### GAZE ESTIMATION FOR MEASURING & Assessing EFFECTIVENESS OF ADVERTISEMENTS

*Ambika Verma Pablo Caballero*

University of Texas at Austin

#### Abstract

Gaze estimation is used comprehensively across industries to map visual gaze of the viewer and derive intriguing results based on the same. This technology has applications ranging from psychological studies to marketing analysis. In majority of the settings invasive or semi-invasive techniques are used to determine the viewer’s fixation, such as Head mounts and Infrared light emitters and sensors, which also end up being expensive.

Our aim is to design a gross gaze estimator using readily available low resolution webcams to map gaze of the viewer while watching advertisements (print and video) on a 1280x1024 resolution screen. The results obtained can be used to analyze the effectiveness of tools used in advertisements to capture the viewer’s interest as well as improve upon them.

**Index Terms—** Gaze Estimation, Face detection, Eye detection, Iris Detection, Low-resolution camera, Gamma correction, Heatmap, Calibration, Motion Estimation.

**1. Introduction**

This project is inspired from the powerful insights provided by widespread subjective studies, such as the Video Quality studies we have been a part of. Our aim is to simplify the process of gaze estimation to provide gross and robust results in not a very controlled environment such as using low resolution webcams on viewer’s computers or systems provided by concerned organizations. This method can also allow us to conduct crowd sourced studies, with subjects from varied demographics.

**2. VIDEO ACQUISITION**

Before proceeding with performing gaze estimation using the video captured of the viewer, we need to test different setups for the accurate positioning of the webcam, computer screen and the viewer. Our first setup (Fig 1.) was to use the laptop to display videos and the integrated webcam to capture videos of the viewer. This setup though simple had a major challenge in capturing the eyes and iris effectively while the viewer is looking at the lower half of the screen. This is caused due to the eyelids covering majority of the iris, thus reducing the accuracy of our circle fitting algorithm. After reading though papers, we realized that a small modification can overcome this shortcoming. Therefore, another setup (Fig 2.) with the webcam at the bottom of the screen showing videos was utilized. This setup ensured that the eyelids never really cover the iris completely (except while blinking) to hamper the iris and eye center detection. The viewer on average is at a distance of 1 meter from the screen playing videos. The webcam used captures videos at a fixed resolution of 1280x720 at 30fps.



Fig.1 Initial setup used which had the issue of eyelids covering the iris while viewer is looking at bottom of screen



Fig. 2 New setup used for conducting the experiments with the webcam aligned with bottom of secondary screen

**3. PRE-PROCESSING**

Some pre-processing methods are applied to the video frames while calibration as well as while watching test images or videos. In this section we will detail on the pre-processing used in this project as well as the ones mentioned in different gaze estimation techniques.

**4.1. Face and Eye region Detection**

Each video frame is read into Matlab and processed one by one. The frame f is converted from RGB to grayscale before moving forward.

Computer vision toolbox object detector utilizing Haar Cascades is applied on the grayscale frame to detect the face of the viewer. The frame is then cropped to only the face region detected, shown in fig 3. On this cropped area the computer vision toolbox eye detector object is applied to detect the eye region within the face. Both the eyes are detected simultaneously as shown in fig. 4. The individual left and right eye detection was also tested, but it is not very robust as it leads to false detections (namely reversing of the left and right eyes). Thus, we detect both the eyes together.

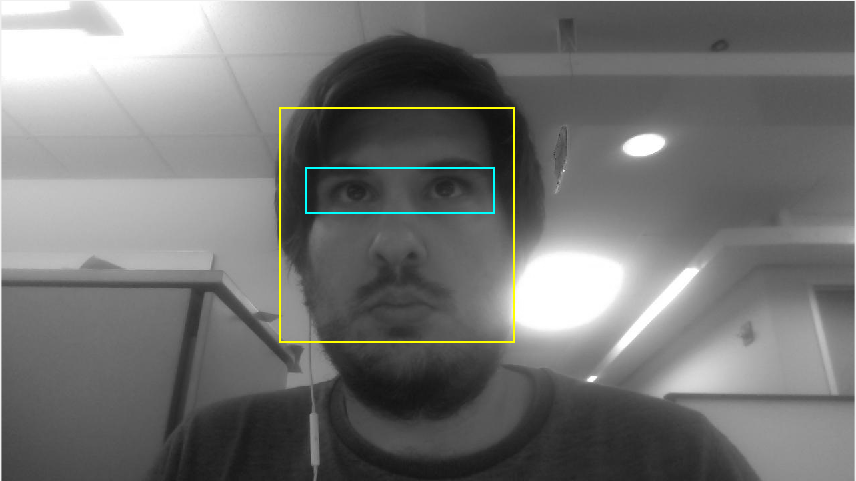


Fig 3. Grayscale of video frame captured with face (yellow) and eye (cyan) regions marked



Fig 4. Eye region retrieved form the video frame

**4.2. Light Illumination Correction**

The eye region obtained from the previous step suffers from shadowing due to light illumination used, which degrades the performance of the circle fitting algorithm as depicted in fig 5. This false detection of circles hampers accurate gaze estimation in our project as the eye centers play a crucial role in determining the point of regard. To overcome these issues we have utilized histogram equalization and gamma correction of the eye region obtained.



Fig. 5 Incorrect detection of circles by circle hough transformation due to light conditions

Histogram equalization is to contrast stretch the eye region image obtained from earlier steps. This helps in defining the iris within the eyes clearly. The result obtained is shown in fig 6.

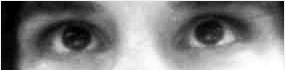


Fig. 6 Histogram equalized eye region

Gamma correction is then applied on the histogram equalized image with gamma=3. The gamma factor controls the non-linearity of the mapping function. As shown in fig. 7 for gamma >1 the mapping is weighted towards darker output values. Thus, the intensity of dark regions remains the same, while changing the intensity of light regions. This method results in removing of the shadow area around the eyes (fig. 8) preventing any false detection by the circle fitting algorithm.

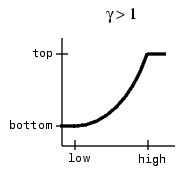


Fig. 7 Gamma correction mapping for gamma factor>1 (low and high are input image intensities, bottom and top are output image intensities)



Fig. 8 Eye region obtained after applying gamma correction

**4.3. Iris and eye center detection**

With the processed eye region image, the iris becomes well defined and is robustly detected by the circle fitting algorithm. We are using CHT (Circle Hough Transform) for identifying the iris and determining its center. The range for permissible radii is defined as a fraction of the eye region obtained, and remains the same for different viewers. The eye centers of both the eyes is detected and stored. This process is repeated for 20 frames for each calibration point. For videos captured while the viewer is looking at test images and videos, this process is conducted for each frame. An example result is shown in fig. 9 with the centers marked in red.



Fig 9 Example output of eye centers being correctly detected

**4. CALIBRATION**

Our next step is to generate a calibration map using which we can calibrate the behavior of the viewer’s eye movement. Different calibration maps were designed (Fig 10.) and tested for their accuracy of replicating the viewer’s eye movement characteristics. Best results are obtained for the calibration map in Fig 11. Calibration maps allow us to generate an estimated map of the eye centers while the viewer is looking at predefined fixed points on the screen. The eye center map thus obtained is later utilized for estimating the point of regard on the screen while the viewer is watching print or video advertisements. The video of the viewer looking at the calibration points is acquired simultaneously while the viewer is looking at each calibration point. For each calibration point 20 frames are acquired. The eye center is then determined using k-means clustering on the set of center values obtained for each calibration point. A calibration map is generated for each eye separately as shown in Fig 12. The eye center map is then averaged across the horizontal and vertical dimensions to decouple the errors in each direction. The final calibration map obtained is depicted in Fig 13.

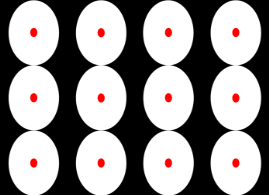
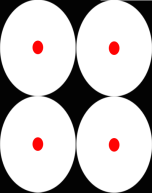
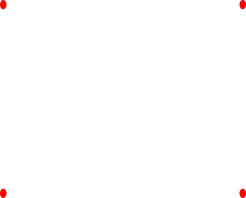
  

Fig. 10 Various calibration maps tested while designing the calibration process



Fig. 11 Final calibration used with 20 points uniformly spread across the screen (5 horizontal, 4 vertical)

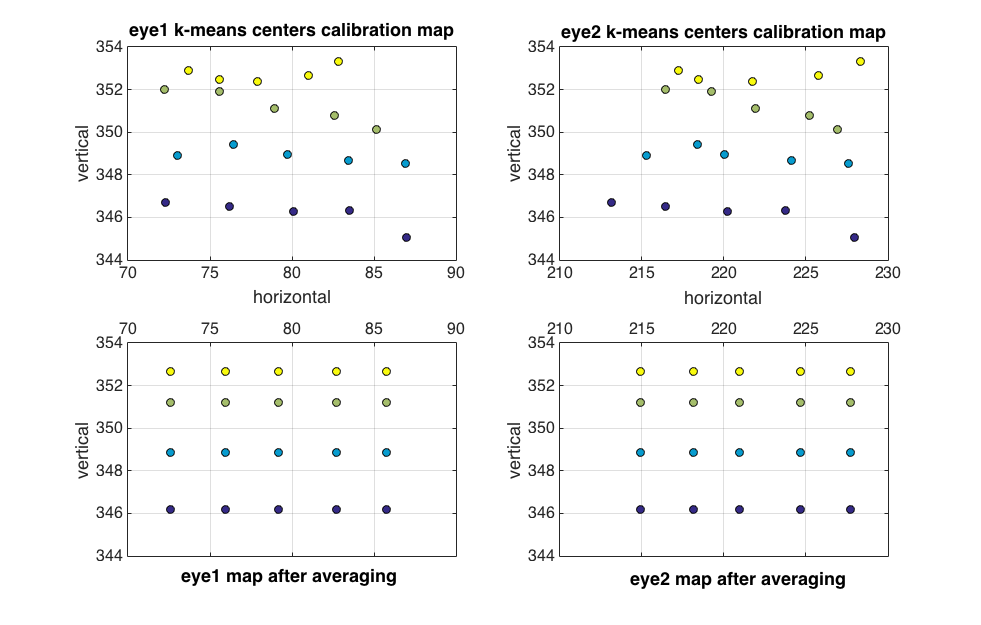


Fig. 12 Eye center calibration maps generated for each eye while the viewer is looking at each calibration point one by one

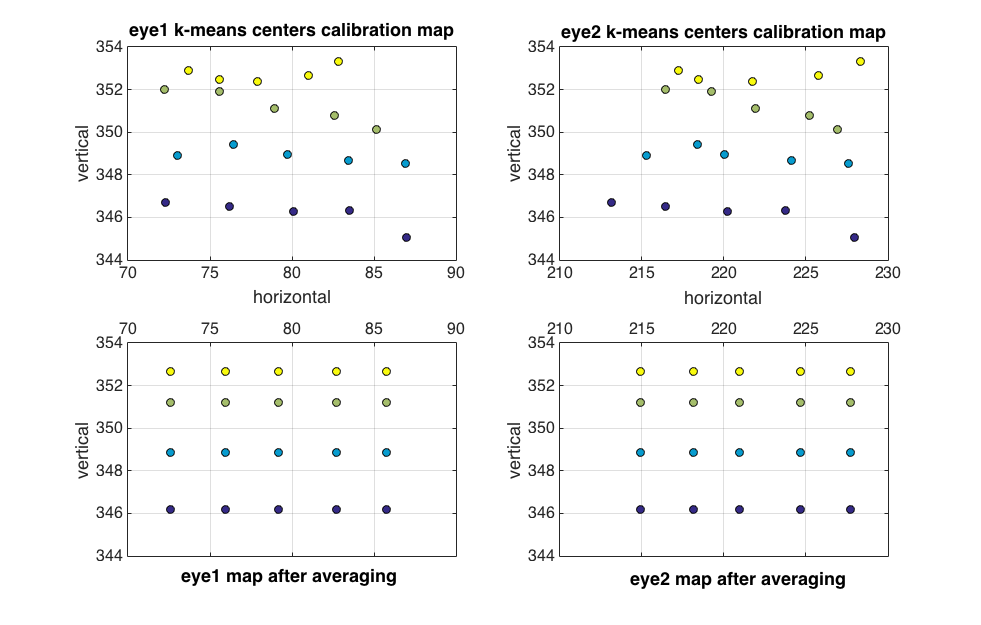


Fig.13 Eye center calibration maps obtained for each eye after averaging in horizontal and vertical directions

These calibration maps indicate that discrete points on the screen can be estimated using the eye centers obtained while viewer is watching each of these points. However, we need a continuous map to estimate any point of regard.

For designing the estimation technique the calibration obtained in Fig. 13 is first normalized. Thus, the resulting calibrations obtained in horizontal direction (5 points) lie between 0 and 1. A non-linear curve fitting is then performed to obtain an estimation function as shown in fig. 14 for both the eyes in horizontal direction.

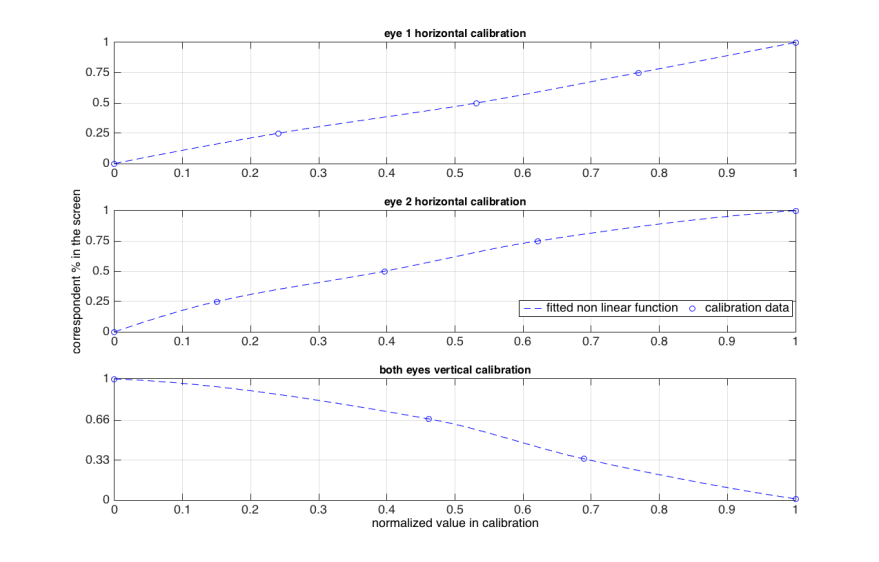


Fig. 14 Estimation functions generated for both eyes for horizontal coordinate estimation

The vertical calibration is obtained in terms of distance of eye centers with reference to the top of the frame. This is then averaged between the two eyes. These calibrations are then normalized to lie between 0 and 1. Again a non-linear curve fitting is performed to generate the vertical position estimator. This is also illustrated in fig. 15.

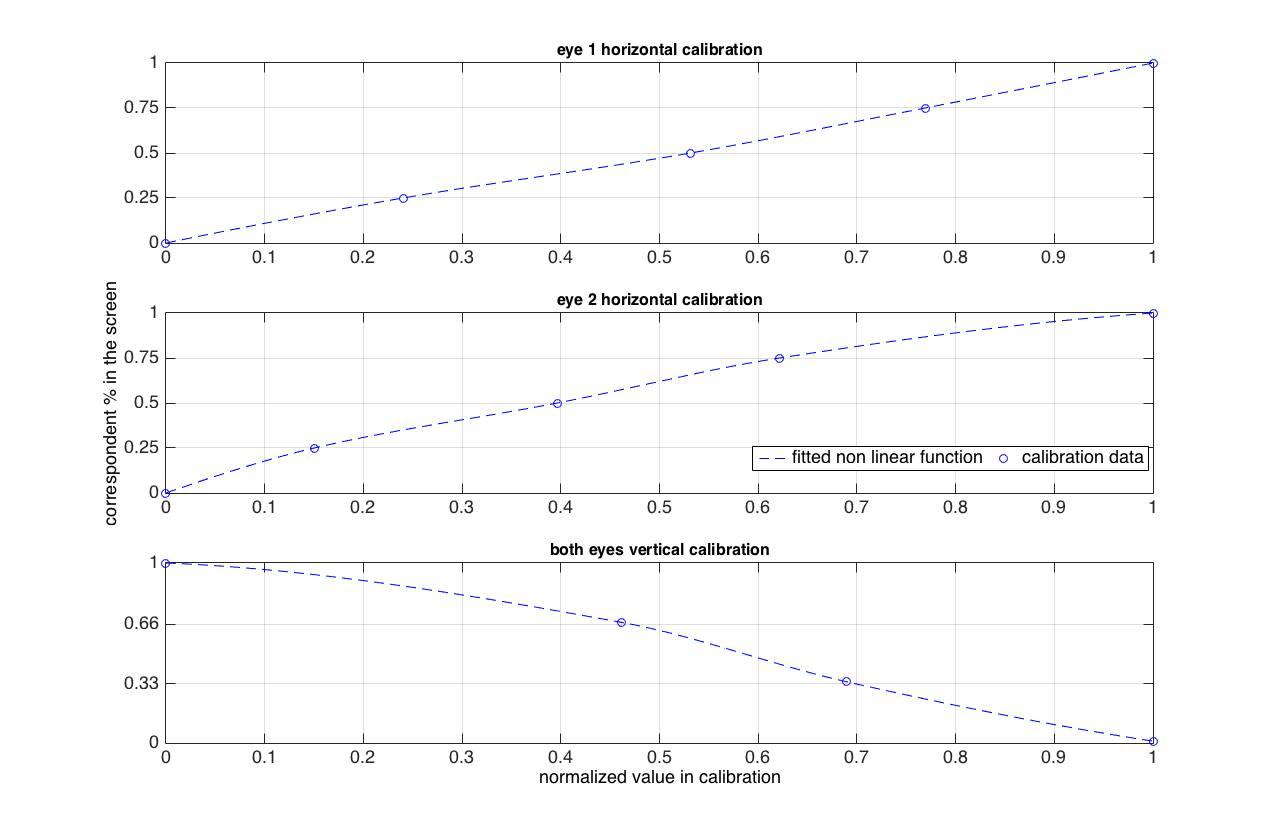
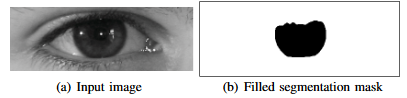


Fig. 15 Estimation function generated for both eyes for vertical coordinate estimation

**5. REFERENCE POINT IDENTIFICATION FOR ACCURATE TRACKING OF EYE MOVEMENT**

Different eye tracker and gaze estimation techniques either use Infrared light reflections or some feature as reference point for accurately tracking eye movement. One of the most commonly utilized feature as reference points are eye corners. Eye corners are chosen for this task as they do not change with facial expressions and are at a fixed position which can be modeled geometrically. A number of techniques are used for efficient eye corner detection. [1] Uses Harris & Stephens method of corner detection to obtain initial estimates for corners which as then filtered based on eye geometry to detect eye corners. Another method detailed in [2] converts the eye image to HSV format and uses the saturation component for eye corner detection. Saturation component is used as the sclera region is mostly dark which is multiplied by an iris mask to obtain an image as shown in fig.16. The corners can then be detected as the leftmost and rightmost points.



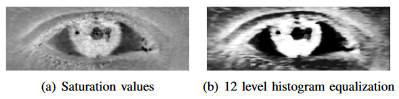




Fig. 16 one of the methods described to determine eye corner positions using saturation image and iris mask

We implemented both these methods for our application but did not get desirable results. The cause was identified to be the low resolution of our video frames which led to fairly severe blocking in the eye image leading to degradation in the performance of the mentioned algorithms. The low resolution also causes blocking when the frame is converted to HSV format. Both these issues are depicted in fig. 17.



(a)



(b)

Fig. 17 Examples of failures of popular eye corner detection methods

1. Saturation component of captured video frame is extremely blocky due to low resolution
2. Incorrect corners detected by Harris & Stephens corner detection method

Therefore, we use a different method for utilizing a reference point. From the calibration maps we realized that the movement of eyes in horizontal direction was effectively captured, whereas the vertical movement of eyes was not captured desirably (very less difference was obtained for vertical component of eye center position for vertically displaced calibration points). This is expected since the iris moves more freely in the horizontal direction as compared to the vertical. The issue is rectified by using the top of the frame as a reference. Hence for each eye center position, the vertical component is measured as the distance of the detected eye centers vertical position with respect to the top of the frame. Using this technique the variation in vertical direction was amplified resulting in desirable calibration maps as shown earlier.

**6. Dataset of ADVERTISEMENTS**

A dataset of interesting advertisements, both print and video, was selected to test the effectiveness of our eye gaze estimation method. We have used 50 print advertisements and 5 video advertisements to conduct our testing. All the data was sourced from advertisement websites listed under references. Some of the print advertisements used are pictured in fig. 18. The advertisements are selected to be a good mix of text, oriented objects, with and without a lot of details, off center (spread out) objects, with and without humans and different types with almost similar data and design.







Fig. 18 A few of the print advertisements used in our experiments

**7. INTERFACE**

Matlab was used to create an interface which displays videos or images in fullscreen mode while simultaneously capturing videos of the viewer. The calibration points, random points, test images and videos were displayed at a resolution of 1280x1024 pixels on the secondary screen.

**8. ESTIMATION**

**8. TESTING**

After generating calibration maps and estimation functions for each eye we proceed with testing the techniques. We have three different testing scenarios as follows –

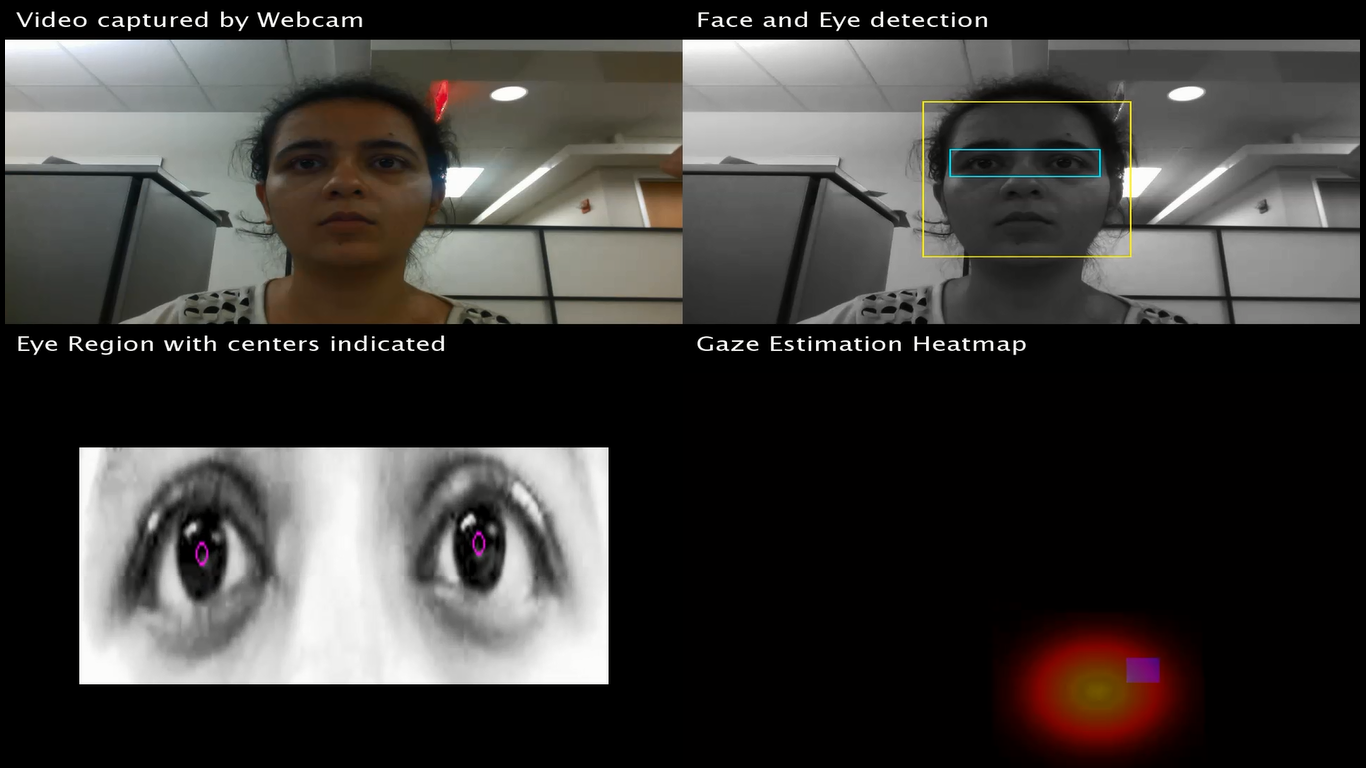
Random points displayed at predefined fixed locations on the screen. At each instance the square is shown at some random location on the screen. The prior knowledge of the location of these points allows us to compare the estimate with the actual values and thus determine the accuracy and error of our method.

Print advertisements are shown at random, each for a constant fixed amount of time. The viewer’s video is captured simultaneously for each advertisement displayed.

Video advertisements are played out on the screen in no particular order while recording the viewer for each of the advertisements shown.

**9. Accuracy, error and complexity**

The experiment of displaying random points at predefined locations to the viewer is to gauge the accuracy of our method. Fig. 19 shows screenshots of a few random points generated along with their corresponding estimates. The error is calculated for both horizontal (x) and vertical (y) directions as follows –



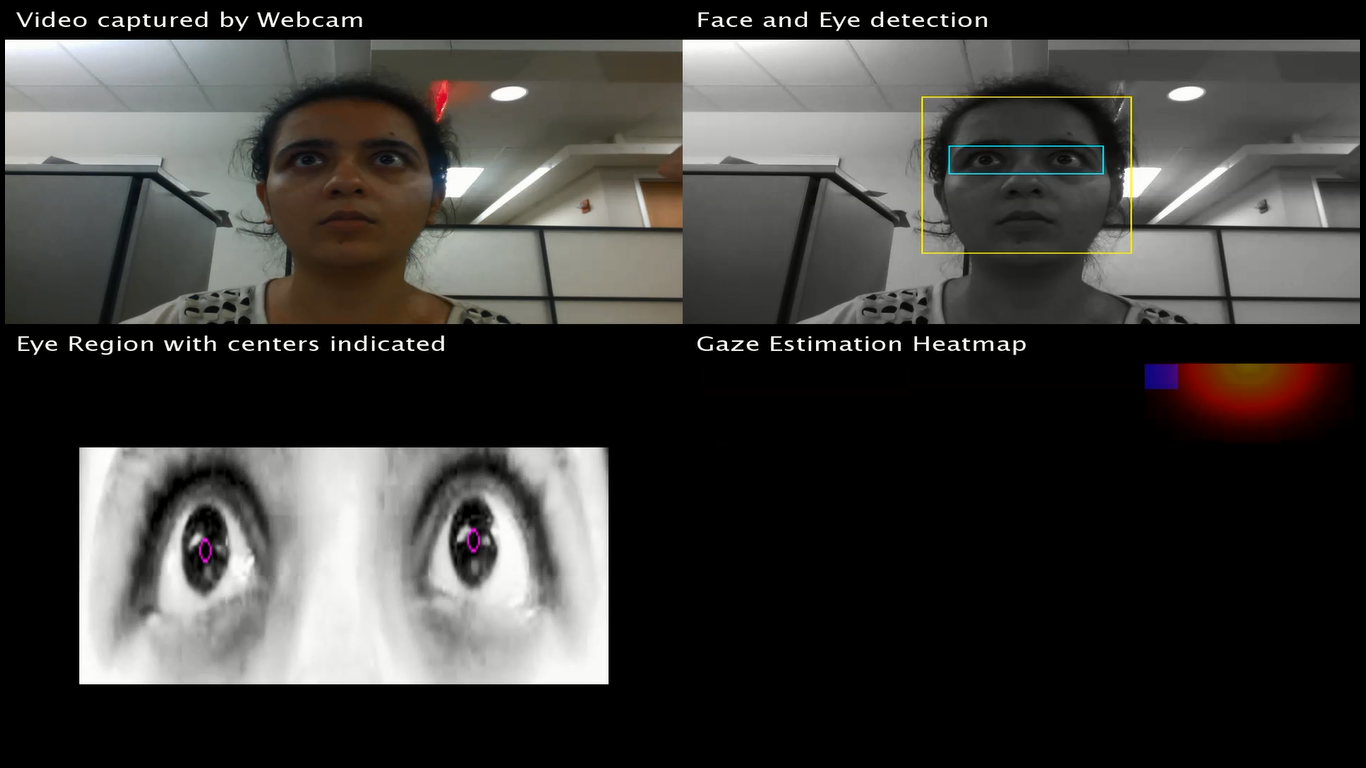


Fig. 19 Screenshots showing random points experiment. Blue squares are the predefined random points displayed. Red area is the Gaussian kernel heatmap corresponding to estimated point of regard

The errors computed are detailed in Fig. 20. It is observed that error in horizontal direction is approximately half of that in vertical implying that we are able to capture better resolution in horizontal direction.

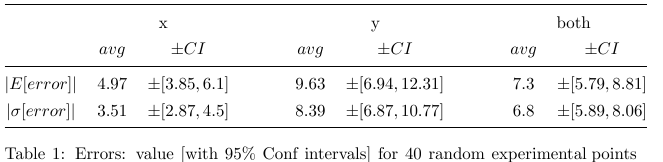
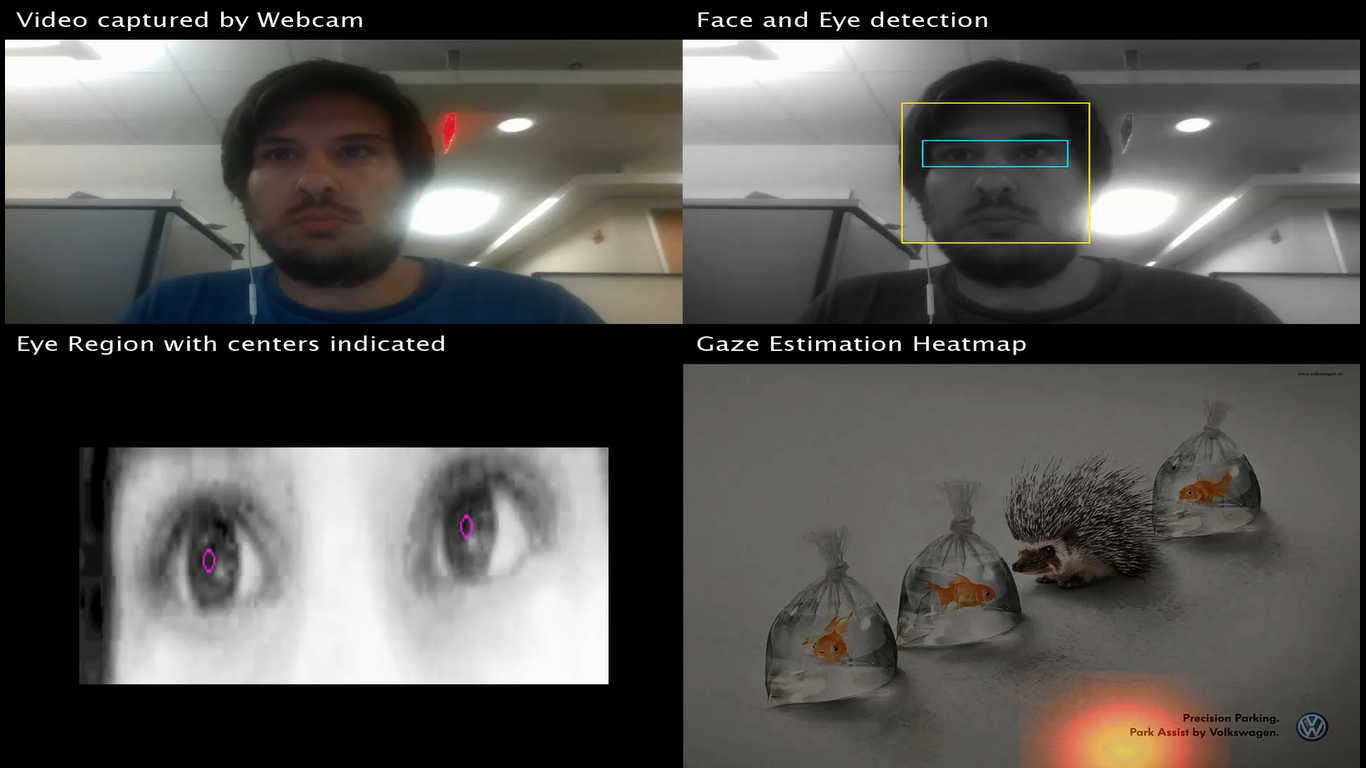
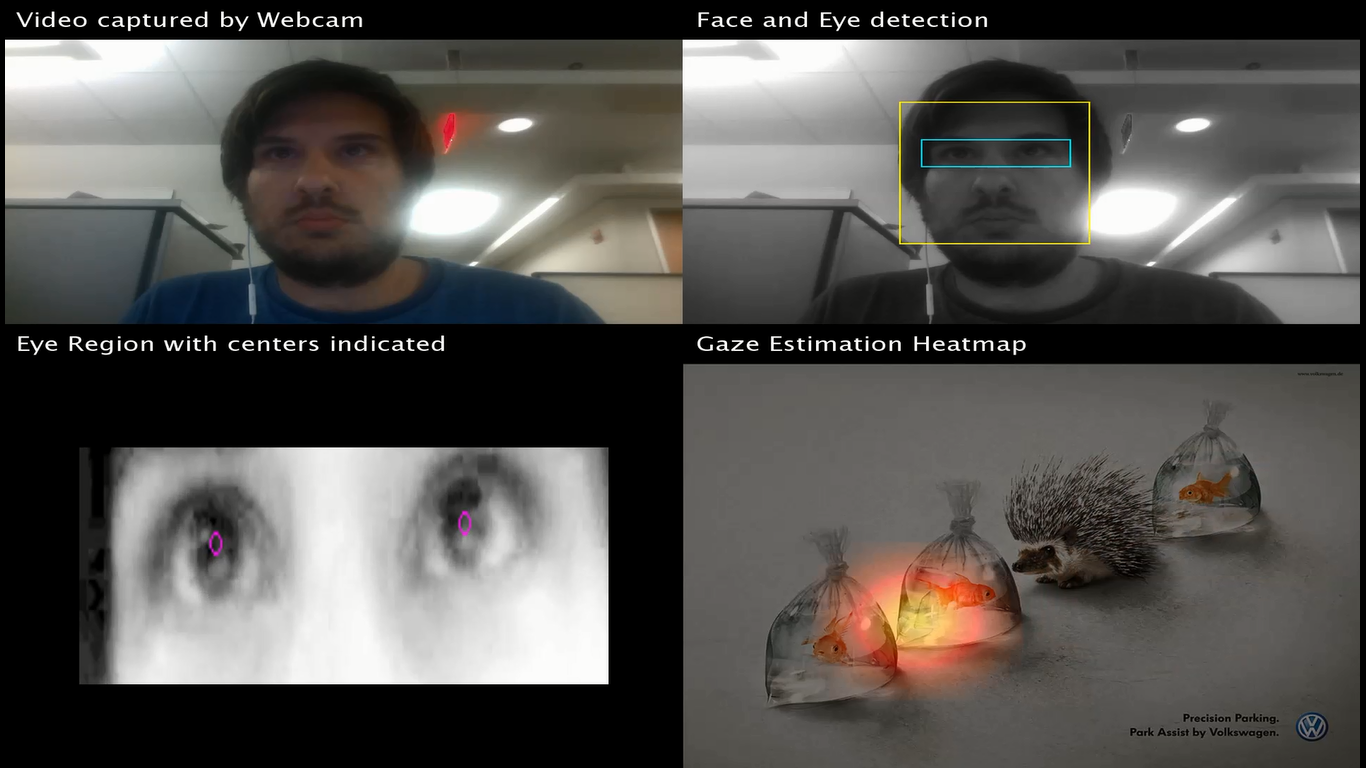


Fig. 20 Mean and standard deviation of error computed for horizontal and vertical directions

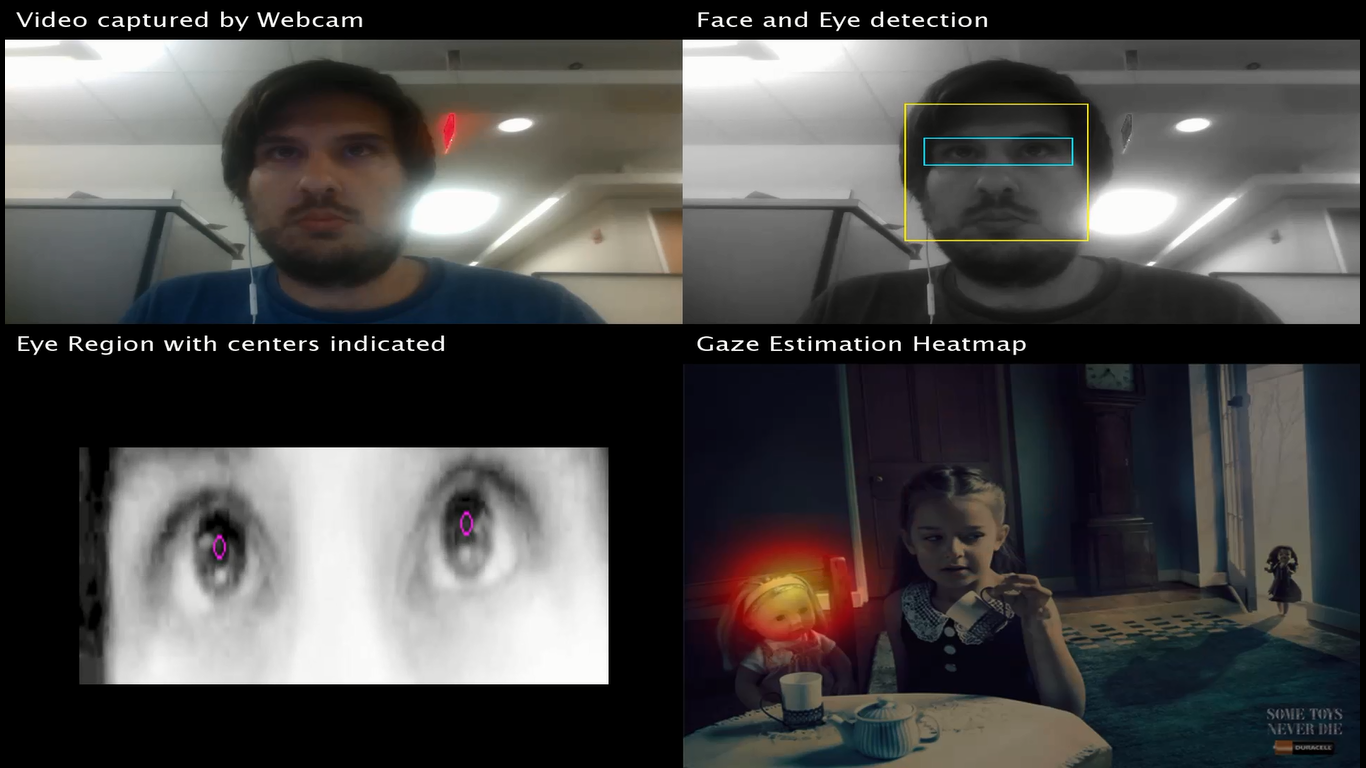
**10. HEATMAP GENERATION**

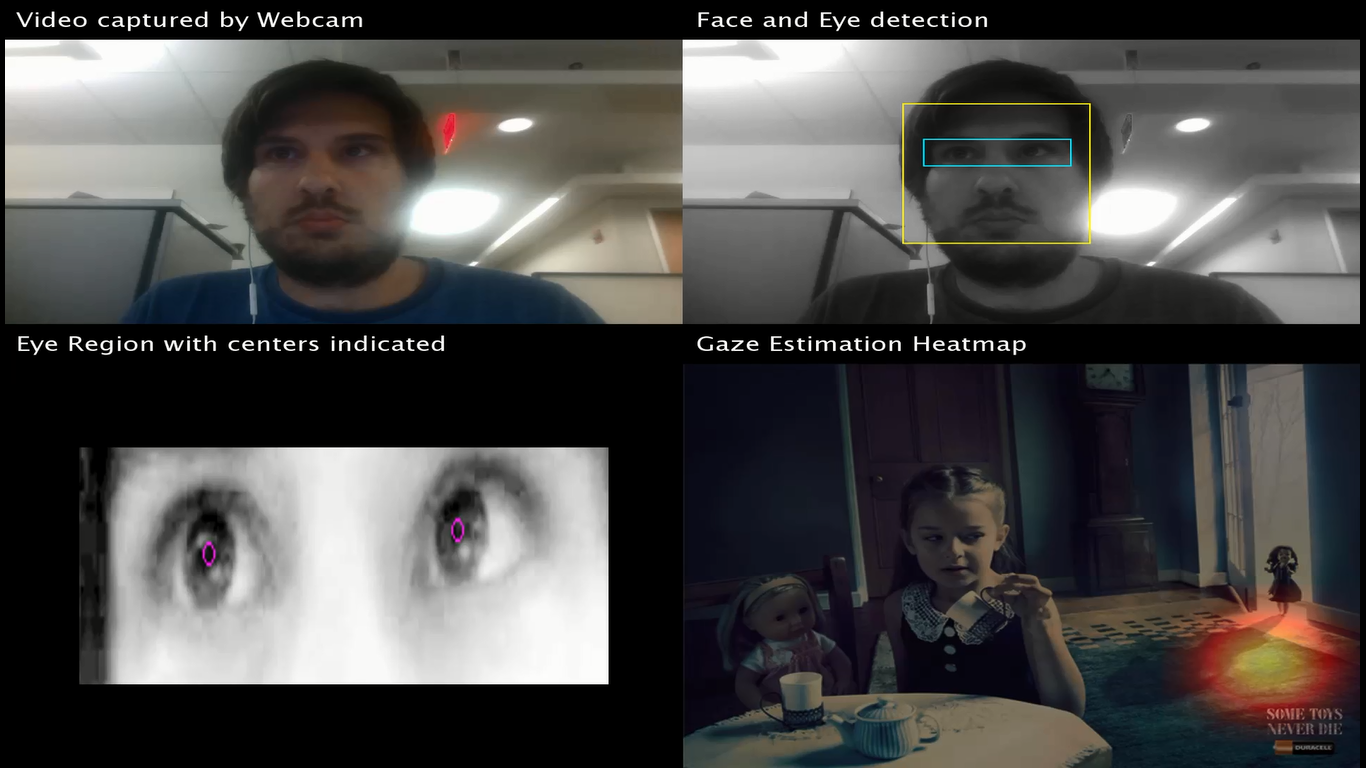
Standard technique of using a heatmap is utilized to indicate the location of the estimated point of regard. The heatmap is generated as a Gaussian kernel with its size in horizontal and vertical coordinates being governed by the standard deviation of the error obtained in respective directions. Care is taken to display only a portion of the heatmap when the estimated point is at or close to the edges of the screen. The generated heatmap is then superimposed on the original data (random point, print advertisement or video advertisement frames) to generate our final results. A few examples of heatmaps generated are shown in fig. 21.





(a)





(b)





(c)

Fig. 21 Examples of gaze estimation heatmaps

1. Example screenshots for Volkswagen print ad
2. Example screenshots for Duracell print ad
3. Example screenshots for Coca Cola video ad

**11. COMPARISON WITH OTHER TECHNIQUES**

While working on this project we came across a number of techniques and methods to implement eye tracking and gaze estimation. In this section we compare some aspects of our project with each of these different methodologies.

Firstly, the method elucidated in [1] records subjects viewing an animation at 210 fps as well as used special lighting arrangements, while resulting in an estimation error of 5.6%. This method uses Harris & Stephens corner detection with constraints as described in [7] to detect eye corners. As we observed this method for eye corner detection suffers heavily with low resolution images and no control on lighting conditions.

Further, the method described in uses binarization of face images obtained using webcams to detect eye regions which is heavily dependent on the threshold specified which is further governed by the lighting conditions. In addition gaze estimation is performed using SVM using a dataset for different iris positions which we believe can become highly complex and exhaustive for estimation of all the possible iris positions and subsequently points of regard.

The algorithms studied in [6], namely DAISMI and DTBGE are compared in terms of human computer interaction application. While DAISMI algorithm has various control factors detailed in which hampers the generalization of this method in different settings, DTBGE is uses binary images which is dependent on threshold specified.

The methods illustrated in [5] are based on high resolution images obtained of the viewer’s eye using a camera that zooms in on one eye based on the head position estimation. Further, canny edge detector is used to detect the edges of iris in eye region which is controlled by high and low threshold values, a method we observed to not work well on low resolution images acquired by integrated webcams.

The technique detailed in [4] for gaze estimation using low resolution images uses high resolution training data, SVD, down-sampled training images and known gaze directions.

The method described in [3] utilizes more than one webcams to record the viewer, elliptical model of iris detected using Hough transform which has higher complexity than using a circular approximation as well as the threshold is set manually for binarization of the eye image obtained, thus limiting its widespread application.

As a comparison, our method utilizes the integrated webcam capturing video of the viewer in low resolution. We use circular approximation for iris detection which has lower complexity than using an elliptical model by hough transformation. We do not use canny edge detector or binary images which are governed by the threshold specified. Calibration is performed once for each viewer while viewing 20 point on the screen which is then used to generate an estimation function. This method has lower complexity than creating a training data set of images and using SVM. In addition the estimation technique used does not depend on the distance of the viewer from the screen thus allowing the viewer to not be at a fixed distance strictly. We observed that placing the webcam at the bottom of the screen allows for robust iris and eye center detection while looking at lower sections of the screen as it prevents extensive occlusion of the iris by eyelids.

12. OBSERVATIONS

For the test data of print and video advertisements we studied following preliminary inferences can be drawn –

* The general trend of the gaze of the viewer is to first gain knowledge of the content (such as faces, objects, as well as interactions between the objects). The viewer is then drawn towards a tagline highlighting the core idea of the advertisement. This is then followed again by a general cursory gaze over the advertisement to connect everything together.
* For video advertisements majority of the viewer’s gaze and attention is drawn towards motion (as mostly interesting things are associated with motion). If there is no central point of attention in the scene the gaze is usually in the middle of the screen, suggesting that the user is looking at the whole scene and not a particular section.
* For Coca-Cola video advertisement it is observed that the brand products are strategically placed in specific positions and frames within the video to use the fact that viewer is attracted towards motion.
* These inferences drawn are fairly shared for the gaze estimates obtained by both of us.

13. CONCLUSION

The techniques used in this project for eye movement tracking and gaze estimation are fairly straight forward with negligible amount of manual control required (grossly keeping our head still, webcam at the bottom of the screen, gamma correction factor).

The results obtained are very promising and can be used for conducting subjective studies to gauge the effectiveness of advertisements.

For widespread application some of the following challenges will need to be addressed –

* The eye tracking does not perform well if the user is wearing spectacles
* Faster processing if needs to be implemented in real time
* Head motion estimation and compensation
* Detection of eye corners for using them as reference points

6. WORK BREAKDOWN

Following is the work breakdown detailing the parts of the project completed by us –

|  |  |
| --- | --- |
| Project Module | Name of team member |
| Advertisement dataset creation | Ambika |
| Interface creation to display calibration points, random points, test advertisements and simultaneous video capture | Ambika |
| Calibration map and estimation function | Pablo |
| Error and complexity computation | Pablo |
| Testing | Pablo and Ambika |
| Heatmap and Demo generation | Ambika |
| Presentation and Report | Pablo and Ambika |
| Data pre-processing | Pablo |

**7. References**

[1] A. Dasgupta, A. Routray, S.N. Mallavollu, “Detection and Estimation of Iris Centre”, arXiv, Cornell University

[2] G. Santos, H. Proenca, “A Robust Eye Corner Detection Method for Real-World Data”, International Journal of Cognitive Biometrics, 2011

[3] T. Fukuda, K. Morimoto, H. Yamana, “Model-Based Eye-Tracking Method for Low-Resolution Eye-Images”, 2nd Workshop on Eye Gaze in Intelligent Human Machine Interaction, 2011

[4] Y. Ono, T. Okabe, Y. Sato, “Gaze Estimation from Low Resolution Images”, *Proc. 2006 IEEE Pacific-Rim Symposium on Image and Video Technology (PSIVT'06)*, pp.178-188, December 2006.

[5] J.G. Wang, E. Sung, R. Venkateswarlu, “Eye Gaze Estimation from a Single Image of One Eye”, Proc. 2009 9th IEEE International Conference on Computer Vision (ICCV 2003) 2-Volume set

[6] I.F. Inke, J.W Kim, “A 2D eye gaze estimation system with low-resolution webcam images”, EURASIP Journal on Advances in Signal Processing, 2011

[7] C. Xu, Y. Zheng, and Z. Wang, “Semantic feature extraction for accurate eye corner detection,” 2008 19th Int. Conf. Pattern Recognition., pp. 1–4, Dec. 2008.