Chapter 54
Comparing Two Webcam-Based Eye Gaze Trackers for Users with Severe Speech and Motor Impairment

Ayush Agarwal, DV JeevithaShree, Kamalpreet Singh Saluja, Atul Sahay, Pullikonda Mounika, Anshuman Sahu, Rahul Bhaumik, Vinodh Kumar Rajendran and Pradipta Biswas

Abstract This paper aims to develop and evaluate two different webcam-based gaze-controlled interfaces for users with severe speech and motor impairment (SSMI). We configured two webcam-based gaze trackers using open-source software (Python and JavaScript) and developed cursor control algorithm using the gaze tracker. We designed a quiz application to evaluate the webcam-based gaze trackers for both users with SSMI and their able-bodied counterparts. We also collected data using a commercial infrared-based eye gaze tracker. We noted that users with SSMI and able- bodied users could use the webcam-based gaze-controlled interface. It was found that for users with SSMI, speed of

---

D. JeevithaShree · K. S. Saluja · P. Mounika · A. Sahu · R. Bhaumik
V. K. Rajendran · P. Biswas (✉)
CPDM, Indian Institute of Science, Bangalore, India
e-mail: pradipta@iisc.ac.in

D. JeevithaShree
e-mail: jeevithas@iisc.ac.in

K. S. Saluja
e-mail: kamalpreets@iisc.ac.in

P. Mounika
e-mail: pulikondam@iisc.ac.in

A. Sahu
e-mail: ansumansahu@iisc.ac.in

R. Bhaumik
e-mail: rahulbhaumik@iisc.ac.in

V. K. Rajendran
e-mail: vinodhr@iisc.ac.in

A. Agarwal · A. Sahay
Indian Institute of Information Technology, Kalyani, India
e-mail: ayush@iiitkalyani.ac.in

A. Sahay
e-mail: atulsahay01@gmail.com

A. Chakrabarti (ed.), *Research into Design for a Connected World*,
Smart Innovation, Systems and Technologies 135,
https://doi.org/10.1007/978-981-13-5977-4_54

interaction was significantly faster for a low-cost infrared-based commercial gaze. Results from this study can be used to develop as well as select low-cost eye gaze trackers for users with SSMI. This might be the first study to evaluate webcam-based gaze trackers in a gaze-controlled interface for users with different range of abilities

54.1 Introduction

Eye gaze trackers are devices used to track eye gaze movements. Before digital computers, they were made of metal contact lenses. With advancement of computing technology, recent eye gaze trackers make use of computer vision algorithms to detect eye ball movement from digital image in real time and use that to predict points of fixation and saccadic gaze movement. Existing commercial eye gaze trackers mostly used infrared camera and bright or dark pupil technique [1]. The bright pupil technique allows image processor to locate centre of pupil. The eye gaze tracker can then locate where the person is looking on the screen based on the relative positions of the pupil centre and corneal reflection within the video image of the eye. However, there also exist eye gaze trackers that utilise cameras in visible spectrum like webcam or high-speed video cameras although those systems are either less accurate (for webcam) or costlier (for high-speed video cameras) than infrared trackers.

Traditionally, eye gaze trackers are used to record and analyse eye gaze movement for reading, visual search, Web browsing and similar tasks involving electronic or non-electronic displays. As eye gaze trackers became able to stream data in real time with negligible latency and same frequency of electronic displays, eye gaze trackers found application as a direct controller of graphical user interfaces. In a gaze-controlled interface, eye gaze movements were either used to facilitate another pointing modality by zooming in part of display or it is used to draw cursor on screen based on the points of eye gaze fixations. A detailed review of different gaze-controlled interfaces can be found in different papers [2].

Gaze-controlled interface found important application for people with different range of abilities where physical impairment impedes use of other input modalities like mouse, touchpad, touch screen or keyboard. There is already a plethora of commercial products [3, 4] available for electronic gaze-controlled interface. Most research for children with cerebral palsy was concentrated on developing applications like augmentative and alternative communication aid, menu structure [5–7], home automation application [8] and so on. Biswas [2] reported a detailed literature survey on state-of-the-art on gaze-controlled interfaces, and it may be noted that gaze-controlled interface require either bigger button size and arrangement [6, 7] or automatic zooming feature [9] or coupling with another interaction device [10] to accommodate inaccuracy in gaze tracking.

This paper proposes and compares two different webcam-based gaze trackers for a set of users with severe speech and motor impairment (SSMI) for an online quiz application. Commercial eye gaze trackers have the advantage of higher accuracy but those need to be separately bought and configured for individual computers. Webcam-based eye gaze trackers are far less accurate than infrared-based commercial ones, but if it is found useful even for a limited set of applications, those can be used without the need of buying or configuring any extra hardware.

In particular, we have compared performance of two gaze trackers both of which are platform independent and developed using open-source software. The first one uses landmark detectors through an OpenCV graphics library written in Python programming language, and the other one uses webgazer.js JavaScript software. We developed bespoke software to control a mouse pointer using eye gaze and an online quiz application with limited screen elements to compare performance of the gaze trackers. In the next section, we presented details on two gaze trackers. Initially, we collected data from able-bodied participants and then repeated the study with users with SSMI. We compared the number of participants able to use the system and reaction times of participants.

54.2 Related Work

Webcam-based gaze tracking is not a new concept [11] although deploying such a system for users with severe speech and motor impairment was not widely reported. Most webcam-based systems initially detect face using standard OpenCV library and then based on the relative position of pupil within the standard geometry of eyes estimate gaze position. However, none of these webcam-based trackers are evaluated as extensively as the commercial infrared-based gaze tracker. Khonglah and Khosla [12] reported an eye gaze tracker that uses Viola–Jones [13] detector to detect face and a blob detection algorithm to detect glint from the pupil. However, the system is only tested using a heat map on interfaces having only two targets. Cuong and Hoang's [14] system did not detect face, rather directly detect eyes and tested for only five positions (right, left, straight, up and down) on screen. Sewell and Komogortsev [15] used a feed-forward two-layer neural network to estimate gaze vectors from the images of eyes but already reported problem in extrapolation about training the network while detecting eye gaze for one of 50 random points on screen. The ITU gaze tracker [16] requires a special hardware to hold the webcam near the eyes, and it was evaluated for a typing application by able-bodied users and one motor-impaired user with ten targets on a projected screen. There are also a few commercial webcam-based gaze trackers (like Web gazer, https://webgazer.cs.brown.edu/ or xLabs gaze, https://xlabsgaze.com/) but they are mainly advertised for recoding browsing behaviour of Web users.

We noted from previous studies [17, 18] that that users with SSMI require a minimum size of screen elements, which reduces the density of screen elements. Additionally, with a nearest neighbourhood prediction algorithm, users need not to accurately place pointer on screen elements; the algorithm activates target nearest to the present pointer position. Hence, we can compromise the accuracy of eye gaze tracking. In the following subsections, we presented a low-cost eye gaze tracker that does not require any special hardware, rather can track eye gaze from standard webcam video feed.

54.3 Proposed Approach

54.3.1 First Implementation—Landmark Detection

Initially, we have used a Viola–Jones-type object detector [13] to detect face from the webcam image. After that, we extracted the eye regions from the facial image for further processing. From the face region, the eyes were extracted. We noted that a Viola–Jones-type detector [13] is precise but expensive in terms of computation time for detecting eye region. So, we used a quicker method based on the dimension of the face and relative position of eyes in the face as eyes are at certain fixed proportions to our face (Fig. 54.1). Once we extracted the eye region, we scaled it down to smaller size to increase computation speed. For every video frame, the eye landmarks were detected and subsequently used to measure the eye aspect ratio (EAR) between the height and width of the eye.

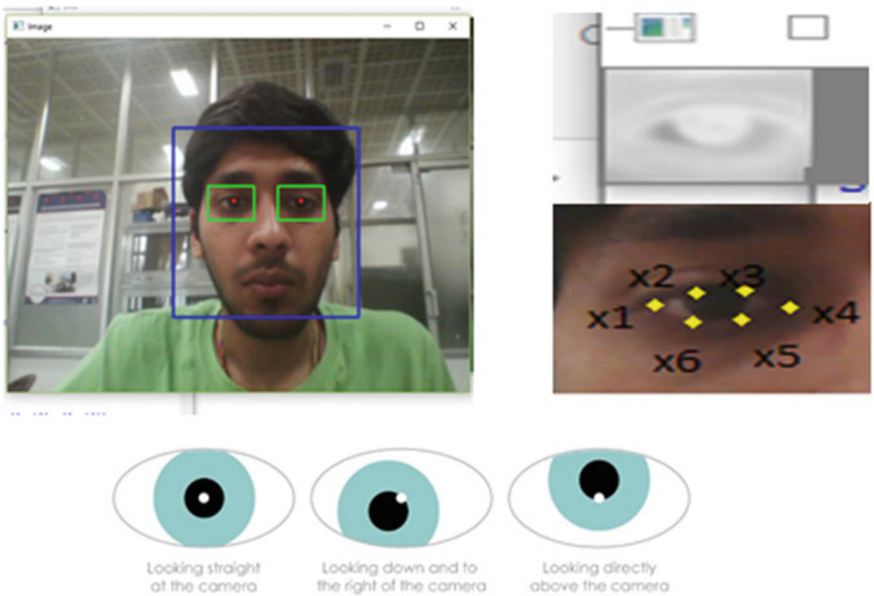

Fig. 54.1 Detecting eye gaze from webcam

$$\mathrm{EAR} = \frac{\|x2 - x6\| + \|x3 - x5\|}{2 * \|x1 - x4\|}$$

The 2D landmarks are depicted $x1\ldots x6$ in Fig. 54.1. It may be noted that EAR remains constant irrespective of the distance between the user and the camera module. With the knowledge of the EAR and displacement of the pupil centre from the reference point, we predicted the nine directions of the gaze (top left, top middle, top right, left, centre, right, bottom left, bottom middle and bottom right). To detect the pupil position, initially we applied a threshold value on the image matrix representing the eye region based on the maximum value, and remove all the remaining values that are connected to the image borders and then find out the maximum value from the remaining set of values. We selected the pupil centre having the highest summation value of their neighbouring pixels.

A linear transformation function has been used to map the EAR and the displacement value of the iris form the centre to screen coordinates. The eye gaze was estimated by calculating mode from a number of EAR values measured in continuous frames. More details on this implementation are described in a different paper [19].

54.3.2 Second Implementation—Using Webgazer.Js

We have used an eye tracking library written entirely in JavaScript, i.e. webgazer.js (https://webgazer.cs.brown.edu/), that uses common webcams to infer the eye gaze locations of users on a web page in real time. WebGazer is an online eye tracker that uses common webcams already present in laptops to infer the eye gaze locations of Web visitors on a page in real time. The eye tracking model self-calibrates by watching users interacts with web page and trains a mapping between features of the eye and positions on the screen. It has two key components, a pupil detector that can be combined with any eye detection library, and a gaze estimator using regression analysis informed by user interactions. WebGazer technology is compatible with three open-source eye detection libraries for locating the bounding box of user's eye. The eye detectors that are evaluated in WebGazer are clmtracker, js-objectdetect and tracking.js. It can also be generalized to include others. There are two gaze estimation methods in WebGazer, one which detects the pupil and uses its location to linearly estimate a gaze coordinate on the screen, and a second which treats the eye as a multi-dimensional feature vector and uses regularized linear regression combined with user interactions. As the number of gaze locations we were getting in a particular time period through webgazer.js was high, so we had taken the mean of last eight points from webgazer.js for better target prediction and accuracy of system. Using the mean value of gaze location for last eight points, we have designed the following algorithm to select five screen locations on a web page through the webcam and webgazer.js.

54.3.3 Controlling Pointer with Inaccurate Eye Gaze Tracker

Both of the proposed eye gaze trackers were not as accurate as compared to commercial eye trackers. We have developed an algorithm to control a graphical user interface through inaccurate eye tracker. Box 1 shows the algorithm to activate one of five elements in a screen using eye gaze. In Algorithm 1, w and h represent the inner width and inner height of a window of a browser. We have taken five points on the screen by initializing appropriate values for dx and dy. The algorithm tracks the nearest screen element from the current gaze position. If the nearest screen element remains same for a particular time interval (dwell time), that element is selected. The duration of dwell time was configurable and set to 1.5 s by default.

Algorithm 1: Controlling the Gaze Point for a screen with five element

function TargetSelection (X, Y);
Input : Mean Prediction of last eight gaze locations as X and Y
Output: Selection of a particular screen element
Initialize dx and dy // We initialized $dx = 300$ and $dy = 200$
$points = [(x : dx, y : dy), (x : w - dx, y : dy), (x : dx, y : h - dy), (x : w - dx, y : h - dy), (x : w/2, y : h/2)]$
Calculate Euler Distance with each points
n = minimum(Euler Distances Point)
if *Minimum Point is n for dwell time* **then**
  Select corresponding screen element
else
  Again start from next minimum point
end

54.3.4 Application Used to Evaluate Gaze Trackers

We have evaluated the webcam-based gaze tracker for a representative application involving children with severe speech and motor impairment due to cerebral palsy. As the webcam-based gaze tracker had limited accuracy, we designed an online quiz application that had only four screen elements at any point. The quiz application displayed a question in middle of the screen, and the four options were rendered as pictures at the four corners of the screen (Fig. 54.2). The questions were

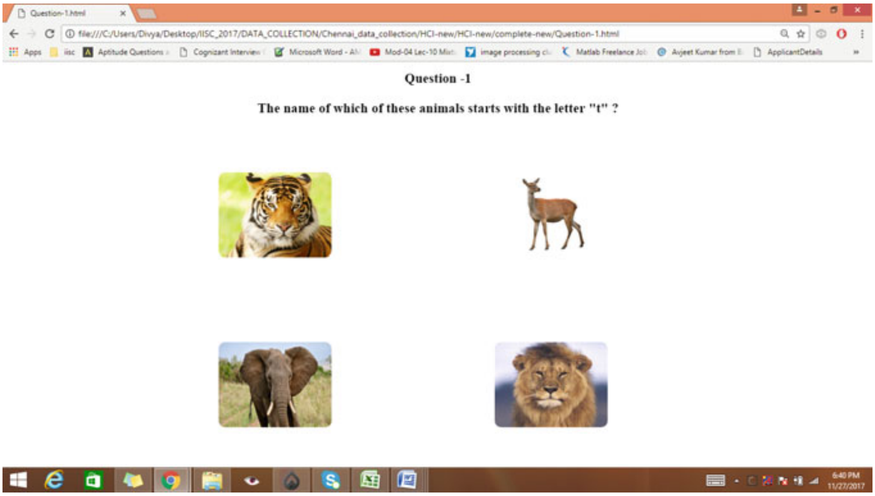

Fig. 54.2 Screenshot of quiz application

designed as simple and interesting for our end-users, who were teenage students. To compare performance of the webcam-based gaze tracker, we also collected data using a low-cost commercial infrared-based gaze tracker. As control group, we also collected data from able-bodied users using the webcam-based gaze tracker with the online quiz application. The following subsections furnished details on the study.

54.4 User Studies

Initially, we evaluated both gaze trackers for able-bodied users and subsequently for users with SSMI.

54.4.1 Study with Able-Bodied Users

Participants: We collected data from seven participants (five males, two females, average age 26.3 years). They have 6/6 corrected vision and do not have any cognitive and motor impairment. All participants were recruited from our university.

Material: The user trial was conducted using a HP Spectre laptop with Intel i7 core processor, 8 GB RAM and running Microsoft Windows 10 operating system. The laptop has a HP TrueVision HD 1.3 MP webcam, which was used to estimate gaze direction.

Design: The trial consisted of two conditions:

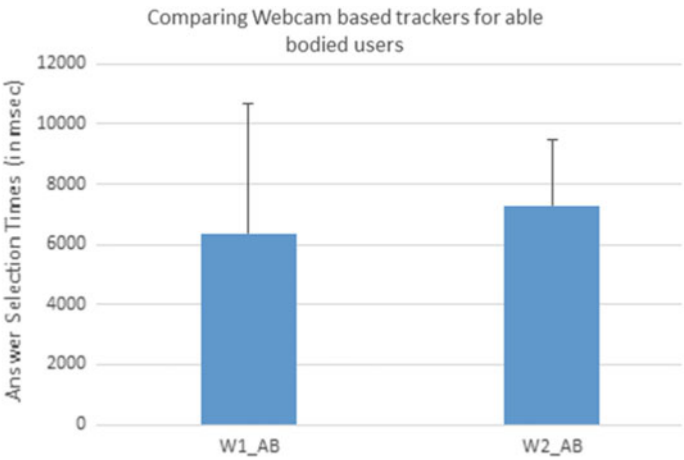

Fig. 54.3 Comparing answer selection times by able-bodied users

1. Using landmark-based webcam-based gaze tracker
2. Using webgazer.js-based webcam-based gaze tracker.

For all trial conditions, we used the online quiz application discussed before. The application consisted of ten multiple choice questions, each question had four answer choices and only one was correct answer. All participants practised the software before undertaking the trial. We also pointed the correct answer during the trial. We recorded timestamps of selection of answers. Order of conditions was randomized to minimize practice or learning effect.

Results: We calculated the answer selection times for each question for all participants. The average answer selection times were statistically significantly lower for the landmark-based gaze tracker compared to the webgazer.js library [$U = 1303$, $z = 3.04$, $p < 0.01$, Fig. 54.3].

As all users could undertake trials using both implementations, we went forward to collect data from users with SSMI, as described in the next section.

54.4.2 User Study with Children with SSMI

In this study, we compared the two different implementations of the webcam-based eye gaze tracker. We also included a low-cost commercial eye gaze tracker for comparing performance of the webcam-based gaze tracker.

Participants: We collected data from 11 participants (seven males, four females, average age 17.2 years). Our participants were quadriplegic due to cerebral palsy and were keen to learn operating computer. The participants were secondary students at *The Spastic Society of India* in Chennai. All trials and interactions with them were undertaken under observation by their care takers and school instructors. All necessary permissions were taken before undertaking user trials. We took help from their teachers, who are rehabilitation experts, to evaluate their physical conditions. According to Gross Motor Function Classification system (GMFCS), they were all at level 5 as they could not move without wheelchair. According to Manual Ability Classification System (MACS), some of them were at level 4 and rest were at level 5. A few of them could manage to move their hand to point to a non-electronic communication chart and others only relied on eye pointing. According to Communication Function Classification System (CFCS), all of them were at level 5 as they could not speak, could make only non-speech sound and communicate only through non-electronic communication board. They did not have access to any commercially available scanning software. Initially, we tried to use a mouse, joystick, trackball and stylus, but they could not manage to undertake any pointing and selection task using any of those devices as they could not make any precise movement using their hands necessary to control those devices. Their teachers and parents informed us that they were accustomed to use eye pointing with non-electronic communication chart.

Material: The webcam-based trial was conducted using a HP Spectre laptop with Intel i7 core processor and running Microsoft Windows 7 operating system. The laptop has a HP TrueVision HD 1.3 MP webcam, which was used to estimate gaze direction. For commercial eye tracker-based condition, we used an Intel NUC computer with dual-core i5 processor and Tobii eyeX gaze tracker [20]. The cursor control algorithm using the Tobii tracker is discussed in a different paper [17]. The display was rendered in an 18” screen.

Design: The trial consisted of three conditions:

1. Using landmark-based webcam-based gaze tracker
2. Using webgazer.js-based webcam-based gaze tracker
3. Using a commercial low-cost eye gaze tracker.

For all trial conditions, we used the online quiz application discussed before. The application consisted of ten multiple choice questions; each question had four answer choices, and only one was correct answer. All participants practised the software before undertaking the trial. We also pointed the correct answer during the trial. We recorded timestamps of selection of answers. Order of conditions was randomized to minimize practice or learning effect. We maintained same distance from camera for all participants, and trials were conducted in a well-lit room with lighting level between 300 and 350 lx.

Results: The average answer selection times in the commercial eye gaze tracker were nearly one-third that of the webcam-based gaze trackers and were significantly lower than both webcam-based trackers. We did not find any significant difference between the two webcam-based gaze trackers for users with SSMI (Fig. 54.4).

It may be noted that for four users, we needed to increase the dwell time of selection from its default value of 1.5 s. For three users, it was set to 2 s and for one

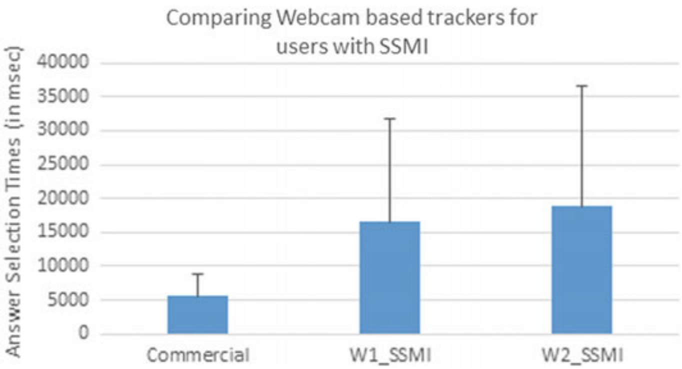

Fig. 54.4 Comparing answer selection times by users with SSMI

it was 3 s. The adjustment was done during the training stage and before start of the actual trial. However, for the commercial gaze tracker, all users could undertake pointing and selection tasks with a dwell time of 500 ms.

Discussion: This study shows that the webcam-based gaze tracker is not yet easily usable by all users with SSMI and the gaze detection algorithm is significantly slower than low-cost commercial gaze tracker. As the screen had only four selectable screen elements, accuracy of the gaze tracker was not the only issue. The latency in processing image and estimate gaze direction increased the pointing times for the webcam-based trackers. Additionally, users needed longer dwell time to select target in webcam-based system compared to the commercial gaze tracker. A follow-up study using a commercial gaze tracker involved users just to fixate gaze at a point on screen and recorded their gaze positions while they were trying to fixate attention [17]. Our study found that users with SSMI could fixate attention although have more uncontrolled saccadic gaze movements than their able-bodied counterparts. The offset did not correlate with screen position or angular deviation of the stimuli. The uncontrolled saccadic movement makes it difficult for webcam-based image processing algorithms to accurately estimate gaze position in a short duration.

Commercial infrared-based eye gaze trackers implement the gaze estimation algorithm in a dedicated circuit resulting less latency in image processing than a shared hardware circuit of a laptop with other programs running on it. Previous studies [17, 18] already indicated presence of nystagmus in people with cerebral palsy, and this nystagmus further increased latency in gaze estimation in webcam-based gaze tracker. Our study indicates that for users with SSMI, a commercial gaze tracker should be preferred over webcam-based ones for making gaze-controlled interface. Webcam-based gaze trackers can be considered to combine with other assistive modalities like single-switch scanning system to increase speed of interaction of the scanning system [21]. However, it may also be noted that the landmark-based gaze tracker worked faster than the webgazer. js-based implementation for able-bodied users, and the average response time was about 6 s. Webcam-based gaze trackers may be a useful alternative input modality for applications, where speed of processing is not a main issue and operators' hands are occupied with different tasks impeding them using traditional touch screen. For example, webcam-based gaze trackers can be considered for operating an electronic display in machines like computer numerical control (CNC) routers. It may be worthy to train webcam-based tracker using supervised machine learning algorithm to quickly detect eye region and run the software on dedicated graphics processing unit (GPU) to reduce latency.

54.5 Conclusion

This paper presents two different systems on developing and configuring a webcam for eye gaze tracking and then used the webcam-based gaze trackers in a gaze-controlled interface for users with severe speech and motor impairment (SSMI). As part of the study, we also implemented an online quiz application with only four screen elements. Our study shows able-bodied users could use the webcam-based gaze tracker to operate the quiz applications, but users with SSMI can operate the system significantly faster with a low-cost infrared-based commercial eye gaze tracker.

References

1. Duchowski, A.T.: Eye Tracking Methodology. Springer, Berlin (2007)
2. Biswas, P., Langdon, P.: Multimodal intelligent eye-gaze tracking system. Int. J. Hum-Comput. Interact. **31**(4), 277–294 (2015)
3. Communicate with the world using the power of your eyes, http://www.eyegaze.com/eye-tracking-assistive-technology-device/ (visited on 8/11/2017.)
4. Eye-gaze control technology, https://www.cerebralpalsy.org.au/about-cerebral-palsy/interventions-and-therapies/eye-gaze-technology-for-children-and-adults-with-cerebral-palsy/#1473737787180-c2ceacf9-ff87 (visited on 8/11/2017)
5. Borgestig, M., Sandqvist, J., Parsons, R., Falkmer, T., Hemmingsson, H.: Eye gaze performance for children with severe physical impairments using gaze-based assistive technology—A longitudinal study. Assistive Technol. **28**(2), 93–102 (2016)
6. Majaranta, P., Majaranta, N., Daunys, G., Špakov, O.: Text editing by gaze: Static versus dynamic menus. In: Proceeding 5th Conference on Communication by Gaze Interaction (COGAIN 2009), pp. 19–24 (2009)
7. Urbina, M.H., Huckauf, A.: Selecting with gaze controlled pie menus. In: Proceedings of the 5th International Conference on Communication by Gaze Interaction (COGAIN 2009), pp. 25–29 (2009)
8. Razzak, F., Castellina, E., Corno, F.: Environmental control application compliant with COGAIN guidelines. In: COGAIN 2009-Gaze Interaction for Those Who Want It Most, pp. 31–34 (2009)
9. Bates, R.: Multimodal eye-based interaction for zoomed target selection on a standard graphical user interface. In: Proceedings of Interact 99, vol. 2, pp. 7–8 (1999)

10. Zhai, S., Morimoto, C., Ihde, S.: Manual and gaze input cascaded (MAGIC) pointing. In: Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems, pp. 246–253 (1999)
11. Dostal, J., Kristensson, P.O., Quigley, A.: Subtle gaze-dependent techniques for visualising display changes in multi-display environments. In Proceedings of the 2013 international conference on Intelligent user interfaces. ACM, pp. 137–148 (2013, March)
12. Khonglah, J.R., Khosla, A.: A low cost webcam based eye tracker for communicating through the eyes of young children with ASD. In: Next Generation Computing Technologies (NGCT), 2015 1st IEEE International Conference on, pp. 925–928, (2015)
13. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. IEEE, vol. 1, pp. I–I (2001)
14. Cuong, N.H., Hoang, H.T.: Eye-gaze detection with a single webcam based on geometry features extraction. In: Control Automation Robotics & Vision (ICARCV), 2010 11th International Conference on. IEEE, pp. 2507–2512 (2010)
15. Sewell, W., Komogortsev, O.: Real-time eye gaze tracking with an unmodified commodity webcam employing a neural network. In CHI'10 Extended Abstracts on Human Factors in Computing Systems. ACM, pp. 3739–3744 (2010)
16. San Agustin, J., Skovsgaard, H., Mollenbach, E., Barret, M., Tall, M., Hansen, D.W., Hansen, J.P.: Evaluation of a low-cost open-source gaze tracker. In: Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications, pp. 77–80 (2010)
17. Biswas, P., Shree, J.D.: Analysing ocular parameters of users with cerebral palsy for developing gaze controlled interface, EC Opthalmology **8**(5) (2017)
18. Vasanth, J., Jacob, N., Viswanathan, S.: Visual function status in children with cerebral palsy. Optom. Vis. Perform. **2**(3), 251–253 (2014)
19. Sahay, A., Biswas, P.: Webcam Based Eye Gaze Tracking Using a Landmark Detector. In: Proceedings of the 10th Annual ACM India Compute Conference on ZZZ. ACM, pp. 31–37 (2017)
20. Tobii Eye Gaze Tracker: https://www.tobii.com/ (visited on 12/12/2017)
21. Biswas, P., Langdon, P.: A new input system for disabled users involving eye gaze tracker and scanning interface. J. Assist. Technol. **5**(2), 58–66 (2011)