ENHANCED CURSOR CONTROL USING EYE MOUSE

[1]MARGI SADASHIV NARAYAN, [2]WARANG PRAVIN RAGHOJI

[1,2]Department of computer Engineering, SSPM's College of Engineering, Kankavli, Maharashtra, India

Abstract— In today's era, the combination of Iris Tracking and gaze estimation shows a person's interest. Cursor control using Eye Mouse is process of performing mouse function using Iris Tracking. Iris Tracking is the process of determining the point of gaze or the motion of an eye. In this paper, we have focused on a web-camera-based gaze estimation algorithm. The proposed algorithm describes the implementation of both iris and movement of cursor according to iris position which can be used to control the cursor on the screen. We are proposing combined use of Viola-jones algorithm and Houghman Circle Detection Algorithm to make existing system faster and more efficient. Eye mouse will helps the people who are suffering from decease like Amyotrophic lateral sclerosis.

Index Terms— Iris Tracking, gaze estimation, human-device interaction, Amyotrophic lateral sclerosis.

I. INTRODUCTION

Eye location, iris tracking and the related visual gaze estimation are important tasks in many computer vision applications and research. Eye tracking is a technique where the position of the eye is used to determine gaze direction of a person at a given time and also the sequence in which they are moved (Poole & Ball, 2006). Some of the most common examples are the application to user attention and gaze estimation in driving and marketing framework, and control devices for disabled people. Eye position and iris tracking techniques can be divided into three distinct modalities: (1) Electro oculography, is used for recording the electric potential differences of the skin surrounding the ocular cavity; (2) Scleral contact lens/search coil, has a mechanical reference mounted on a contact lens, and (3) Photo/ Video oculography, has an image processing technique to locate the centre of the eye. Miserably, the most common problem of the above techniques is the use of invasive and expensive sensors. While photo/video oculography is considered the least invasive of the methods, commercially available trackers still require the user to be either equipped with a head mounted device, or to use a high resolution camera combined with a chinrest to limit the allowed head shift. Moreover, daylight applications are prohibited due to the common use of active infrared (IR) illumination, used to obtain accurate eye location through corneal reaction. Non infrared appearance based eye locators [15, 9, 7, 6, 5] can successfully locate eye regions, yet are unable to track eye movements accurately.

The goal of this paper is to present an iris tracking technique alternative to the infrared illumination which is based on the visible light spectrum analysis. Eye trackers working in visible light are often called infrared-free (IR-free) eye trackers. A complete procedure is presented in this paper that tracks the iris and moves the mouse from one place to another on desktop through user's eyes movement. This paper consists of: (1) an eye tracker that can quickly and accurately locate and track iris in low resolution images and videos (i.e., coming from a simple web cam); (2) the Iris tracking is done by the system and error rate is shown and then comparison is done with database; and (3) third part shows accuracy and precision value.

II. RELATED WORK

In this section, we introduce related work.

As gaze estimation algorithms, corneal reaction based methods are popular. They employ infrared (IR) illumination mounted near the camera to illuminate the eye region and estimate gaze direction by detecting the pupil positions and the reflected illumination images on the cornea (Purkinje images). Since Purkinje images are very small, high-resolution images of the eye region are required. In addition, for easy detection of Purkinje images and the pupils, the eye regions must get enough bright illumination. Therefore, these methods have a limitation: they require a very short distance (lessthan1m) between user and camera (IR illumination).

As gaze-tracking methods that require no calibration process, Wang et al. and Wu et al. proposed a "circle algorithm" method in which irises are observed as ellipses to estimate gaze direction by ellipse shape. Ellipses are fitted to the observed iris regions to estimate gaze direction by fitting the ellipse parameters. The circle algorithm doesn't need any calibration process. However, for accurate fitting of ellipses, high-resolution images are required. In addition, estimation accuracy worsens when the angle between the gaze and camera directions is small (users are looking in the camera direction), since in that case the irises are observed as nearly circles.

The eye tracking system based on the IR illumination is often used in head mounted and hands-free interfaces. A gaze estimation algorithm based on the facial feature tracking is proposed. By solving the equations, the 3D visual axis can be estimated, and then the gaze point on the screen can be obtained by intersecting visual axis with the screen. The preliminary experiment shows that this method can achieve the accuracy under 3 degree under free head movement.

In this paper, we propose a single-camera-based gaze estimation algorithm. Here we tried to show that the iris can also be used for gaze estimation and it can be detected easily. The objective of this work is to devise a simple web-cam based iris tracking algorithm that can provide reasonably good accuracy without using IR Illumination or any sensor camera.

III. PROPOSED METHOD

A. Overview of Proposed Method

The developed iris tracking system using a single web-cam without infrared illumination, detects iris position by the two processes. First, the Viola-jones algorithm detects the face. Then the iris gets tracked using Houghman Circle Detection Algorithm and values are stored in database. Then, the gaze position is computed by using calibration points on the screen after that using the eyeball model comparison is done between the input images with the stored database image. The accuracy and precision are measured by looking at the calibration points on the screen.

B. Process flow of proposed method

The process flow of the proposed method that consists of three processes: facial-feature detection/tracking, eye model estimation and iris tracking, and gaze estimation using calibration points. The facial-feature detection/tracking process detects facial features that are used in both face/eye model estimation and gaze estimation processes. First, we detect the face position in the image using a face detection method based on the Viola-Jones algorithm which uses a complex combination of Haar features for object detection. In the eye model estimation and iris tracking process, first we capture N images for face/eye then using Haar cascade object detector extract the necessary features.

IV. SYSTEM ARCHITECTURE

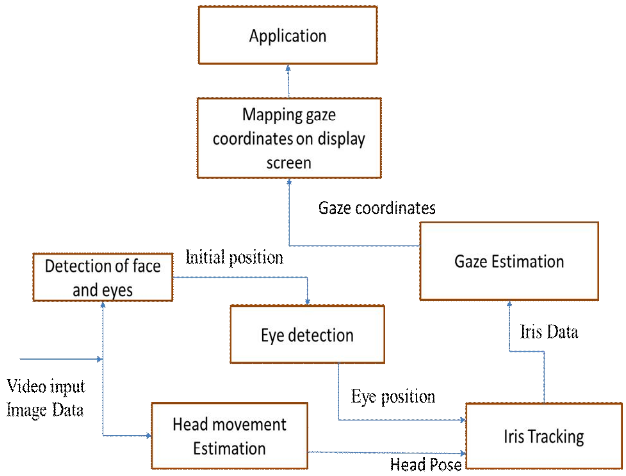

Fig 1: System Architecture

The system is divided into three components:
1) Face detection and tracking component
2) Eye tracking component
3) Gaze to screen coordinates mapping component

The proposed system uses a camera with 20 megapixels (480 pixels- interpolated 20M pixels still image resolution, interpolated 2.1M pixels video resolution) to capture the images of the user for iris tracking and gaze estimation. The flow chart of the proposed system is depicted in Figure1. The first image is used for initial face location and eye detection. If any one of these detection procedures fails, then go to the next frame and restart the above detection processes. If iris tracking fails, the processes of face location and eye detection restart on the present image. These procedures continue until there are no more frames. The detailed steps are described in the following subsections.

V. FACE AND EYES DETECTION

A. Face Detection

The Viola-Jones Algorithm has been used for detecting the face in a live video. It uses a complex combination of Haar features for object detection. The technique relies upon placing a sub frame of

24x24 pixels within an image, and afterwards storing rectangular features inside it in every position with every size possible. These features can consist of two, three or four rectangles. Some examples of such features are shown below in figure2. Open CV has cascades for faces which have been used for detecting faces in live videos.

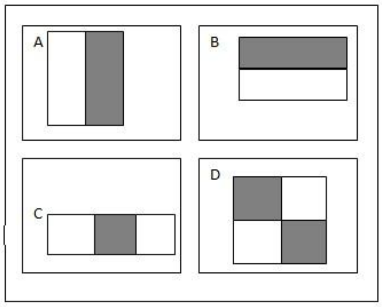

Fig 2: Haar Features

1) Figure2 shows 2 two-rectangle features, 2 three-rectangle features and 1 four-rectangle feature. The idea behind this is that we are observing not explicitly the color or luminance but the difference between the highlighted rectangles.

2) Feature type A cannot be defined as a 1x1 pixel block; it must at least 1x2 pixels. Here, type D must be at least 2x2 pixels, and this rule holds accordingly other features holds this rule.

3) Feature type A cannot be defined as 1x3 pixel block as partition is not possible in middle pixel, and by subtracting it from itself is same to a 1x2 pixel block; even widths define this type of feature. Also, the feature type C width must be divisible by 3, and other features hold this rule.

4) Feature with a width and/or height of 0 cannot be defined. Therefore, iteration is required for x and y to 24 minus the size of the feature. .

B. Eyes Detection

The first step of detection of the gaze is to locate the eye regions, in order to extract the necessary features. The fundamental requirement of an iris tracking and gaze detection system is to accurately detect the eye sockets, which can easily be achieved by Haar-like object detectors. It allows a classifier trained with sample views of a particular object to be detected in a whole image. An eye image is captured with a zoom-in camera of high resolution. This provides iris images with more number of pixels.

C. Center of IRIS is tracked by using Houghman Circle Detection Algorithm

Hough circle algorithm which detects all the circles in the image and draws them in the accumulator variable. It was experimentally noted in the experiments that the circle with largest radius detected in the frame matched the boundary of the pupil. The other smaller circles were detected due to the incidental arcs detected by the eyelashes on the upper and lower eyelids. These circles were ignored by setting a threshold size of the circle radius which helped in filtering out smaller circles from processing. The accumulator variable stores the higher numerical value in the array with a colour having brighter intensity as shown in the figure.

The above process returns the coordinates of the cornea in the frames of the live video from the webcam. The next step in the process is to determine the current live location of the centre detected with respect them to centre point of the frame where the centre resides when the user looks at the object keeping the face aligned to the object such that the normal vector to the face plane passes through the object. Consider the following virtual grid we have used to implement the cursor control system.

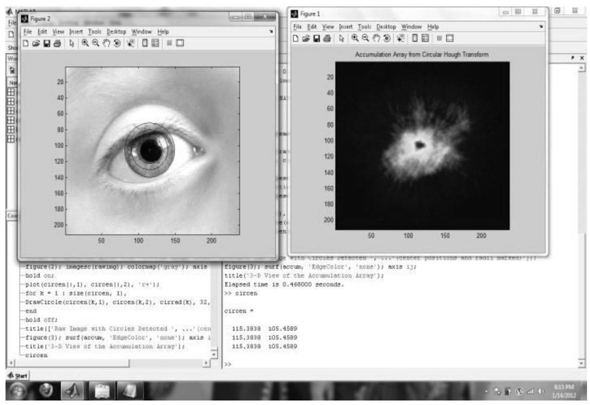

Fig 3: Eye Detection

When the user moves his eyes, the image processing function returns the new coordinates of the center and the relative position of the new coordinate with respect to the center point are calculated which derives the equation of a line using the following formula.

$$y-y1 = [(y2-y1)/(x2-x1)]/(x-x1)$$

This equation can be used to initiate the mouse movement from current point in the direction where the user gazes. The following algorithm can be used to perform the mouse movement. We assume an N by N square grid to be used in the process.

i. Repeat steps ii to iv while the cornea lies outside the center region of the grid. (Denotes active mouse movement)

ii. Obtain the equation of the line in the form y= m x + c by using the coordinates at the center of the grid (x0,y0) and current detected coordinate of the center of the eye (xd,yd) and substituting in the equation.

iii. Keep initial coordinates (x0, y0) as (N/2, N/2)

iv. Calculate value y0,y1, y2, y3… by substituting values for x1,x2,x3 into the line equation calculated in step ii using values between (x0 and xd) and simultaneously move the cursor to position [(x1,y1), (x2,y2), (x3,y3) …..(xd,yd)] .

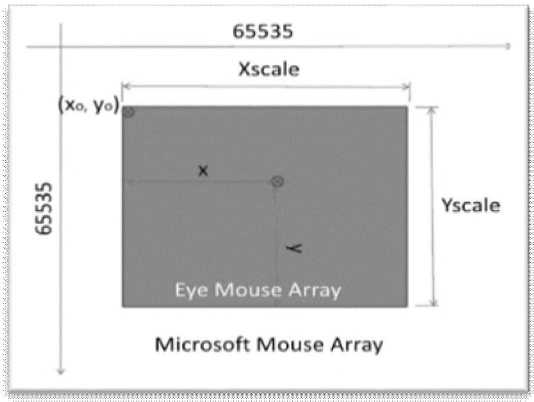

Fig 4: grid mapping

D. clicking event

The average **time** it takes for a complete **human blink** is about 300 to 400 milliseconds or 3/10ths to 4/10ths of a second. Of course this is an average only and can differ from person to person. So to perform clicking operations, user has to blink with one eye for at least 700 to 800 milliseconds. In that period viola-jones algorithm will detect only one eye. If left eye is detected then it will be left click. If right eye is detected then it will be right click. If both eyes are not detected then it will consider that it is normal blink.

Right Eye	Left Eye	Action
True	False	Left Click
False	True	Right Click
True	True	Moving Mode
False	False	Blink

Fig 5: clicking conditions

VI. PROPOSED ALGORITHM

A complete procedure is presented that tracks the iris and moves the mouse from one place to another on desktop through user's eyes movement. Before the processing for the movement of cursor begins, detailed processing is presented below:

1. Camera receives the input from the eye.
2. After receiving these streaming videos from the camera, it will break into frames.
3. After receiving frames, it will check for lighting conditions because cameras require sufficient lights from external sources otherwise error message will be displayed on the screen.
4. Images (frames) from the input source focusing the eye are analyzed for Iris detection (center of eye).
5. Finding the center of the iris. This step forms the major part of the algorithm. The algorithm first finds an 'eye window' – a rough estimate of the eye using voila-jones algorithm.
6. The second step is to find the exact position of the iris within the eye window using Houghman Circle Detection Algorithm.
7. Mapping the iris to a point in the video from the scene camera. Using pre-determined calibration points, the position of the iris is mapped to a position on the screen.
8. After this, a mid-point is calculated by taking the mean of left and right eye center point.
9. Finally the mouse will move from one position to another on the screen.

CONCLUSION AND FUTURE WORK

An IC localization method based on the SEM (spherical eyeball model) was proposed. Proposed method is more appropriate for visible image-based gaze tracking systems due to its superior performance of the IC localization compared with the conventional methods. It is applicable to emerging head-pose-free gaze tracking systems. It promotes a wider use of gaze tracking technology in consumer electronics.

The gazing point found out in the proposed method is carried out using a single web-cam without any infrared camera and sensors. The results are promising with the best result with iris tracking and gaze estimation.

The conclusion of this method is that an efficient and accurate gaze tracking system can be created with a simple web camera. Future work includes implementation of iris tracking on mobile phones.

The system shall be refined to increase the accuracy without using calibration points. When the system is mounted on a moving vehicle, vibration causes the captured image to become blurred. This in turn decreases the tracking rate. These issues need to be addressed.

ACKNOWLEDGMENT

I would like to express heartfelt gratitude towards my faculty guide Mr. R.R.Bane, Assistant .Professor, Computer Science & Engineering Department, SSPM's College of Engineering Kankavli, Maharashtra, India for his guidance and help in completing this work successfully with good results. I would also like to express my gratitude to my teacher Mr. Dagare Sir for his timely support and help.

REFERENCES

[1] Reji Mathews,Nidhi Chandra, Amity University, Noida,UP," 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

[2] W. Zhang, T.-N.Zhang, and S.-J. Chang, "Eye gaze estimation from the elliptical features of one iris," *Opt. Eng.* , vol. 50, no. 4, pp. 047003-1-9, Apr. 2011.

[3] Alexander Kadyrov, Hui Yu,**"** Explore New Eye Tracking and Gaze

[4] Locating Methods", 978-1-4799-0652-9/13 $31.00 © 2013 IEEE

[5] Jixu Chen Qiang Ji "3D Gaze Estimation with a Single Camera without IR Illumination",978-1-4244-2175-6/08/$25.00 ©2008 IEEE [5] Fares Alnajar Theo Gevers "Calibration-Free Gaze Estimation Using Human Gaze Patterns", *ICCV* 2013 paper.

[6] J Ali Javed, Zeeshan Aslam "An Intelligent Alarm Based Visual Eye Tracking Algorithm for Cheating Free Examination System", *I.J. Intelligent Systems and Applications*, 2013, 10, 86-92, [online] Published Online September 2013 in MECS (http://www.mecs-press.org/)

[7] Dongheng Li, David Winfield, Derrick J. Parkhurst "Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches"

[8] Lalonde M, Byrns D, Gagnon L, Teasdale N, Laurendeau D :"Real –time eye blink detection with GPU-based SIFT tracking",Fourth Canadian Conference on Computer and Robot Vision, pp.481-487(2007).

[9] M S. Miyazaki, H. Takano, and K. Nakamura, "Suitable Checkpoints of FeaturesSurrounding the Eye for Eye Tracking Using Template Matching,"Proc. SICE Annual Conference 2007, Sept.17-20, pp.356-360, 2007.

[10] Roberto Valenti, Jacopo Staiano," Webcam-Based Visual Gaze Estimation" Faculty of Science, University of Amsterdam, The Netherlands,ICIAP 2009, LNCS 5716, pp. 662–671, 2009. c Springer-Verlag Berlin Heidelberg 2009.

[11] Yuan-Pin Lin, 1 Yi-Ping Chao, 2 Chung-Chih Lin, and 1 Jyh-Horng Chen "Webcam Mouse Using Face and Eye Tracking in Various Illumination Environments" Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference Shanghai, China, September 1-4,2005

[12] S. Asteriadis, N. Nikolaidis, A. Hajdu, I. Pitas "An Eye Detection Algorithm Using Pixel to Edge Information"

[13] Shang-Lin Wu, Lun-De Liao, Shao-Wei Lu, Wei-Ling Jiang, ShiAn Chen and Chin-Teng Lin, "Controlling a Human-Computer Interface

[14] System With a Novel Classification Method that Uses ElectrooculographySignals," IEEE transactions on biomedical engineering,vol. 60, no. 8, pp. 2133-2141, August 2013.

[15] A. Lopez, 1. Rodriguez, FJ. Ferrero, M. Valledor and lC. Campo,"Low-Cost System Based on Electro-oculography for Communication of Disabled People," Proceedings of the IEEE II th International Multi Conference on Systems, Signals and Devices, Feb 2014, pp. 1-6.

[16] Robert Gabriel Lupu, Florina Ungureanu and Valentin Siriteanu, "Eye Tracking Mouse for Human Computer Interaction," Proceedings of the 4th IEEE International Conference on E-Health and Bioengineering - EHB, Nov 2013, pp. 1-4.

[17] Paul Viola, Michael J.Jones, "Robust Real-TimeFace Detection".International Journal of ComputerVision 57(2), 137–154, 2004.

[18] Y. Tsuge, H. Takano, M. Kurokawa, and K. Nakamura, "Non-contact Pointing Device Using Real-time Eye Tracking," SICE Annual Conference 2005, August 8-10, pp.1521-1525, 2005.

★★★