Wearable Eye Tracking System

Xiaoyong Ye
xye@g.clemson.edu

Franz Alexander Van Horenbeke
fvanhor@g.clemson.edu

David Abbott
dabbott@clemson.edu

ABSTRACT
Tracking the user's gaze point has great importance for the research in humnan-computer interfaces. Nowadays People are trying to build cheaper, reliable and wearable eye trackers. In this paper we built and developed accurate and efficent algorithms for a wearable real-time eye tracker. Our implementation is based on pupil center localization and homographic mapping. We also proposed possible improvements including corneal reflection detection and different mapping techniques.

Keywords
Wearable eye tracker, ellipse fitting

INTRODUCTION
Wearable eye trackers have the capability to know where the user looks while they are doing various actions. The current wearable eye tracking equipment used is heavy, expensive, uncomfortable, and limited in their functionality. To cut down on these difficulties, we will develop a more customized eye tracker using less expensive and lighter materials, which will help improve its capabilities. Some of this work has already been established which will be used as a basis for our project.
To understand our wearable eye tracker, its important to note where eye tracking originated from. The development of eye trackers has drastically evolved from where it began because of continuous efforts in its design and implementation. Dodge and Cline [6] helped develop the first precise, non-invasive eye tracking technique, using light reflected from the cornea [7]. This helped give instruction for future developments to occur based on their work concerning the location of eye fixations.
In this paper we will describe the algorithms for the eye tracking system we are implementing. Details about the algorithms will be compared with previous work also.

BACKGROUND
Some previous work that has been written about wearable eye trackers involves techniques denoted by Babcock and Pelz's eye tracker[11], which is based on infrared LEDs. The problem encountered with this technique is that any environment where infrared light cannot be controlled may produce inaccurate results. Some work has also been done in the visible spectrum by Li and Parkhurst[12] and Ryan[2]. The main advantage of Ryan's work compared to Li's and Parkhurst's is that it is designed to work under variable light conditions, which is important in a wearable eye tracker. Ryan[3] developed an eye tracking system based on the Daugman's[5] elastic sheet model of the iris and on the Starburst algorithm.
Some of our work is based off of existing algorithms including the Starburst algorithm. One of those techniques is called RANSAC (Random Simple Consensus) paradigm, which helps draw the ellipse from detected feature points[13]. The mapping we used was called homographic mapping, involving a 3x3 matrix that has eight degrees of freedom, between the scene point and eye coordinate.

DEVICE
Our eye tracker is composed of two cameras, one tracking the eye (eye camera) and the other recording the scene (scene camera). These cameras are held together by a head gear structure (Figure 1), which has an adjustable knob placed on top heightens or lowers the head gear based on the users head. It also has a back knob that can tighten or loosen around the users head. At the top of the head gear is bounded the scene camera, which only moves with the direction of the users head. The scene and eye camera will use a 6mm and 25mm focal diameter lens, respectively. Both have a maximum resolution of 640x480 pixels and a pixel size of 7.4 x 7.4μm. The eye camera is attached by mounted aluminum material, located in the front of the mount. Both cameras were made by Point Grey Research.

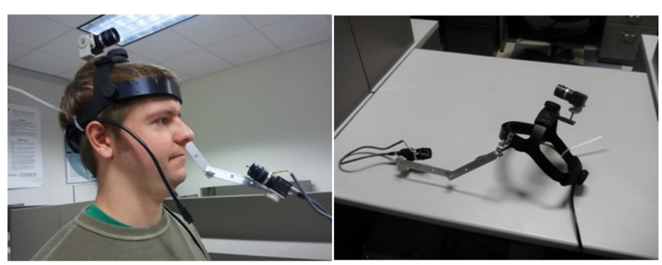

Figure 1. Our eye tracker

METHODS

The flow chart of this program is shown on Figure 2, the following paragraphs will give more detailed explaination of all the algorithms.

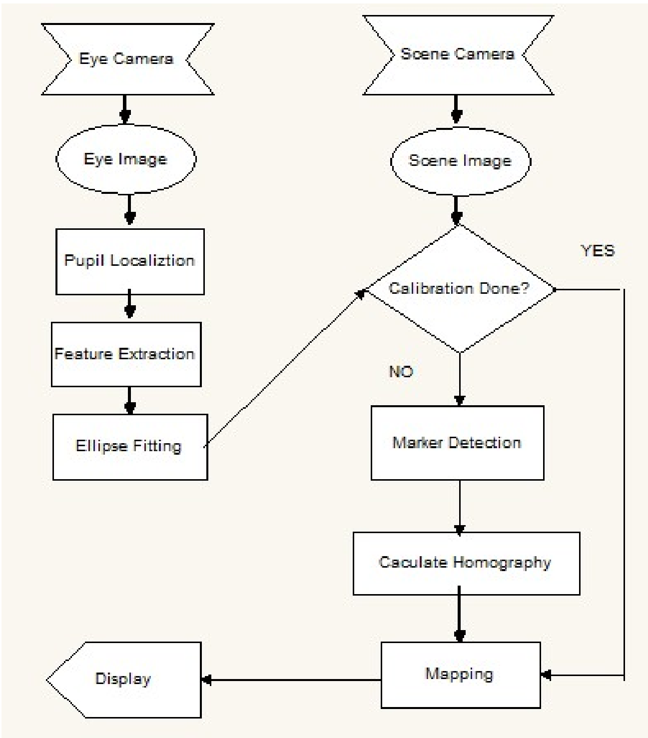

Figure 2. Flow Chart

Pupil Localization

It is obvious that the pupil area is much darker compared to the rest of the area inside the image. So the common method would be apply a threshold to the image. But the most difficult problem is how to get the threshold of binarization for every image. The Threshold I used for this project is derived from the Mean value of the graylevel of the whole image:

$$Threshold = \frac{0.4}{Size_{image}} \Sigma Intensity \qquad (1)$$

The histogram of an eye image would contain three peaks, one peak for pupil, one for iris ring and the last one for other. I come up with an automatic thresholding algorithm that use a modified Ostu's method twice on the image. The first time it seperate the iris from the image and the second time it seprates the pupil from the iris. The results I get is decent almost in every frame of image. But Ostu's Method is time consuming and not suitable for this real time application.

After thresholding, doing image morphology is important because the eyelash has the similar intensity value of the eyeball and when it is connected with pupil it will give us trouble on finding the location of the pupil. So we here do an erosion on the binary image we get. Since the eyelash and eyelid are thin and long, erosion will get rid of them.

Connected components analysis is applied on the binary image later. If a component pass all the criterias, it will be accepted as candidate for pupil. Depending on the geometric and shape features we will be able to automatically locate the pupil center by calculating the centroid of the best fitting component.

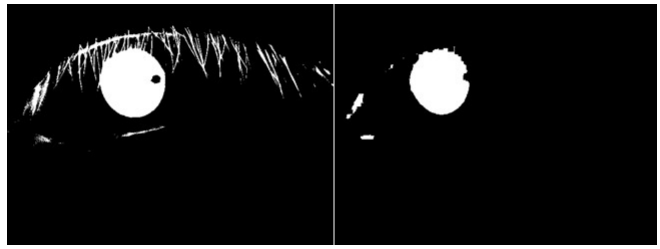

Figure 3. Thresholding and Erosion

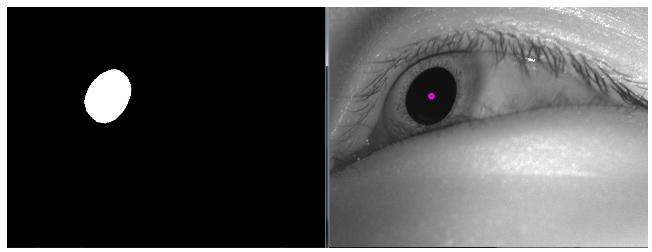

Figure 4. Object recognition and centroid finding

Feature Extraction

After finding the pupil center, I implemented Starburst algorithm on the edge image. Then feature points for pupil bondary will be extracted.

1. Apply Canny Edge Detection to the image.
2. Shoot rays extending from the pupil center in a range of directions.
3. If the ray hits any edge within a bounding box, place a feature point.

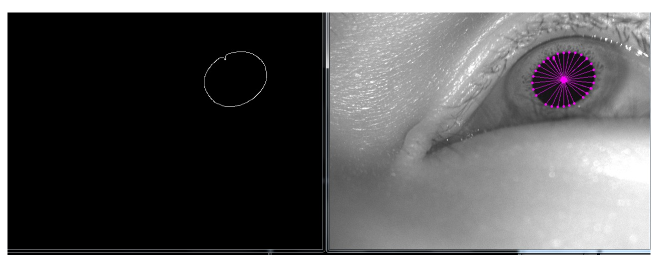

Figure 5. Feature Points Extraction

Ellipse Fitting

Given a set candidate feature points, the next step of the algorithm is to find the best fitting ellipse. We apply the Random Sample Consensus(RANSAC) paradigm for model fitting. By selecting five pupil points at random we can create a 5x5 quations for an ellipse:

$$ax^2 + by^2 + cx + dy + exy + f = 0 \qquad (2)$$

Singular Value Decomposition (SVD) on the conic constraint matrix generated with normalized feature points is used to find the parameters of the ellipse that fits the five points.

1. Randomly select five points.
2. Normalize the feature points.
3. Build conic constrait matrix using the normalized feature points.
4. SVD is applied to solve the parameters for the ellipse.
5. Apply the ellipse to all the feature points and caculate the number of inliers
6. Repeat step 1 to 5 for 100 times.
7. Select the ellipse that has the most inliers

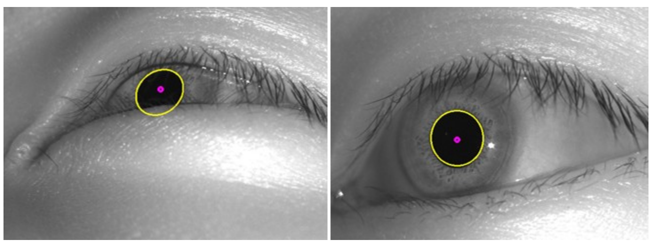

Figure 6. Good fitting Results

Figure 7. Not ideal fitting Results

Calibration

In order to calculate the point of gaze of the user in the scene image, a mapping between locations in the scene image and an eye position must be determined. The mapping will be calculated by calibration. During calibration, the user is to look at scene markers which the coordinates are deteted $\vec{s}$ =(x_s, y_s, 1). At the mean time the pupil center of $\vec{e}$=(x_e, y_e, 1) is measured. Then we generate the mapping between eye and scene using linear homographic mapping. Ihe mapping H is a 33 matrix and has eight degrees of freedom.

The markers can be placed anywhere where there is a high contrast between the markers and background. The center of each markers and eye coordinates are stored through several steps:

1. Threshold the graylevel image using Ostu's method.

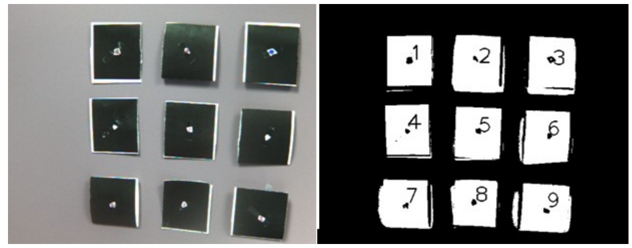

Figure 8. Scene Markers for Calibration

2. Connected components Analysis is applied to the foreground objects
3. Each component is measure by its shape and geometry, if pass threshold then marker accepted
4. Sort the markers by its center coordinates
5. User then look at each marker through 1 to 9 and press the number on keyboard. Then the coordinates of both marker and eye will be stored

Homographic mapping

After we recorded a group of pairs from the scene and eye image, we will gnerate the mapping between the two sets of points using homographic mappping. RANSAC is implement to select the best mapping matrix. The steps of the algorithm is below:

1. Randomly choose 9 pairs of points and normalize them
2. Direct Linear Transform and SVD are implemented and we will get a mapping matris H.
3. Use the H we get and map the eye coordinates to scene and calculate the total error.
4. Repeat 1 to 3 and select the H with the smallest error.
5. Once the mapping is determined, the user's gaze point would be as $\vec{s} = H\ \vec{e}$.

EXPERIMENTAL RESULTS

After calculating the H matrix, we then can map the eye coordinates to the scene point. We display the gaze point as a cross in the scene image as shown in Figure 9 and Figure 10.

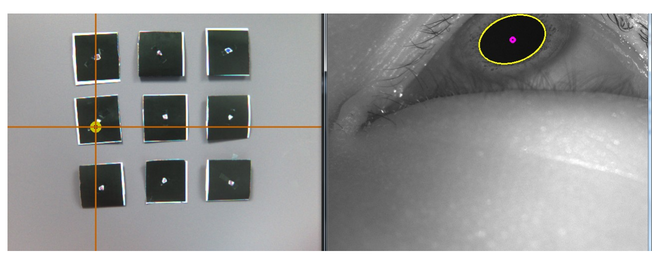

Figure 9. Point of Gaze at the block 4

The camera is taking images 25 frames per second, the quality of the processed video is smooth.

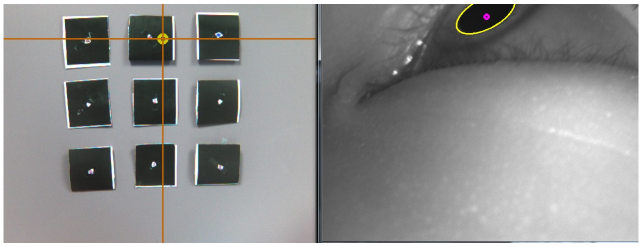

Figure 10. Point of Gaze at the block 2

We evalute the accuracy of our implementation by mapping the eye coordinates to the scene points using the coefficents we calculated. For the above experiment, I got an H matrix:

$$H = \begin{bmatrix} -3.40346 & -0.266491 & 1459.11 \\ 0.79225 & 5.06758 & -344.293 \\ -0.000645615 & 0.00193906 & 1 \end{bmatrix}$$

The corresponding coordinates are:

$$eye(x, y) = \begin{bmatrix} 365 & 44 \\ 328 & 50 \\ 287 & 60 \\ 356 & 63 \\ 318 & 68 \\ 280 & 82 \\ 360 & 88 \\ 324 & 94 \\ 287 & 106 \end{bmatrix}$$

$$scene(x, y) = \begin{bmatrix} 258 & 184 \\ 379 & 192 \\ 501 & 190 \\ 244 & 299 \\ 369 & 296 \\ 488 & 302 \\ 221 & 406 \\ 343 & 401 \\ 462 & 403 \end{bmatrix}$$

$$s' = H * e = \begin{bmatrix} 241 & 197 \\ 372 & 190 \\ 500 & 200 \\ 258 & 288 \\ 387 & 272 \\ 495 & 299 \\ 224 & 412 \\ 340 & 399 \\ 445 & 411 \end{bmatrix}$$

The average mapping error given in pixels is 13 in our 640X480 image.

CONCLUSION

We built and developed accurate and efficent algorithms for a wearable real-time eye tracker. The pupil center is located through thresholding and image morphology. Then RANSAC is applied to maximize the accuracy of ellipse fitting in the presence of feature-detection errors. Our ellipse fitting is robust to variation in the distance and luminance condition. Finally homographic mapping is used to calculate the user's gaze point in the scene.

FUTURE WORK

A number of improvements could be made to our current implementation. For example, instead of using the pupil center's coordinates for mapping, we could use the vector difference between the pupil center and corneal refletion point. The vector difference will lead to better performance because the corneal reflection stays stable on the eyeball.

The ellipse fitting algorithm we use is RANSAC, there are other methods like Fitzgibbon's Direct Least Square fitting which proves work better even with few very noisy points. Next step is to implement this algorithm and test its efficency and accuracy.

Also different Mapping techniques can be used besides a homographic mapping. There are other techniques like linear mapping and second-order polynomial mappings. We will need to test the other methods and calculate their accuracy.

In the hardware, the head gear tends to vibrate when the user makes a fast move with their head. This affects the eye camera critically due to the system's high sensitivity to the position of the eye. Moreover the tracker is quite heavy and it is not comfortable to wear, lighter cameras are preferred. As the scene camera is placed over the user's head, the image captured is higher above what we really see. If the scence camera could be placed beside the user's eye, the mapping would be more accurate.

REFERENCES

1. Dongheng Li, Derrick J. Parkhurst. Starburst: A robust algorithm for video-based eye tracking. *Elsevier Science*. 2005.

2. Wayne J. Ryan, Andrew T. Duchowski and Stan T. Birchfield. Limbus/pupil switching for wearable eye tracking under variable lighting conditions. *In ETRA 08 Proceedings of the 2008 symposium on Eye tracking research & applications*. pages 61-64, New York, NY 2008. ACM.

3. Wayne J. Ryan, Damon L. Woodard, Andrew T. Duchowski, and Stan T. Birchfield. Adapting Starburst for Elliptical Iris Segmentation. *In Biometrics: Theory, Applications and Systems* 2008. BTAS.

4. P. Li and X. Liu.An incremental method for accurate iris segmentation. *Proc. Pattern Recognition, 2008. ICPR 2008. 19th International Conference*. pages 1-4, 2008.

5. J. Daugman. High confidence visual recognition of persons by a test of statistical independence. IEEE Transactions on Pattern Analysis and Machine Intelligence, 15(11):11481161, November 1993.

6. Dodge and Cline (1901). The angle velocity of eye movements. Psychological Review , 8, 145-157.

7. Jacob, R.J.K., and Karn, K.S., Eye tracking in humancomputer interaction and usability research: Ready to deliver the promises (Section Commentary), in The Mind's Eye: Cognitive and Applied Aspects of Eye Movement Research, J. Hyona, Radach, R., and Deubel, H., Editor 2003, Elsevier Science. Amsterdam. p. 573-605.

8. Eyetracking. Retrieved from http://www.cis.rit.edu/mvrl/eyetracking.html.

9. Crundall, D. E., Underwood, G., and Chapman, P. R. (1998). How Much Do Novice Drivers See? The Effects of Demand on Visual Search Strategies in Novice and Experienced Drivers. In G. Underwood (Ed.), Eye Guidance in Reading and Scene Perception (p. 395-418). Oxford, England: Elsevier.

10. Eye Tracking: The Importance of Landing Pages. 2010. Retrieved from http://eyetrackingupdate.com/2010/09/17/eye-tracking-importance-landing-pages/

11. Babcock, J. S. and Pelz, J. B. 2004. Building a Lightweight Eyetracking Headgear. In Eye Tracking Research&Applications (ETRA) Symposium. ACM, San Antonio, TX, 109114.

12. Li, D. and Parkhurst. D. 2006. Open-Source Software for Real-Time Visible-Spectrum Eye Tracking. In Conference on Communication by Gaze Interaction (COGAIN). COGAIN. Turin. Italy. 18-20.

13. M. Fischler, R. Bolles, Random sample consensus: a paradigm for model tting with applications to image analysis and automated cartography, Communications of the ACM 24 (6) (1981) 381-395.

14. R. K. G. Annapoorani, P. Gifty Jeya, and S. Petchiammal.Accurate and Fast Iris Segmentation. *International Journal of Engineering Science and Technology*. vol. 2, 2010.