IJCSE International Journal of Computer Sciences and Engineering Open Access
Review Paper Volume-3, Issue-4 E-ISSN: 2347-2693

Real Time Eye-Tracking Using Web Camera

Riddhi Chavda[1],Madhura Barve[2] , Amit Doshi[3]and Ruhina Karani[4th]

[1,2,3,4,]Dept. of Computer Engineering, D.J. Sanghvi College of Engineering, Mumbai University, India.

riddhic17@gmail.com, madhura.b2805@gmail.com, amitdoshi4293@gmail.com, ruhina.karani@gmail.com

www.ijcseonline.org

Received: April /02/2015 Revised: April/11/2015 Accepted: April/23/2015 Published: April/30/ 2015

Abstract — Traditional input devices such as keyboard, mouse and joystick have been around for some while now. With the advancements in the field of Human Computer Interaction, eye tracking or iris tracking is the most promising field. It will fundamentally change the way we interact with computers. The main aim of this project is to develop a low cost application running in an open source environment and a widely used operating system Linux, to replace the traditional computer mouse with the human iris for cursor movement. The target audience majorly consists of handicapped people or people with physical impairment. The system designed aims at detecting the user's eye movements for navigating the cursor, analyzing the nature and timing of blinks, which in turn is used as an input to the computer as a mouse click. The system consists of a good resolution Logitech C270 HD webcam, as opposed to the otherwise popular infrared cameras available in the market. The existing cameras used for tracking are highly expensive but our system is affordable and easy to use. In the project we have used the Fabian Timm image processing algorithm to achieve iris tracking. With future development, we believe our system has the potential to be used as a fully functional substitute for the mouse pointer.

Keywords—pattern recognition; Human Computer Interface; eye-tracking; Fabian Timm algorithm; blink detection

I. Introduction

Many devices which exploit the remained abilities of people, who are physically challenged to operate computers, have been invented in the recent years. Speaking of which, computer vision has made considerable research under the category of object-tracking. Thus, now the requirement of direct contact with the cornea has been eliminated to replace it with more non-invasive techniques. The high-cost factor of the available tracking devices has hampered its popularity in terms of usage. Eyes or iris tracking has added a new paradigm in the world of Human Computer Interaction. In recent years, economical web cameras are easily available, which facilitates incorporation of these systems on a larger scale, thus annulling the requirement of rather expensive equipment and high end video cameras. We implemented the Fabian Timm algorithm which locates and tracks the user's eye in consecutive frames of the video stream. The intended input is the region of interest, where the search procedure takes place, i.e. only the eye image. To navigate the mouse pointer on the screen, user will have to move our iris to the desired position and then blink for over threshold number of frames duration to establish a valid click.

II. REVIEW OF LITERATURE

Different techniques for eye-tracking, its detection, movement, etc. have been used for different implementations.

One of the papers provides a robust reimplementation of the system described by Grauman *et al.* implemented in the BlinkLink blink detection system that is able to run in real time at 30 frames per second on readily available and affordable webcams. [5]

The other experiments involve the use of neural networks for eye-detection. The intensity of each pixel in the image of the eye was used as an input to the neural network. The network's two outputs corresponded to the X and Y locations of the user's gaze on the screen. Assuming linear separability would not pose a problem, a feedforward, two-layer neural network is employed. [7]

The prototype of yet another thesis is subdivided into two parts—hardware and software implementation. Eye-GUIDE focuses on video-based gaze tracking, which consist Hannah camera that records the eye of the user and a computer. The webcam can be used with or without infrared (IR) illumination. For the software implementation the proponents used three different softwares for eye gazing, eye clicking and messaging system. The eye gazing software extracts eye features like pupil and iris centre from the image captured by the web camera. For clicking purposes, the Eye GUIDE Clicker software was introduced. This tool allows user to click templates, keyboard letters and supporting buttons on the Eye GUIDE Messenger provided by this study by hovering the cursor several seconds over the desired button or icon. Finally, software for this messaging system allows the user to select from the provided templates, what the user wanted to say. [1]

One of the research involves the Houghman circle detection algorithm for eye-tracking. This basic algorithm processes the input video frames from the camera to detect the cornea. Then this position is compared to the centre point calibrated initially using a square grid on which the algorithm is applied. This calculates the angle and the speed at which the mouse should move [4].

Based on the evaluation of the above techniques, along with the other evaluations, setup using a simple web camera and implement the Fabian Timm algorithm was decided to be set up, the details of which are mentioned in the following section.

III. METHODOLOGY

A) Set-up

1) Hardware set-up

- An economical web camera

Currently used cameras for video chat applications support high-definition (HD) images with high resolutions up to 1920x1080 pixels. Smartphones and tablets now have an in-built camera which faces the user, and have gained considerable popularity in the recent years. They can provide the necessary baseline for our proposed system. This baseline is defined by resolutions of 1280x720 (iPad FaceTime camera resolution) and 640x480 (the common VGA cameras resolution) [5]. After considering all of these and also the differences in the camera placements and camera qualities, we proceeded for finalizing one model. In order to find an apposite camera that meets the requirements such as good resolution, frame rate, quality of image, etc., we have tested the following cameras: Microsoft HD-3000, Logitech (c270, c615), and other cheaper models of Trust, NIUM, and Hercules. As expected, the expensive models give good HD quality images available at acceptable frame rates (10 fps). After a compromise to match the project's requirements, Logitech c270 HD was finalized.

Fig 1: Logitech c270 HD.

- Display

The display is obviously a crucial component for this implementation. Hence the display screen is kept at pre-known distances away from the subject's face appropriate for the camera's point of view also.

2) Software set-up

- Fabian Timm algorithm

The algorithm used by the system for detecting and analyzing blinks is automatically initialized and it is only dependent upon the involuntary blinking of the user. For doing the same, motion analysis techniques are used, along with creation of a template of the open eye online which is used for further tracking and template.[2]

IV. IMPLEMENTATION

A) Architecture

Initially an image is captured from a webcam using OpenCV's built-in function. This image is duplicated and the duplicate copy is converted to grayscale. This grayscale image will be further used for all processing.

Next, the Haar cascade classifiers are used to detect and track face and eyes. These classifiers come along with the OpenCV library and can be applied using the built-in functions. Then apply Face Haar cascade classifier to detect and localize the face region. After the face region is localized, the face region of interest is used to track eyes, using Eye Haar cascade classifiers. There are various methods for blink detection most of which involve Histograms. However, the Eye Haar cascade classifiers are used instead of Histograms to increase the accuracy of detecting blinks. Once an effective voluntary blink is detected (5-6 frames of continuous blinks), xdotool is invoked through a System call in C++. Xdotool is a tool in linux, which can be used to handle mouse events like click, movement of cursor. The mouse double click is called in xdotool after an effective blink is established. For the iris tracking part, the Fabian Timm algorithm is used. A reference frame is kept with the eyes straight looking at the screen and centered. This reference frame is used to make decisions about mouse movements.

Depending upon what the difference is between the reference frame and the new frame, user can determine in which direction to move the cursor. Cursor is relatively moved in a particular direction using xdotools.

B) Algorithm

Older eye tracking algorithms were highly dependent upon the amount of light, resolution and contrast. Hence, the images which suffered from lower resolution and contrast gave sub-standard tracking. Thus the Fabian Timm algorithm was devised. The advantage of this algorithm is that it works in even poor lighting conditions.

The algorithm first detects the face, then the eyes and then the center of the iris. It tracks the irises in real time thereby detecting where the user is currently looking. It maps the iris position as the change in cursor location on the screen.

- It works by measuring gradients of the iris.
- It uses segmentation of Image Processing.
- It will keep on measuring the gradient using the gradient function. The value of gradient will be maximum for the iris part and that will be it.
- It takes the screenshot the eyes so as to keep it as a backup making the system more robust.
- The gradient functions are implemented using OpenCV libraries.

step 1: capture image from the camera
step 2: use face Haar cascades to detect face
step 3: use eye cascades to localize eye region
step 4: if eye region is detected then proceed to step 5, else go to step 8
step 5: create reference template for the eye
step 6: detect iris after pupil localization
step 7: determine the direction of mouse pointer motion and go to step12
step 8: detect and analyze blink
step 9: if blink < threshold, discard the frame as an involuntary blink, go to step 8
step 10: if the blink > threshold, consider it as a voluntary blink and initiate click
step 11: issue mouse click command
step 12: end

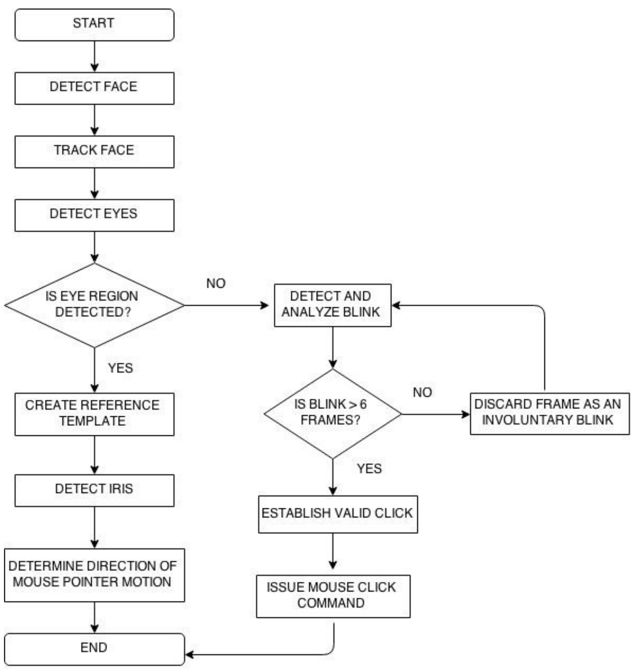

Fig 3: Flowchart of implemented algorithm.

V. RESULTS

Based on the reviews of the various papers studied, the following are the shortlisted parameters that need to be considered for implementation:

• Camera resolution: A trade-off between cost and effective detection is being done while choosing the camera resolution.

• Camera position: The distance of the camera from the subject's eye is very crucial for eye detection. Easier and more accurate detection can be obtained by reducing the distance.

• Stability: Based on the evaluation of other experiments as well as a set-up of our own, it can be averred that the use of a chinrest improves the performance by 32% horizontally as compared to the setup without it. This reduces the variance among the people or subjects, thus increasing the experiment's reliability.

• Subjects: Different people have different eye structures causing differences in their eye detections. Depending if they use eyeglasses or not, their head movements during the experiment, their involuntary blinks, the output may vary. The aim to create a model which is minimally affected by these factors is met.

• Lighting conditions: This is an important factor as it affects not only the image quality but also the camera frame rate. Based on our research, it is perceived that this factor may have an effect on corner detection algorithms and part detectors, as disturbances like shadows and noise can appear in the webcam image.

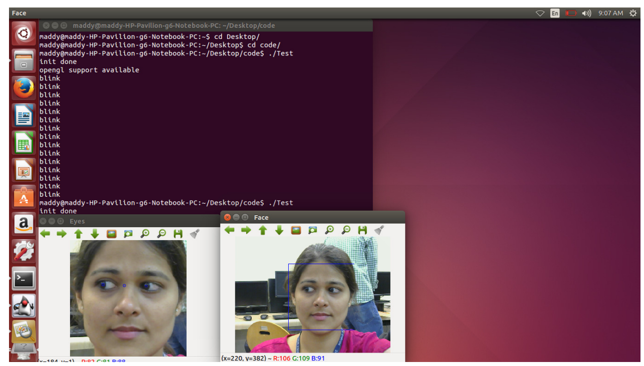

Fig 4: Implementation GUI.

VI. CONCLUSION

The paper has successfully demonstrated that the prototype can track user eye features and detect the pupil under various environments with complex background having clutters. These two variables can also be acquired with various distances and various angle of the face of the user being tracked. The system is calibrated enough to ensure that the user is able to move the cursor through his eye so that the system can track its movement.

The system shows that it has a potential to be used as generalized user interface in many applications such as determining web usability in heat maps. A heat map is a graphical representation of data where the individual values contained in a matrix are represented as colors.

There are many unexplored aspects of the system which are waiting for improvements. We believe there is still much work to do to take the achieved performance levels even further.

ACKNOWLEDGMENT

We have great pleasure in presenting this paper on "Real time eye tracking using web camera". We take this opportunity to thank all those who have contributed in successful completion of this report.

We wish to express sincere thanks and deep sense of gratitude to respected mentor Asst. Prof. Ruhina Karani**,** in Computer Engineering department of D.J. Sanghvi College of Engineering, for her support with all of her kindness. We are also thankful to our parents for their continuous encouragement to pursue higher studies and our friends for the help and support they provided.

References

[1] Rommel Anacan, James Greggory Alcayde, Retchel Antegra, Leah Luna, "Eye-GUIDE (Eye-Gaze User Interface Design) Messaging for Physically-Impaired People", International Journal of Distributed and Parallel Systems (IJDPS) Vol.4, No.1, January 2013.

[2] Fabian Timm, Erhardt Barth, "Accurate Eye Centre Localisation By Means Of Gradients", Institute for Neuro- and Bioinformatics, University of L¨ubeck, Ratzeburger Allee 160, D-23538 L¨ubeck, Germany, Pattern Recognition Company GmbH, Innovations Campus L¨ubeck, Maria-Goeppert-Strasse 1, D-23562 L¨ubeck, Germany.

[3] Weston Sewell, Oleg Komogortsev, "Real-Time Eye Gaze Tracking With an Unmodified Commodity Webcam Employing a Neural Network", Proceedings of ACM Conference on Human Factors in Computing Systems (CHI), Atlanta, GA, 2010.

[4] Reji Mathews, Nidhi Chandra, "Computer Mouse using Eye Tracking System based on Houghman Circle Detection Algorithm with Grid Analysis", International Journal of Computer Applications (0975 – 8887) Volume 40– No.13, February 2012.

[5] Michael Chau, Margrit Betke, "Real Time Eye Tracking and Blink Detection with USB Cameras", Boston University Computer Science Technical Report No. 2005-12, May 12, 2005.

[6] Onur Ferhat, Fernando Vilariño, "Eye-Tracking with Webcam-Based Setups: Implementation of a Real-Time System and an Analysis of Factors Affecting Performance, Master in Computer Vision and Artificial Intelligence Report of the Master Project Option: Computer Vision".

[7] Erna Demjén, Viliam Aboši, Zoltán Tomori, "Eye Tracking Using Artificial Neural Networks For Human Computer Interaction", Physiological Research Pre Press Article.