology & how it works for computer access & communication. Retrieved online January 2015 from https://www.youtube.com/watch?v=m_QfUIIs7Fg

Zia, M. A., Ansari, U., Jamil, M., Gillani, O., & Ayaz, Y. (2014), Face and eye detection in images using skin color segmentation and circular Hough transform, *IEEE International Conference on Robotics and Emerging Allied Technologies in Engineering*, 211-213.