Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2016) 24: 5162 – 5172

doi:10.3906/elk-1503-132

Real-time motorized electrical hospital bed control with eye-gaze tracking

Nesrin AYDIN ATASOY[1,*], **Abdullah ÇAVUŞOĞLU**[2], **Ferhat ATASOY**[1]
[1]Department of Computer Engineering, Faculty of Engineering, Karabük University, Karabük, Turkey
[2]Council of Higher Education, Bilkent, Ankara, Turkey

Received: 17.03.2015 • **Accepted/Published Online:** 03.11.2015 • **Final Version:** 06.12.2016

Abstract: Patients with motor neuron disease and most terminal patients cannot use their hands or arms, and so they need another person for their all needs. However, the mental functions and memories of such patients are generally sound, and they can control their eyes. Using an eye-gaze tracking technique, we have realized a real-time system for such patients. The system controls a motorized electrical hospital bed (EHB) by eye gaze with 4 degrees of freedom, using a low-cost webcam. Contactless systems that require calibration cannot be used for EHB control. The system developed in this work does not require any calibration process and it is contactless. These properties are the most innovative part of the proposed approach. To begin, the system detects the eye region and computes the iris centers. It then tracks the centers and moves a mouse pointer on a screen with the eye gaze. The specific movements of the mouse pointer are evaluated as position changing requests and the completed movements of the mouse pointer change the EHB position electrically. The communication between the computer and the EHB is provided by a relay control card driven by Arduino Mega. The system works under day/artificial lighting conditions successfully with or without eyeglasses. The system was tested with 30 volunteers on the EHB safely and was completed with 90% success (the exceptions being people with slanted eyes).

Key words: Calibration-free eye tracking, electrical hospital bed, eye-based mouse control, real-time eye-gaze tracking

1. Introduction

Motor neuron disease refers to a group of disorders of the motor system. This group includes amyotrophic lateral sclerosis, progressive muscular atrophy, progressive bulbar palsy, and primary lateral sclerosis [1]. These diseases cause weakness of the muscles and myolysis, but mental functions and memory are not affected. People who suffer from them usually live bedridden and need helpers. In addition, because of the loss of muscular movement and speaking ability, some of the patients' communication with their environment and quality of life is significantly reduced.

Technology has been used for improvement in the quality of life and the survival rate of patients for many years; medical engineering applications are one of the subjects that researchers have focused on [2]. Thus, devices controlled by head movements, eye movements, hand movements, and other gestures have been developed. Eye-gaze tracking has been a research topic for many years; it has been used for cases such as human–computer interaction, virtual reality, eye disease diagnosis, human behavior studies, etc. [3]. The term eye gaze refers to the direction of a person's focus of attention and interest [4]. When a user looks at a computer screen, the point of the user's gaze can be estimated via an eye-gaze tracker [3].

---

[*] Correspondence: nesrinaydin@karabuk.edu.tr

Electronic, contact lens, and video-based methods are used for eye-gaze tracking. Patients' skin touch on hardware is used in some systems that can use a word processor with infrared oculography and electrooculography [5,6] to control an electrical wheelchair using a camera-mounted glass [7,8]. These systems limit the movement ability of patients. Video-based systems in which the camera is located remotely are more useful than the methods mentioned above. Remote camera video-based systems do not reduce life quality for the patients. In video-based systems, IR illumination is usually used [9,10]. In studies without IR illumination, the iris and pupil are detected in eye region images provided by a remote video camera by applying a morphological process [11–15]. Eye tracking technologies based on digital image analysis have been developed with rapid technological progress for video cameras and microcomputers. Tracking speed, accuracy, and robustness are important challenges when developing real-time eye tracking systems. Here the most important goal is developing systems independent of head movements [16]. At present, end-user eye tracking systems are expensive. Thus, researchers work on low-cost systems for eye-gaze trackers [17–19].

When comparing eye-gaze EHB control with previous publications [5–8], contactless systems have some application difficulties. In the case of EHB control, contactless systems that require calibration cannot be successful, because when the head partition of an EHB changes its position, the angle between the eyes and the camera will change. As a result, calibration is lost. Hence, among systems that require calibration, only those that are head-mounted can be used for EHB control. However, head-mounted systems reduce the quality of the patient's life.

This study aims to realize a system for controlling EHB using a low-cost webcam. It is not a head-mounted system and does not require a calibration process. Thus, the EHB can be controlled without reducing the patient's life quality, and the patient does not need anyone to control and calibrate the system. Our proposed approach is unique in these aspects.

2. Methods

2.1. Detecting eye region computing and tracking centers of irises

In this study, we have used the Viola–Jones algorithm for eye region detection in real time. The algorithm works in 3 basic steps:

- Creating an "integral image" for fast feature detection with rectangle features called Haar features;
- AdaBoost learning method;
- Haar cascade classifier (HCC) for integrating many features efficiently.

The Viola–Jones object detection algorithm classifies images with simple features. Haar features are the differences of pixel intensity values between adjacent rectangular regions [20]. If the difference is greater than a threshold, the feature is present. For example, Figure 1 shows 2 Haar features [21]. The first feature is the difference in intensity between the eye region and a region across the upper cheeks. It shows that the eye region is darker than the cheeks. The second feature compares the eye region and bridge of the nose [21,22].

An integral image is used for computing rectangular features rapidly. The integral image is computed with Eq. (1). In Eq. (1), AI refers to the integral image, A is the source image, and x, y are the coordinates of each pixel [20].

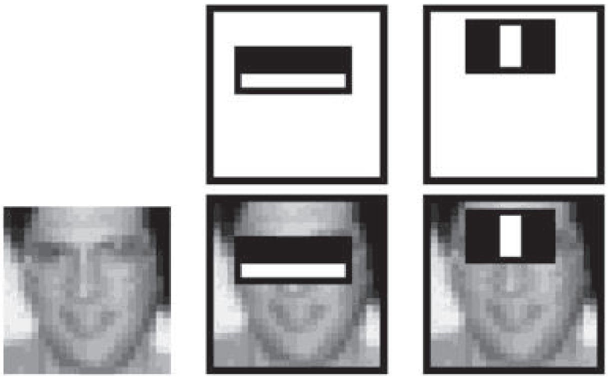
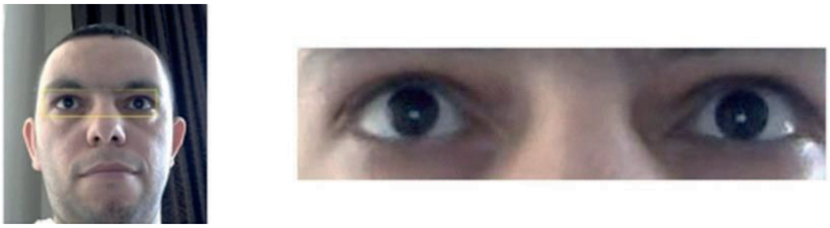


Figure 1. Application of edge and line Haar features on eye region [21].

$$AI\left(x,y\right)=\sum_{x'\leq x,y'\leq y}A\left(x^{'},y^{'}\right). \tag{1}$$

The AdaBoost machine learning method is used for selection of specific Haar features and the setting of threshold levels. AdaBoost combines weak classifiers to build a strong classifier [23]. If the response of any classifier is false, the trained weak classifiers are applied to the image respectively until all the classifiers have passed. Each of the classifiers examines different regions in feature space. If the image regions pass through all filters, it will be classified as a searched object.

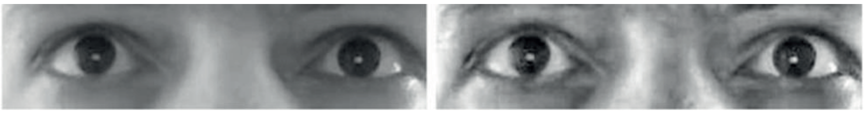
The eye region is detected from the face image captured from the webcam in real time, shown in Figure 2a. Extraction of eye region features is a preprocess for real-time eye-gaze tracking. The face image is cropped depending on the detected eye region as shown in Figure 2b.



(a) (b)

Figure 2. Face image: (a) detected eye region; (b) cropped region.

The cropped image is converted to grayscale as shown in Figure 3a. The cropping and conversion to grayscale processes are performed to reduce computing time.



(a) (b)

Figure 3. Processes on the cropped region: (a) grayscale; (b) after CLAHE.

A variation of histogram equalization is used for making iris boundaries sharp. Histogram equalization is a technique for adjusting color or intensity frequency to enhance contrast. Global histogram equalization is applied depending on the values of all pixels of the image. In the system, we used contrast-limited adaptive histogram equalization (CLAHE), which enhances the local contrast [24]. When CLAHE is applied to a gray image, the difference of contrast between the sclera (the white portion of the eyeball) and the iris is more salient, as shown in Figure 3b.

The center coordinates of the irises of both eyes are then computed with the circular Hough transform method (CHTM) on the image, shown in Figure 3b. The Hough transform method is an effective shape-based method for detecting geometric shapes such as lines, circles, and ellipses [25,26]. The method has been used for detecting pupils and irises in recent studies [26–28]. The CHTM is applied, circle lines are drawn on the boundaries of the irises, and the centers are marked with an asterisk, as shown in Figure 4a. Computed centers of the irises of 2 eyes are shown in Figure 4b. Circular objects are detected with Eq. (2):

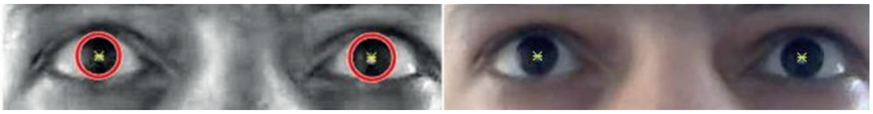

(a) (b)

Figure 4. Irises and centers: (a) detected and computed; (b) on real-time video image.

$$r^2 = (x-a)^2 + (y-b)^2. \tag{2}$$

CHTM requires voting for $a\,b\,r$ in the parameter space. In Eq. (2), a and b represent the coordinates of the center, and r is the radius of the circle. The parametric representation of the circle equation is shown in Eq. (3).

$$\begin{aligned} x &= a + r(\sin\theta), \\ y &= b + r(\cos\theta) \end{aligned} . \tag{3}$$

If the angle θ sweeps through the full 360° range, the points (x, y) trace the perimeter of a circle [27]. The location $(a,\ b,\ r)$ with the maximum value of Hough space is chosen for the parameters for the strongest circular boundary [29]. Figure 4a shows the detected irises and their centers. After applying CLAHE, the CHTM results are more robust in different lightning conditions and with different colored eyes.

For each frame, the processes include detection of eye region, cropping of the eye region, conversion to grayscale, applying CLAHE to the grayscale image, and using CHTM to compute centers and boundaries of irises; the processes are computationally expensive. Thus, after computing center points, point tracking is an effective way to track the real-time eye gaze.

Subsequently, the center points of the irises are tracked by the Kanade–Lucas–Tomasi (KLT) tracking algorithm once detected. If frame brightness constancy constraint is satisfied and the motion of image is sufficiently small, the KLT algorithm can be used [30–32]. Thus, it was chosen in our study. The KLT method defines an alignment template T(x) and the tracker tries to find tracked points in T(x) according to the previous state, called initial points. The tracker will search different points to find alignment if the image and the template do not converge. The search process will stop when the best alignment is found. Thereafter, the new state will constitute the initial state for the next state [33].

Figure 4b shows that the tracked center points of the irises using the KLT algorithm and the points are matched in the video frame. The computed center coordinates of the irises are assigned as reference coordinates and the mouse pointer position on screen moves to the direction in which the eyes are looking. The direction is determined by computing the difference between the current coordinates of the centers and the reference coordinates. The reference coordinates are updated after each eye detection process. Thus, the calibration process originating from personal differences, distance, etc. is not a requirement.

3. Controlling the mouse pointer

The algorithm for controlling the mouse pointer is presented. The movements are calculated as the difference between the last location of the centers and the reference coordinates. In the algorithm, k is the movement constant. Usually 2 centers are tracked, but the algorithm is designed to be dynamic in case one of the eyes is blind.

The sensitivity value of the vertical direction is half that of the horizontal direction, because movement ability in the vertical direction is smaller than in the horizontal direction. In computer graphics, coordinates of the y axis increase from top to bottom, so coordinate conversion is a necessity in the algorithm at the fourth and fifth steps.

4. The developed system

The system was developed on a machine with an Intel Core i7-2670HQ CPU, 2.20 GHz, 8 GB RAM, 64-bit operating system. The application program was developed in MATLAB R2014b and the interface was developed with C# Visual Studio 2013. The source for video images is a Logitech C310 720p HD webcam. Arduino 2560 codes are compiled in the Arduino compiler, version 1.6. A block diagram of the system is given in Figure 5.

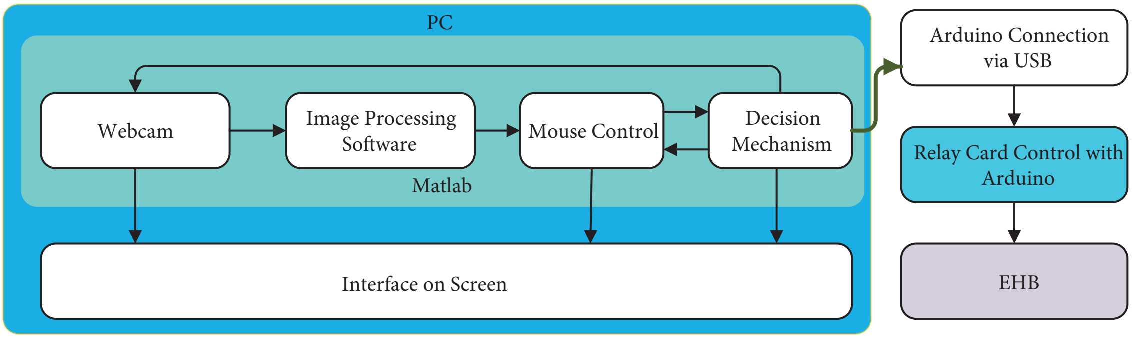


Figure 5. The block diagram of the developed system.

The webcam and the monitor are installed on the motorized EHB as shown in Figure 6. All installed parts move together. The fixation provides an approximately constant angle for images from the webcam. The distances of the camera and the monitor to the patient are approximately 45 cm and 50 cm, respectively.

A flowchart for the EHB control system is given in Figure 7. The system operates in 2 modes: Mode 0 and Mode 1 for detection and tracking, respectively. The software starts in detection mode and locates the mouse pointer to the center of the screen. The patient then looks at the mouse pointer. At the same time, video images are taken from the webcam, and the centers of the irises are computed on the cropped eye region. The center coordinates are then assigned as reference points and tracking mode is activated. Furthermore, the patient is informed when tracking mode is activated. The system works in tracking mode until the EHB control command is completed or the duration of Timer 2 is expired. If the eyelids are closed or the points go out of the line of sight of the webcam, the points will be lost. The detection mode is then reactivated. If centers of the irises cannot be detected 2000 times or the application is closed, the system will stop. If the counter causes the system to stop during the first use, it is not available for the patient. The algorithm of the developed system works without a counter in an infinite loop for a patient whose iris centers are detected.

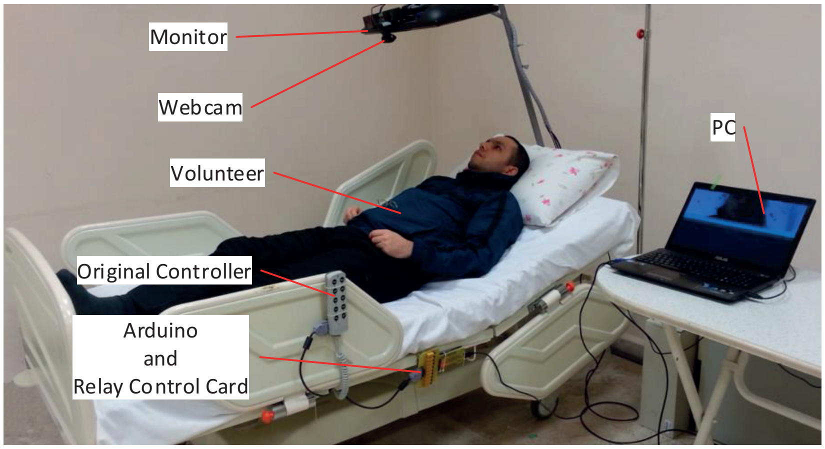


Figure 6. The developed system.

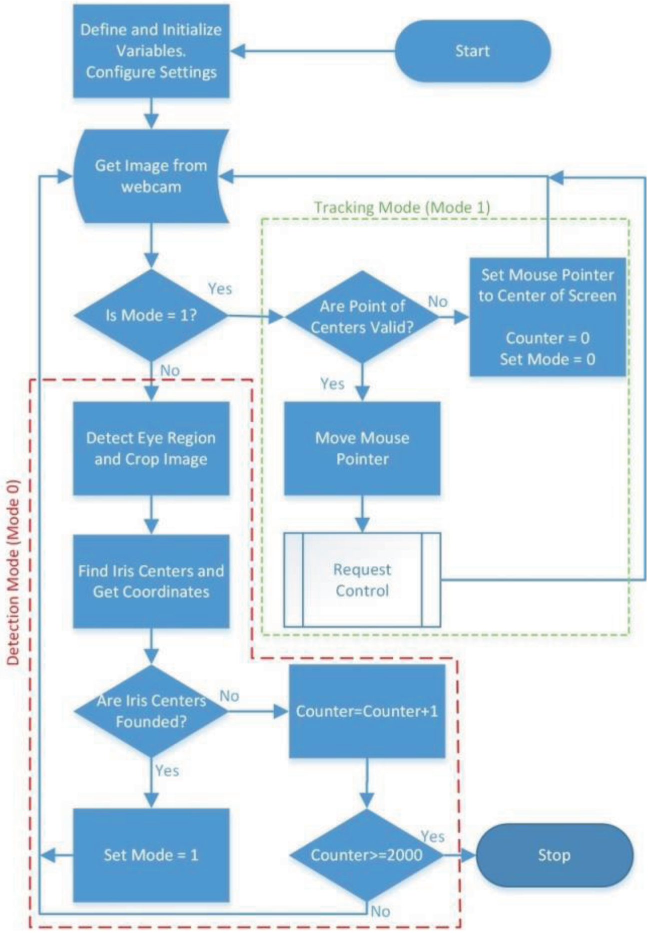


Figure 7. Flowchart of the developed software.

The system is easy to use. In tracking mode, the “Request Control” subroutine seen in Figure 7 and “Decision Mechanism” in Figure 5 are synchronized and integrated with the mouse pointer to realize the EHB control. A command set corresponding to eye movements has been set up and matched with EHB movements. This command set is shown in Table 1. If Table 1 is examined carefully, all of the commands start with “top”. Our proposed approach has been based on healthy people who can use their arms and hands. These people can take the controller and press the button. The taking and pressing processes are specific activities for changing EHB positions. Thus, the patient who controls the EHB by eye-gaze tracking must perform specific eye movements. The interface and integrated control software change the EHB positions with specific combinations of mouse movements, because ordinary eye movements can cause unintended changes in position. Generally, people look in opposite or downward directions, so looking up can be evaluated as a specific eye movement to start. When the patient looks at the center of the screen, the centers are detected. The eyes are then moved up and the mouse pointer goes to the top. This command set can correspond to different actions by modifying the interface and integrated software if required.

Table 1. Eye movements-based command set.

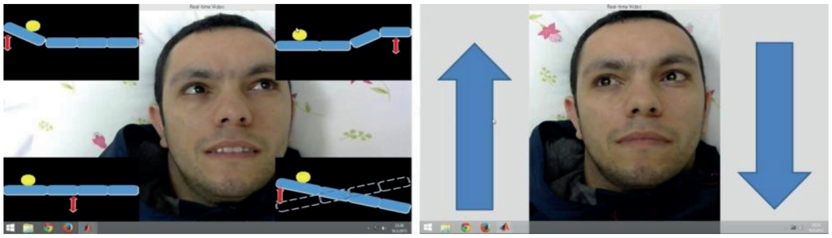
Function	Direction	Command steps by moving the mouse pointer
Back-rest lifting	Up	Top - Wait on top - Top Left - Wait on Top Left - Left
	Down	Top - Wait on top - Top Left - Wait on Top Left - Right
Knee-rest lifting	Up	Top - Wait on top - Top Right - Wait on Top Right - Left
	Down	Top - Wait on top - Top Right - Wait on Top Right - Right
Height adjustable	Up	Top - Wait on top - Bottom Left - Wait on Bottom Left - Left
	Down	Top - Wait on top - Bottom Left - Wait on Bottom Left - Right
Trendelenburg	Up	Top - Wait on top - Bottom Right - Wait on Bottom Right - Left
Anti-Trendelenburg	Down	Top - Wait on top - Bottom Right - Wait on Bottom Right – Right

There are 2 timers in the request control subroutine. Timer 1 is used for detection of the mouse pointer waiting long enough on the correct coordinates before the required movements. Timer 2 is used for returning to the main menu if there is not any significant movement of the mouse pointer.

In tracking mode, initially, the mouse pointer is located on the center of screen. If the mouse pointer goes to the top region of the screen, system wait for a defined duration (i.e. Timer 1) and then it is evaluated as a bed-position changing request. The control menu given in Figure 8a is then shown and the mouse pointer is located at the center of the screen again. Timer 2 starts when the control menu is on the screen. If the patient moves the mouse pointer to the movement symbols on the edges and waits for the duration of Timer 1, the mouse pointer is moved to the center of the screen again and the direction selection screen given in Figure 8b is shown. The patient can select the direction of movement by moving the mouse pointer to the left or right side of the screen. Position change starts when the mouse pointer is on the arrow symbols and continues until the eyelids close. Tracking mode is disabled and detection mode is enabled after each command is completed. Thus, the developed system is not influenced even if the angle between the webcam and the eyes is changed. If the patient does not move the mouse pointer to the symbols until the defined duration of Timer 2, the main screen returns.

5. Experimental results

In the system, movement in 8 directions using 4 electrical motors on an EHB control system is attained with an additional graphical user interface. The system also employs image processing software, a relay control card, and a patient interaction interface.



(a) (b)

Figure 8. Control menu screens: (a) movement type selection screen; (b) direction of the movement screen.

The system was tested with 30 people. The age range of volunteers was between 18 and 70. The system works well under day/artificial light conditions. Eye detection performance is 90% with a 50-cm webcam–volunteer distance. Sex, eye color, and nationality do not have any influence on the detection results, except for a slanted eye shape. The group of volunteers with slanted eyes corresponds to 10% of the total. The training time increases with increasing age.

Captured video time is 2.22 ms. Image processing time is 1750 ms, and total time for each loop of detection mode seen in Figure 7 is 1752.22 ms. Image processing time is 31.30 ms, and total time for each loop of tracking mode seen in Figure 7 is 33.52 ms. The average time to reach the direction of the movement screen as seen in Figure 8b is 8 s. Position changing time depends on the EHB motor speed and the patient's request.

Volunteers were asked to score the developed system and the results of the test assessment questionnaires (adopted from [34]) are shown in Figure 9. "Strongly disagree" refers to 1 point and "strongly agree" refers to 5 points. As expected, the volunteers found that the system was not complex, easy to use, stable, and feasible. The distance between the screen and the volunteers was found to be suitable (Appendix).

The important parameter of iris detection depending on distance is the radius of the iris, because the CHTM takes radius range as a parameter. Figure 10 shows the iris radius value of the same person in different camera distance conditions at 1280 × 720 resolution. If the distance is more than 50 cm, the CHTM becomes unstable.

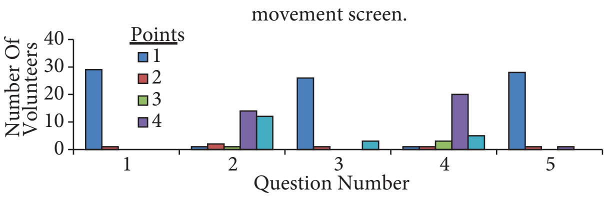


Figure 9. The results of the test assessment questionnaires.

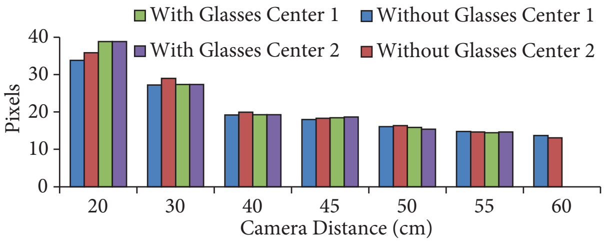


Figure 10. Experimental results according to distance.

6. Conclusions and future work

Eye tracking methods have been compared in some references [35–37]. Recent studies have focused on video camera-based methods, because other methods are intrusive. An ideal eye tracker should be accurate, reliable, robust, and nonintrusive. It should also allow for free head motion, not require calibration, and have real-time response. Table 2 shows prominent features of general video-based eye tracking techniques. In the visible spectrum, success is dependent on ambient light, so results in the dark are not good. Infrared illuminated systems can solve this problem, but more than one reflection is detected if the patient uses glasses. Systems that use corneal reflection require a calibration process and complex calculations. Head-mounted systems are low-cost and provide mobility and good performance in real-time applications. Generally, remote eye trackers use a light source (infrared or ambient) and a camera located opposite of the user; they need calibration for every session.

Table 2. Prominent features of general video based eye tracking techniques.

Spectrum	Technique	Detection algorithm	Head-mounted	Remote
Visible	Iris/pupil contour	Feature based	Low cost	Expensive
Infrared	Corneal reflection	Model based	Uncomfortable	Complex algorithms
		Hybrid		Calibration required

The aim of this study has been to develop a system to improve the quality of life for patients. Well-known techniques of image processing and control hardware were used for developed this unique system. The system is unique because it uses simple and rather cheap hardware when compared to sophisticated professional devices. In addition, it is calibration-free and contactless. Furthermore, it has proven effectiveness. The user's ability to work certain controls on a screen can be further developed and the system can be further enhanced with the addition of more instruments to be controlled by the system. The user's ability to navigate through different screens can actually provide a limitless number of controls that can be embedded in the system. Since the system is complete and well defined, these extras can easily be incorporated.

Acknowledgment

This study was supported by Yıldırım Beyazıt University as a preliminary research project (project number 592).

References

[1] Ross MA. Acquired motor neuron disorders. Neurol Clin 1997; 15: 481-500.

[2] Akıncı G, Polat E, Koçak OM. A video-based eye pupil detection system for diagnosing bipolar disorder. Turk J Elec Eng Comp Sci 2013; 21: 2367-2377.

[3] Zhu Z, Qiang J. Novel eye gaze tracking techniques. IEEE T Bio-Med Eng 2007; 54: 2246-2260.

[4] Lu HC, Fang GL, Wang C, Chen YW. A novel method for gaze tracking by local pattern model and support vector regressor. Signal Process 2010; 90: 1290-1299.

[5] Yamada M, Fukuda T. Eye word processor (EWP) and peripheral controller for the ALS patient. IEE Proc-A 1987; 134: 328-330.

[6] Yıldız M. A new coding technique for EOG based writing systems. In: 19th Signal Processing and Communications Applications Conference (SIU); 20–22 April 2011; Antalya, Turkey. Piscataway, NJ, USA: IEEE. pp. 90-93.

[7] Arai K, Mardinyato R. Eye based electric wheelchair control system. International Journal of Advanced Computer Science and Applications 2011; 2: 98-105.

[8] Gautam G, Sumanth G, Karthikeyan KC, Sundar S, Venkataraman D. Eye movement based electronic wheel chair for physically challenged persons. International Journal of Scientific & Technology Research 2014; 3: 206-212.

[9] Morimotoa CH, Koons D, Amir A, Flickner M. Pupil detection and tracking using multiple light sources. Image Vision Comput 2000, 18: 331-335.

[10] Beymer D, Flickner M. Eye gaze tracking using an active stereo head. In: 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition; 18–20 June 2003; Madison, WI, USA. Los Alamitos, CA, USA: IEEE. pp. 451-458.

[11] Moravcikr T, Bubenikova E, Muzika L. Detection of determined eye features in digital image. Annals of Faculty Engineering Hunedoara 2011; 9: 155-160.

[12] Ghoshal D, Das A. A study on human eye detection in a color image based on Harris corner detector and Sobel edge operator. International Journal of Electron Communication and Computer Technology 2012; 2: 130-133.

[13] Dhiman A, Kumar A, Arora M. Design of a real time system for acquiring and analyzing iris images in MATLAB. International Journal of Electron & Communication Technology 2011; 2: 106-110.

[14] Kulkarni S, Gala S. A simple algorithm for eye detection and cursor control. International Journal of Computer Technology & Applications 2013; 4: 880-882.

[15] Mohammed AA, Anwer SA. Efficient eye blink detection method for disabled-helping domain. International Journal of Advanced Computer Science and Applications 2014; 5: 202-206.

[16] Boulay G. Eye pose tracking & gaze estimation. MSc, Chinese Academy of Sciences, Beijing, China, 2008.

[17] Aydın Atasoy N, Çavuşoğlu A. Detection of eye motion direction in real time video image. In: 22nd Signal Processing and Communications Applications Conference, 23–25 April 2014; Trabzon, Turkey. Piscataway, NJ, USA: IEEE. pp. 1487-1490.

[18] Barrett M, Jensen HTSH, San Agustin J. Performance evaluation of a low-cost gaze tracker for eye typing. In: Proceedings of the 5th Conference on Communication by Gaze Interaction; 26 May 2009; Lynby, Denmark. Lynby: Technical University of Denmark. pp. 1-5.

[19] Lemahieu W, Wyns B. Low cost eye tracking for human-machine interfacing. Journal of Eye Tracking, Visual Cognition and Emotion 2010, 1: 1-12.

[20] Wilson PI, Fernandez J. Facial feature detection using Haar classifiers. Journal of Computing Sciences in Colleges 2006; 21: 127-133.

[21] Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. In: Proceedings of CVPR IEEE 2001. Silver Springs, MD, USA: IEEE. pp. 511-518.

[22] Viola P, Jones M. Robust real time face detection. Int J Comput Vision 2004; 54: 137-154.

[23] Freund Y, Schapire RE. Experiments with a new boosting algorithm. In: Machine Learning: Proceedings of the Thirteenth International Conference; 3–6 July 1996; Bari, Italy. San Francisco, CA, USA: Morgan Kaufmann Publishers. pp. 148-156.

[24] Yu CY, Lin HY, Ouyang YC, Yu TW. Modulated AIHT image contrast enhancement algorithm based on contrast-limited adaptive histogram equalization. Applied Mathematics & Information Sciences 2013; 7: 449-454.

[25] Chen X, Lu L, Gao Y. A new concentric circle detection method based on Hough transform. In: 7th International Conference on Computer Science & Education; 14–17 July 2012; Melbourne, Victoria, Australia. Piscataway, NJ, USA: IEEE. pp. 753-758.

[26] Schwarz L, Gamba HR, Pacheco FC, Ramos RB, Sovierzoski MA. Pupil and iris detection in dynamic pupillometry using the OpenCV library. In: 2012 5th International Congress on Image and Signal Processing; 16–18 October 2012; Chongqing, Sichuan, China. Piscataway, NJ, USA: IEEE. pp. 211-215.

[27] Nkole IU, Sulong GB. An enhanced iris segmentation algorithm using circle Hough transform. In: 4th International Conference on Informatics and Technology; 10–11 November 2011; Kuala Lumpur, Malaysia.

[28] Lee Y, Michaels RJ, Phillips PJ. Improvements in video-based automated system for iris recognition (VASIR). In: Workshop on Motion and Video Computing; 8–9 December 2009; Snowbird, UT, USA. Piscataway, NJ, USA: IEEE. pp. 1-8.

[29] Hilal A, Daya B, Beauseroy P. Hough transform and active contour for enhanced iris segmentation. Int J Comput Sci Issues 2012; 9: 1-10.

[30] Sinha SN, Frahm JM, Pollefeys M, Genc Y. GPU-based video feature tracking and matching. In: Workshop on Edge Computing Using New Commodity Architectures; 2006.

[31] Tomasi C, Kanade T. Detection and Tracking of Point Features. Technical Report CMU-CS-91-132. Pittsburgh, PA, USA: Carnegie Mellon University, 1991.

[32] Fassold H, Rosner J. Realtime KLT Feature point tracking for high definition video. In: International Workshop on Computer Graphics, Computer Vision and Mathematics; 2009.

[33] Bagherpour P, Cheraghi SA, Mokji MbM. Upper body tracking using KLT and Kalman filter. Procedia Computer Science 2012; 13: 185-191.

[34] Brooke J. SUS: a 'quick and dirty' usability scale. In: Jordan PW, Thomas B, McClelland IL, Weerdmeester B, editors. Usability Evaluation In Industry. Bristol, UK: Taylor & Francis, 1996. pp. 189-194.

[35] Lupu RG, Ungureanu F. A survey of eye tracking methods and applications. Polytechnic Institute of Science Bulletin 2013, 3: 71-86.

[36] Tien T, Pucher PH, Sodergren MH, Sriskandarajah K, Yang GZ, Darzi A. Eye tracking for skills assessment and training: a systematic review. J Surg Res 2014; 191: 169-178.

[37] Morimoto CH, Mimica MRM. Eye gaze tracking techniques for interactive applications. Comput Vis Image Und 2005; 98: 4-24.

Appendix

Questionnaire (1-> Strongly Disagree to 5-> Strongly Agree)	1	2	3	4	5
1. The system was very complex					
2. The system was easy to use					
3. The working of the system was unstable					
4. I thought most people can learn to use the system quickly					
5. I felt uncomfortable because of the distance between the screen and me					