LONG PAPER

Eye-blink detection system for human–computer interaction

Aleksandra Królak · Paweł Strumiłło

Abstract A vision-based human–computer interface is presented in the paper. The interface detects voluntary eye-blinks and interprets them as control commands. The employed image processing methods include Haar-like features for automatic face detection, and template matching based eye tracking and eye-blink detection. Interface performance was tested by 49 users (of which 12 were with physical disabilities). Test results indicate interface usefulness in offering an alternative mean of communication with computers. The users entered English and Polish text (with average time of less than 12s per character) and were able to browse the Internet. The interface is based on a notebook equipped with a typical web camera and requires no extra light sources. The interface application is available on-line as open-source software.

Keywords Human–computer interface · Eye-blink detection · Face detection

1 Introduction

Human–Computer Interface (HCI) can be described as the point of communication between the human user and a computer. Commonly used input devices include the following: keyboard, computer mouse, trackball, touchpad and a touch-screen. All these devices require manual control and cannot be used by persons impaired in movement capacity. Therefore, there is a need for developing alternative methods of communication between human and computer that would be suitable for the persons with motor impairments and would give them the opportunity to become a part of the Information Society. In recent years, the development of alternative human–computer interfaces is attracting attention of researchers all over the world. Alternative means of interacting for persons who cannot speak or use their limbs (cases of hemiparesis, ALS, quadriplegia) are their only way of communication with the world and to obtain access to education or entertainment.

A user friendly human–computer interface for severely movement impaired persons should fulfill several conditions: first of all, it should be non-contact and avoid specialized equipment, it should feature real-time performance, and it should run on a consumer-grade computer.

In this paper, a vision-based system for detection of voluntary eye-blinks is presented, together with its implementation as a Human–Computer Interface for people with disabilities. The system, capable of processing a sequence of face images of small resolution (320×240 pixels) with the speed of approximately 30 fps, is built from off-the-shelf components: a consumer-grade PC or a laptop and a medium quality webcam. The proposed algorithm allows for eye-blink detection, estimation of the eye-blink duration and interpretation of a sequence of blinks in real time to control a non-intrusive human–computer interface. The detected eye-blinks are classified as short blinks (shorter than 200 ms) or long blinks (longer than 200 ms). Separate short eye-blinks are assumed to be spontaneous and are not included in the designed eye-blink code.

Section 2 of the paper includes an overview of earlier studies on the interfaces for motor impaired persons. The

A. Królak (✉) · P. Strumiłło
Institute of Electronics, Technical University of Lodz,
Wolczanska 211/215, 90-924 Lodz, Poland
e-mail: akrolak@p.lodz.pl

P. Strumiłło
e-mail: pstrumil@p.lodz.pl

proposed eye-blink detection algorithm is described in Sect. 3. Section 4 presents the eye-blink controlled human–computer interface based on the proposed algorithm. Research results are discussed in Sect. 5 and the conclusion is given in Sect. 6.

2 Previous work

For severely paralyzed persons who retain control of the extraocular muscles, two main groups of human–computer interfaces are most suitable: brain–computer interfaces (BCI) and systems controlled by gaze [1] or eye-blinks.

A brain–computer interface is a system that allows controlling computer applications by measuring and interpreting electrical brain activity. No muscle movements are required. Such interfaces enable to operate virtual keyboards [2], manage environmental control systems, use text editors, web browsers or make physical movements [3]. Brain–computer interfaces hold great promise for people with severe physical impairments; however, their main drawbacks are intrusiveness and need for special EEG recording hardware.

Gaze controlled and eye-blink-controlled user interfaces belong to the second group of systems suitable for the people who cannot speak or use their hands to communicate. Most of the existing methods for gaze communication are intrusive or use specialized hardware, such as infrared (IR) illumination devices [4] or electrooculographs (EOG) [5]. Such systems use two kinds of input signals: scanpath (line of gaze determined by fixations of the eyes) or eye-blinks. The eye-blink-controlled systems distinguish between voluntary and involuntary blinks and interpret single voluntary blinks or their sequences. Specific mouth moves can also be included as an additional modality. Particular eye-blink patterns have the specific keyboard or mouse commands assigned, e.g., a single long blink is associated with the TAB action, while a double short blink is a mouse click [29]. Such strategies can be used as controls for simple games or for operating programs for spelling words.

The vision-based eye-blink detection methods can be classified into two groups, active and passive. Active eye-blink detection techniques require special illumination to take advantage of the retro-reflective property of the eye. The light falling on the eye is reflected from the retina. The reflected beam is very narrow, since it comes through the pupil and it points directly toward the source of the light. When the light source is located on the focal axis of the camera or very close to it, the reflected beam is visible on the recorded image as the bright pupil effect (Fig. 1). The bright pupil phenomenon can be observed in the flash photography as the red eye effect.
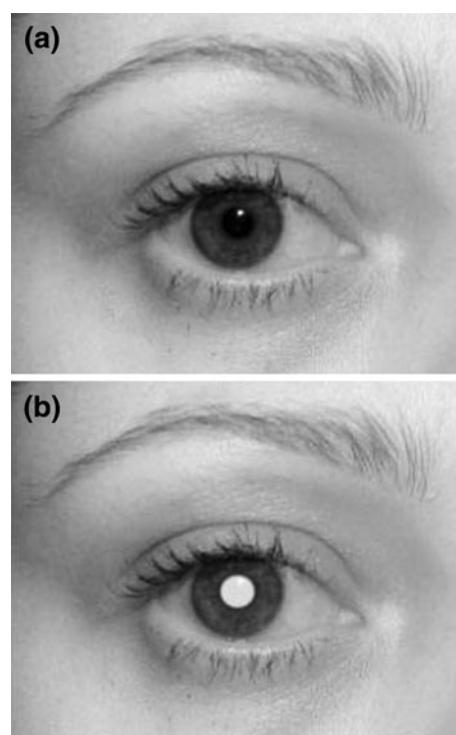

Fig. 1 Eye image in IR spectrum: **a** eye in natural lighting conditions, **b** bright pupil effect after IR illumination of the eye

An example of the gaze-communication device taking advantage of IR illumination is Visionboard system [4]. The infrared diodes located in the corners of the monitor allow for the detection and tracking of the user's eyes employing the bright pupil effect. The system replaces the mouse and the keyboard of a standard computer and provides access to many applications, such as writing messages, drawing, remote control, Internet browsers or electronic mail. However, the majority of the users were not fully satisfied with this solution and suggested improvements.

A more efficient system was described in [9]. It uses two webcams—one for pupil tracking and second for estimating head position relative to the screen. Infrared markers placed on the monitor enable accurate gaze tracking. The developed system can replace the computer mouse or keyboard for persons with motor impairments.

The active approach to eye and eye-blink detection gives very accurate results, and the method is robust [8]. The advantages of the IR-based eye-controlled human–computer interfaces are counterbalanced by high end-user costs due to specialized hardware. They are also ineffective in outdoor environment, because of the impact of direct sunlight on the IR illumination. Another concern is about the safety of using such systems for a long time, since the prolonged exposure of the eyeball to the IR lighting may cause damage to the retina [32]. Therefore, the best

solution is the development of the gaze-controlled or eye-blink controlled passive vision-based systems.

Passive eye-blink detection methods do not use additional light sources. The blinks are detected from the sequence of images within the visible spectrum at natural illumination conditions. Most eye-blink detection techniques are in fact methods detecting eye regions in images. Many approaches are used for this purpose, such as template matching [10], skin color models [11], projection [12], directional Circle Hough Transform [13], multiple Gabor response waves [14] or eye detection using Haar-like features [15, 16].

The simplest way of detecting blinks in the image sequence is to subtract the successive frames from each other and to create a difference image (Fig. 2).

Nevertheless, this method requires keeping still the position of the head because, head movements will introduce false detections. This type of blink detection is used mainly as the initialization step in eye tracking systems for eye localization.

Template matching based methods use an eye image model that is compared with the segments of the input image. The matching part of the input image is determined by correlating the template against the current image frame. The main drawback of this approach is that it is not able to deal with scaling, rotation or illumination changes. For this reason, the idea of deformable template has been introduced [17]. In this approach, the parametric model of eye image is defined together with its energy function. In an iterative process of energy minimization, the model image rotates and deforms in order to fit the eye image. A recently proposed approach to detecting objects in images is using Haar-like features in a cascade of boosted tree classifiers [15].
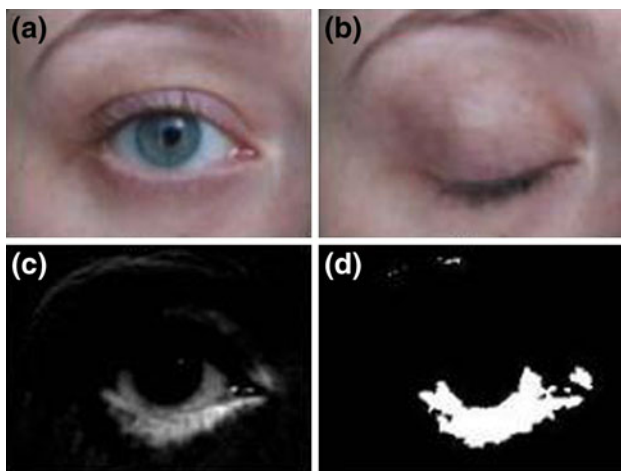

Fig. 2 Eye-blink detection by means of difference images: **a** image of opened eye, **b** image of closed eye, **c** difference image, **d** thresholded difference image

An example of a passive eye detection system is the solution for eye tracking developed by Applied Science Laboratories [18]. It requires wearing head-mounted cameras to look at the eye movements. Such devices enable easier and more reliable performance of the system since the face in such conditions is always in the same location in the video image. However, wearing a headgear, that is not suited for all users, can be uncomfortable, especially for children.

An example of a non-intrusive vision system is presented in [19]. It uses a USB camera to track the eye movements (up, down, left and right) based on the symmetry between the left and the right eye. The system was used to play a specially prepared computer game and for navigating a Web browser (scrolling and simulating the use of TAB and Enter keys).

3 Eye-blink detection system

Vision-based eye-blink monitoring systems have many possible applications, like fatigue monitoring, human–computer interfacing and lie detection. No matter what the purpose of the system is, the developed algorithm must be reliable, stable and work in real time in varying lighting conditions.

The proposed vision-based system for voluntary eye-blink detection is built from off-the-shelf components: a consumer-grade PC or a laptop and a medium quality webcam. Face images of small resolution (320×240 pixels) are processed with the speed of approximately 28 fps. The eye-blink detection algorithm consists of four major steps (Fig. 3): (1) face detection, (2) eye-region extraction, (3) eye-blink detection and (4) eye-blink classification. Face detection is implemented by means of Haar-like features and a cascade of boosted tree classifiers. Eye localization is based on certain geometrical dependencies known for human face. Eye-blink detection is performed using the template matching technique. All the steps of the algorithm are described in more details in Sects. 3.1–3.4. The algorithm allows eye-blink detection, estimation of eye-blink duration and, on this basis, classification of the eye-blinks as spontaneous or voluntary.
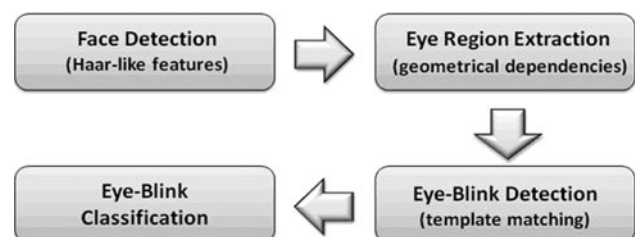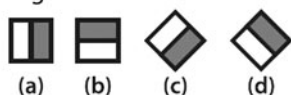

Fig. 3 Scheme of the proposed algorithm for eye-blink detection
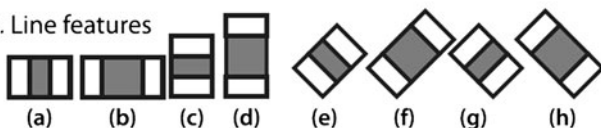
3.1 Face detection

Face detection is a very important part of the developed eye-blink detection algorithm. Due to the fact that face localization are computationally expensive and therefore time consuming, this procedure is run only during the initialization of the system and in cases when the tracked face is "lost". Thus, for the system working in real time, the chosen method should work possibly fast (less than 30ms per single image for speed of 30 fps). On the other hand, the accuracy of the selected approach is also important. The trade-off must be found between the high detection rate, misdetection and error rate. The chosen method should perform robustly in varying lighting conditions, for different facial expression, head pose, partial occlusion of the face, presence of glasses, facial hair and various hair styles. Numerous solutions have been proposed for face detection. They can be divided into: (1) knowledge-based methods employing simple rules to describe the properties of the face symmetry and the geometrical relationships between face features [28], (2) feature-based methods based on the detection of mouth, eyes, nose or skin color [20, 21], (3) template matching methods based on computing the correlation between the input image and stored patterns of the face [22] and (4) model-based methods, where algorithms are trained on models using neural networks [23], Support Vector Machines (SVM) [24] or Hidden Markov Models (HMM) [25]. In the developed algorithm, the method derived from the template matching group developed by Viola and Jones [6], modified by Leinchart and Maydt [26] and implemented according to [15] was used.

The Haar-like features are computed by convolving the image with templates (Fig. 4) of different size and orientation. These feature prototypes can be grouped into 3 categories: edge, line and centre-surround masks. Each template is composed of two or three black and white rectangles.
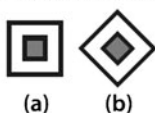

Fig. 4 *Rectangular masks* used for object detection [26]

The feature value for the given mask is calculated as the weighted sum of the intensity of the pixels covered by black rectangle and the sum of pixel intensities covered by the whole mask.

Not all calculated features are necessary to correctly detect faces in the image. An effective classifier can be built using only a part of the features with smallest error rates. In order to find these features, boosting algorithm was used, namely Gentle Adaptive Boosting (Gentle AdaBoost) [27]. The boosting process is repeated several times to build a cascade of classifiers. Each stage of the cascade is a weighted sum of weak classifiers, where the complexity of the stages increase with the number of the stage. For fast feature computation, the new image representation was introduced in [6]—integral image.

The method was tested on a set of 150 face images, and accuracy of 94% was achieved. The speed of the algorithm was tested on face images of different resolution using Intel Core2 Quad CPU at 2.4 GHz processor. The results are presented in Table 1. An example image with faces detected is shown in Fig. 5.

Table 1 Face detection times using cascade of classifiers and Haar-like templates

Image size	Single face (ms)	4 faces (ms)
3,264 × 2448	11,860	11,937
2,448 × 1,836	6,531	6,547
1,632 × 1,224	2,953	3,000
1,280 × 960	1,828	1,875
640 × 480	484	516
320 × 240	156	157

Fig. 5 Result of face detection using Haar-like masks and cascade of boosted tree classifiers [photo taken by Adam Kozłowski, Medical Electronics Division, 2006]

3.2 Eye-region extraction

The next step of the algorithm is eye-region localization in an image. The position of the eyes in the face image is found on the basis of certain geometrical dependencies known for the human face. The traditional rules of proportion show the face divided into six equal squares, two by three [7]. According to these rules, the eyes are located about 0.4 of the way from the top of the head to the eyes (Fig. 6). The image of the extracted eye region is further preprocessed for performing eye-blink detection.

The located eye region is extracted from the face image and used as a template for further eye tracking by means of template matching. The extraction of the eye region is performed only at the initialization of the system and in cases when the face detection procedure is repeated.

3.3 Eye-blink detection

The detected eyes are tracked using a normalized cross-scorrelation method (1). The template image of the user's eyes is automatically acquired during the initialization of the system.

$$R(x', y') = \frac{\sum_{x',y'}[T'(x',y')I(x+x', y+y')]}{\sqrt{\sum_{x',y'} T(x',y')^2 \sum_{x',y'} I(x+x', y+y')^2}} \quad (1)$$

where R correlation coefficient, T template image, I original image, x, y pixel coordinates. The correlation coefficient is a measure of the resemblance of the current eye
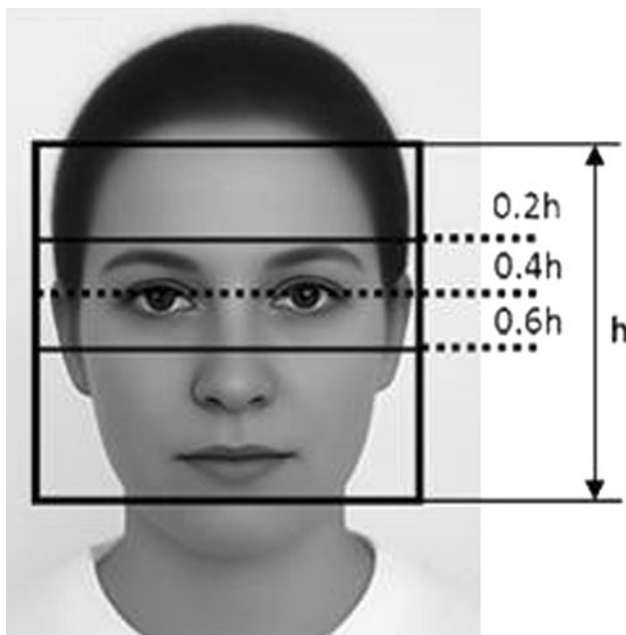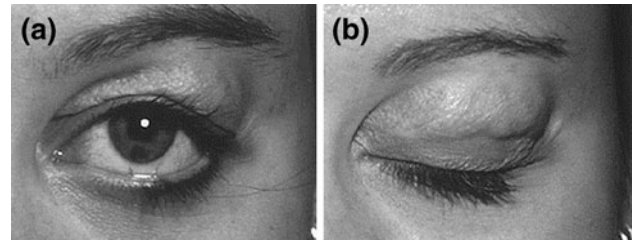

Fig. 7 Example eye images used as templates

image to the saved template of the opened eye (Fig. 7). Therefore, it can be regarded as the measure of the openness of the eye. An example plot of the change of the correlation coefficient value in time is presented in Fig. 9.

3.4 Eye-blink classification

The change of the correlation coefficient in time is analyzed in order to detect the voluntary eye-blinks of duration larger than 250 ms. If the value of the coefficient is lower than the predefined threshold value TL for any two consecutive frames, the on-set of the eye-blink is detected. The off-set of the eye-blink is found if the value of the correlation coefficient is greater than the threshold value TH (Fig. 8). The values of the thresholds TL and TH were determined experimentally. If the duration of a detected eye-blink is longer than 250 ms and shorter than 2 s, then such blink is regarded as a voluntary "control" one.

3.5 Performance of the system

The developed system for eye-blink detection and monitoring was tested using IntelCoreQuad CPU at 2.4 GHz processor on the sequences from the USB Logitech QuickCam 9000 Pro webcam. The size of the input image sequence was equal to 320×240 pixels. Testing of the system took place in a room illuminated by 3 fluorescent lamps and daylight from one window. The person was sitting in front of the monitor, on which the instructions for blinking were displayed. Each person was asked to blink 40 times (20 long blinks and 20 short blinks, alternately). The USB camera was fixed at the top of the monitor, about 70 cm away from the person's face. The detection of the eye-blinks was done in real time with the average speed of 29 fps. Two kinds of errors were identified: false detection (the system detected an eye-blink when it was not present) and missed blinks (a present eye-blink that was not detected by the system). The possible distribution of the eye-blink detector output is presented in graphical form in Fig. 9.

The correctly detected eye-blinks are denoted as True Positives (TP), false detections are denoted as False Positives (FP), and missed eye-blinks are denoted as False

Fig. 6 Rules of human face proportions

Fig. 8 Change of the correlation coefficient value in time during an eye-blink
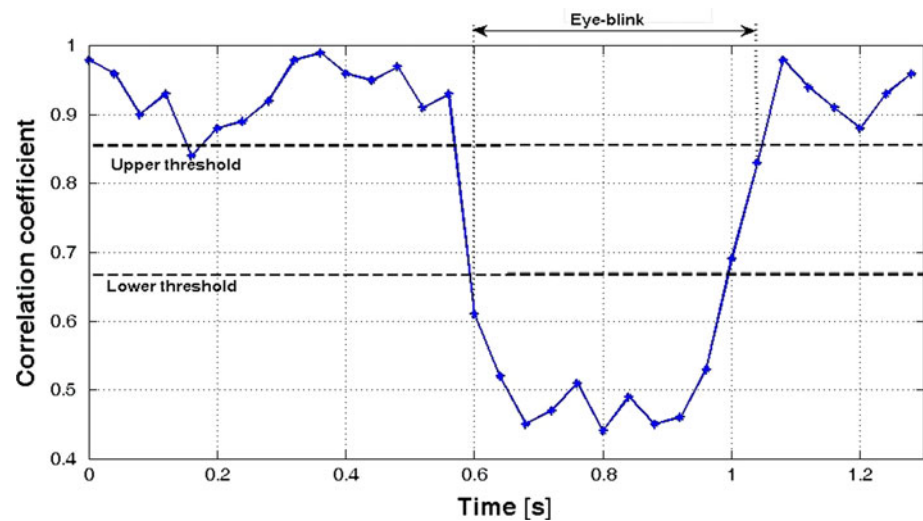
Fig. 9 Possible distribution of the eye-blink detector output

Table 2 Eye-blink detection system effectiveness parameters (for good illumination conditions)

Measure	Precision (%)	Recall (%)	Accuracy (%)
Long eye-blink detection	96.91	98.13	95.17
Short eye-blink detection	96.99	98.50	95.53
Overall system accuracy	96.95	98.31	95.35

Table 3 Eye-blink detection system effectiveness parameters (for poor illumination conditions)

Measure	Precision (%)	Recall (%)	Accuracy (%)
Long eye-blink detection	78.43	82.11	77.38
Short eye-blink detection	76.15	81.93	77.02
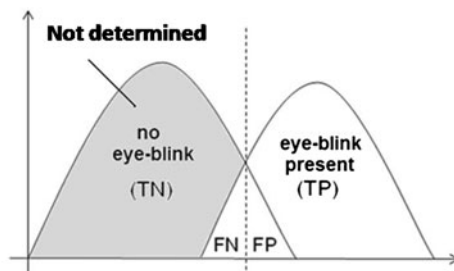
Negatives (FN). Based on these parameters three measures of the system performance were introduced: precision, recall and accuracy, defined by 2–4, respectively.

$$\text{precision} = \frac{TP}{TP + FP} \qquad (2)$$

$$\text{recall} = \frac{TP}{TP + FN} \qquad (3)$$

$$\text{accuracy} = \frac{TP}{TP + FP + FN} \qquad (4)$$

These three measures were used to assess the system robustness in detecting short eye-blinks and long eye-blinks separately, as well as for the overall system performance. The tests of the system in poor illumination conditions and with nonuniformly illuminated faces were also performed. The results are presented in Tables 2 and 3. Note that the achieved results are comparable to the performance rates reported for active vision-based eye-blink detection systems [9, 31]. In the proposed solution, the correct eye localization rate is the same as the correct face detection rate, that is 99.74%. This result locates the developed algorithm among the best algorithms available. The accuracy of the system is equal to approximately 95%, with precision of almost 97% and recall of 98%. These figures are close to the results obtained for eye-blink detection systems using active illumination techniques. The developed algorithm works at a rate of 25–27 fps for medium-class PC. These parameters show that the proposed algorithm fulfills the assumptions made to develop a reliable eye-blink detection system.
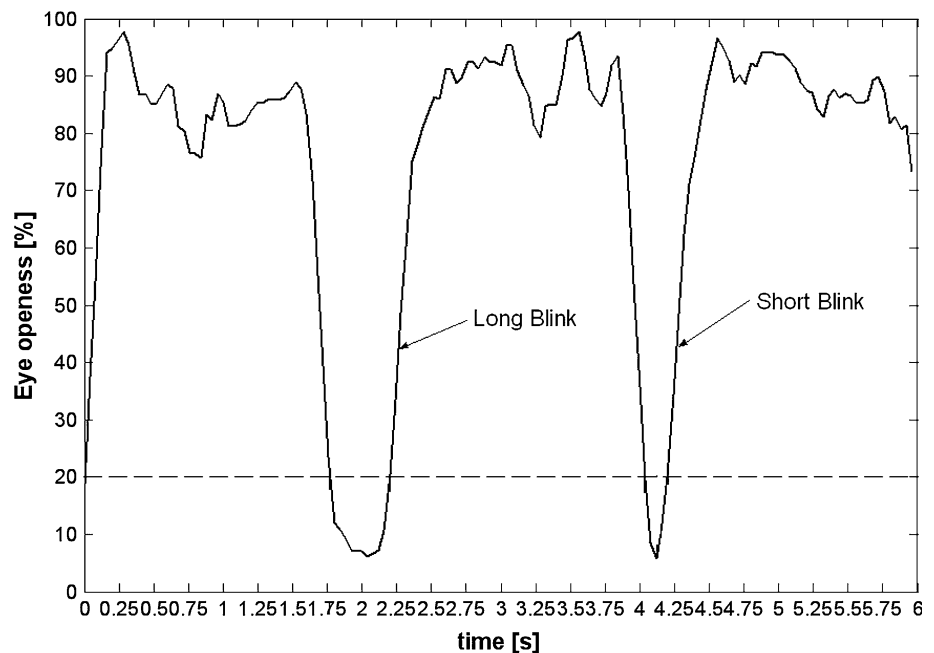
4 Eye-blink interaction

The algorithm for automatic detection of voluntary eye-blinks was employed in the development of a user interface. The applications were written using C++ Visual Studio and OpenCV library. The system is built from off-the-shelf components: an internet camera and a consumer garde personal computer. For best performance of the system, the distance between the camera and the user's head should be not greater than 150cm. The system set-up is presented in Fig. 10.

The proposed interface is operated by voluntary eye-blinks. The average duration of a spontaneous eye-blink is equal to about 100ms. The detected eye-blinks are classified as short (spontaneous) and long (voluntary) ones. In

order to avoid errors caused by interpretation of the spontaneous eye-blink as a voluntary one, the eye-blinks of duration larger than 200ms are considered as "control" ones and are interpreted as commands for the system. An example plot of eye openness in time with a long- and a short eye-blink is presented in Fig. 11.

The proposed interface, designed for Windows OS, has the following functionalities:

– loading and navigating web pages
– controlling the mouse cursor and emulating keyboard and mouse keys
– entering text to an arbitrary text editor, spreadsheet or messenger
– turning off the computer.

The main elements of the developed user interface are a virtual keyboard and a screen mouse. The operation of the interface is based on activating certain "buttons" of the virtual keyboard or mouse by performing control blinks. Subsequent buttons are highlighted automatically in a sequence. If the control blink is detected by the system, the action assigned to the highlighted button is executed. The user is informed about the detection of the eye-blink in two steps: a sound is generated when the start of the eye-blink is detected and another sound is played when the control blink is recognized by the system.

In case of the virtual keyboard, the alphanumeric signs are selected in two steps. The first step is selecting the column containing the desired sign. When the control blink is detected, the signs in the column are highlighted in a sequence and a second control blink allows for entering the selected letter to the active text editor.

Fig. 10 Test bench for the proposed user interface

Fig. 11 Eye-openness signal with an example of long and short blink detected

(a)

Shift		e	t	s	d	w	,	;
Ctrl	.	o	i	l	f	b	/	5
Alt	a	n	c	g	k	1	6	9
Bsp	h	u	m	j	2	7	0	-
Enter	r	y	x	3	8	[	=	PgUp
TAB	p	q	4	↑	\	`	'	PgDn
Esc	v	z	←	↓	→	]	del	Win

(b)

Shift		a	i	w	s	l	,	;
Ctrl	.	o	r	t	j	b	/	5
Alt	e	n	c	u	g	1	6	9
Bsp	z	y	m	h	2	7	0	-
Enter	k	p	f	3	8	[	=	PgUp
TAB	d	q	4	↑	\	`	'	PgDn
Esc	v	x	←	↓	→	]	del	Win

Fig. 12 On-screen keyboard arrangement for **a** Polish language, **b** English language

Shift		A	I	W	S	L	<	:
Ctrl	>	O	R	T	J	B	?	%
Alt	E	N	C	U	G	!	^	(
Bsp	Z	Y	M	H	@	&	)	_
Enter	K	P	F	#	*	{	+	Home
TAB	D	Q	$	↑	|	~	"	End
Esc	V	X	←	↓	→	}	del	Win

Fig. 13 Polish on-screen keyboard arrangement for "Shift" key selected

Shift	ą	ń	ż				
Ctrl	ó	ć	ł				
Alt	ę	ś	ź				
Bsp							
Enter							
TAB							
Esc							

Fig. 14 Polish on-screen keyboard arrangement for "Alt" key selected
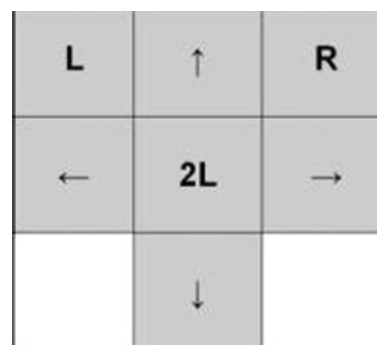
Fig. 15 On-screen mouse panel
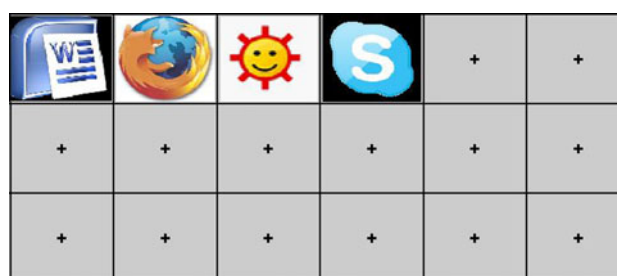
Fig. 16 Shortcuts panel

www.google.com	+
www.yahoo.com	+
www.youtube.com	+
+	+
+	-

Fig. 17 Bookmarks panel

Variuos keyboard arrangements have been proposed for the text entry, such as alphabetical, QWERTY or Scanning ambiguous keyboards (SAK) [30]. The alphanumeric signs on the virtual keyboard in the proposed system are arranged in such a away that the access time to the particular sign is proportional to the frequency of occurrence of this sign in a particular language. Therefore, for each language, a different arrangement of letters is required. The keyboard designs for Polish (a) and English (b) are illustrated in Fig. 12.

The on-screen keyboard also contains several function keys, such as "Shift", "Alt" or "Ctrl". The user may use the "Shift" key to enter capital letters or alternative button symbols (Fig. 13).

The "Alt" key allows the user to enter special signs, e.g., letters native to the Polish alphabet (Fig. 14). Both "Shift" and "Alt" keys, as well as other function buttons remain pressed until a desired key is chosen so that the desired keyboard combination is formed.

The screen mouse menu consists of seven function buttons and the "Exit" button (Fig. 15). The activation of one of the four arrow buttons causes the mouse cursor to move in the selected direction. The second control blink stops the movement of the cursor. Buttons L, R and 2L are responsible for left, right and double left click,

respectively. In this way, the screen mouse gives the user access to all functions of the Internet browser.

The developed interface also offers the user some stage of personalization. The user may define up to 17 shortcuts (Fig. 16) that are used to quickly run the chosen application, e.g., text editor, Internet browser or e-mail client. The shortcuts may be added, removed and substituted as new ones.

The bookmark menu allows the user to add links to most commonly visited websites (Fig. 17). Selecting an on-screen button with a bookmark causes the default Internet

browser to run and then to display the selected webpage. The last functionality is the shut down menu, which allows the user to safely exit all running programs and turn off the computer.

5 Results

The developed eye-blink controlled user interface was tested by 49 users: 12 participants with disabilities and a control group of 37 participants with no disabilities. All persons completed the obligatory training sessions lasting approximately 10–15 minutes. The testing sessions consisted of three parts: using the interface with a Polish virtual keyboard, using the interface with an English virtual keyboard and using web browsing. In the first and second part of the testing session, the subjects were asked to construct complete words or phrases in Polish and English. The input times were measured and expressed in seconds. The number of incorrectly entered characters was counted and divided by the number of characters in the original word. During the test runs, no corrections (backspacing) were allowed. The results are presented in Tables 4, 5, 6 and 7. The average input speed for the user interface with the Polish virtual keyboard was equal to 5.7 characters/minute, while for the English virtual keyboard, the input speed was equal to 6.5 characters/minute. The overall error ratio for the on-screen keyboard testing was equal to 0.4 (i.e., one incorrectly entered sign per 36 characters). The third part of the testing session concerned the assessment of Web browsing performance. The subjects were asked to perform the following tasks:

1. load a predefined Web page by activating appropriate button on the virtual keyboard
2. scroll the loaded Web page
3. move the mouse cursor to an indicated position and perform a click action
4. enter a given internet address and load a new Web page
5. move the cursor from the top left corner to the bottom right corner of the screen of 1,440 × 900 pixels resolution.

Table 4 Time performance of the proposed interface with English keyboard (results before training)

Word	Time range (s)	Avg. time (s)	SD
MAIL	30.4–76.3	40.1	±6.85
NAME	37.1–78.9	46.2	±7.02
MY NAME IS	91.3–142.7	111.8	±8.39
INFORMATION	98.1–157.9	132.5	±14.2
GOOD MORNING	104.2–178.3	141.3	±13.98

Table 5 Time performance of the proposed interface with English keyboard (results after training)

Word	Time range (s)	Avg. time (s)	SD
MAIL	27.3–50.1	35.03	±6.36
NAME	30.8–56.9	39.77	±6.95
MY NAME IS	82.6–109.3	93.45	±7.05
INFORMATION	82.9–131.2	100.57	±13.13
GOOD MORNING	93.1–140.3	108.27	±14.47

Table 6 Time performance of the proposed interface with Polish keyboard (results before training)

Word	Time range (s)	Avg. time (s)	SD
POCZTA	63.1–105.3	81.3	±8.37
IMIĘ	50.8–82.1	69.8	±6.42
NAZYWAM SIĘ	118.3–152.7	133.5	±9.61
WIADOMOŚĆ	119.1–146.8	127.4	±7.18
DZIEŃ DOBRY	122.4–165.2	142.7	±6.09

Table 7 Time performance of the proposed interface with Polish keyboard (results after training)

Word	Time range (s)	Avg. time (s)	SD
POCZTA	49.6–75.1	61.39	±7.89
IMIĘ	37.1–57.1	44.85	±5.34
NAZYWAM SIĘ	96.7–129.3	108.61	±9.25
WIADOMOŚĆ	90.6–116.0	99.71	±6.12
DZIEŃ DOBRY	95.8–129.4	111.94	±8.98

The results of the third part of the test are summarized in Table 8. After completing the testing sessions, 7 subjects complained about slight eye fatigue and tension.

The developed user interface was also presented to two persons suffering from athetosis. Athetosis is a continuous stream of slow, writhing, involuntary movements of flexion, extension, pronation and supination of the fingers and hands, and sometimes of the toes and feet. It is said to be caused by damage to the corpus striatum of the brain and can also be caused by a lesion of the motor thalamus.

The testing sessions with these two persons were organized with the cooperation of the Polish Telecom and the The Friends of Integration Association (Fig. 18). In case of one the impaired persons, the system failed in detecting user's eyes due to continuous head movements. The second person found the system useful and was ready to use the fully functional prototype.

The functionality of the interface was assessed in two ways: by estimating the time required to enter particular sequences of alphanumeric signs and by assessing the

Table 8 Eye-blink HCI time performance

Action	Mean time (s)	SD
Load predefined webpage (1)	6.21	±0.58
Scroll webpage (2)	10.48	±1.03
Move cursor to given position and click (3)	13.24	±1.38
Enter internet address: http://www.yahoo.com (4)	61.33	±7.81
Move cursor along the screen diagonal (5)	9.54	±0.51

Fig. 18 Tests of the eye-blink controlled user interface with the persons with disabilities

precision of using the screen mouse. Each user performance was measured twice: before and after the 2-h training session. The results are summarized in Tables 4 and 5. The average time of entering a single sign was equal to 16.8 s before the training session and 11.7 s after the training session. The average time of moving the cursor from the bottom right corner to the center of the screen was equal to 7.46 s. Also the percentage of detected control eye-blinks was calculated and it was equal to approximately 99%.

The users were asked to assess the usefulness of the proposed interface as good or poor. 91% of the testers described the interface as good. The main complaints were about difficulties with learning the interface and the necessity of long training. Also 84% of the users complained about eye fatigue appearing after approximately 15 min of using the proposed interface. It was caused by high intension in picking up the candidates on the screen. This problem may be reduced by introducing autofill predictive text options. The proposed interface, after some modifications, may be used for other purposes. In its reduced form it may be introduced in hospitals for severely paralyzed patients as a tool for informing about their basic needs or for calling medical personnel. The described eye-blink detection system may also be used for controlling external devices, such as TV sets, radio sets or wheelchairs by long, voluntary blinks.

6 Conclusions

Obtained results show that the proposed algorithm allows for accurate detection of voluntary eye-blinks with the rate of approximately 99%. Performed tests demonstrate that the designed eye-blink controlled user interface is a useful tool for the communication with the machine. The opinions of the users with reduced functioning were enthusiastic. The system was deployed to market as an open-source software by the Polish Telecom and the Orange Group as an interface for people with disabilities under the name b-Link (http://www.sourceforge.net/projects/b-link/).

References

1. Starner, T., Weaver, J., Pentland, A.: A wearable computer based American sign language recognizer. Assist. Technol. Artif. Intell. 84–96 (1998)
2. Materka, A., Byczuk, M.: Alternate half-field simulation technique for SSVEP-based brain–computer interfaces. Electron. Lett. **42**(6), 321–322 (2006)
3. Ghaoui, C.: Encyclopedia of Human Computer Interaction. Idea Group Reference (2006)
4. Thoumies, P., Charlier, J.R., Alecki, M., d'Erceville, D., Heurtin, A., Mathe, J.F., Nadeau, G., Wiart, L.: Clinical and functional evaluation of a gaze controlled system for the severely handicapped. Spinal Cord. **36**, 104–109 (1998)
5. Gips, J., DiMattia, P., Curran, F., Olivieri, P.: Using EagleEyes-an electrodes based device for controlling the computer with your eyes-to help people with special needs. In: Proceedings of the 5th International Conference on Computers Helping People with Special Needs, vol. 1, pp. 77–83 (1996)

6. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society, vol. 1, pp. 511–518 (2001)

7. Oguz, O.: The proportion of the face in younger adults using the thumb rule of Leonardo da Vinci. J. Surg. Radiol. Anat. **18**(2), 111–114 (1996)

8. Seki, M., Shimotani, M., Nishida, M.: A study of blink detection using bright pupils. JSAE Rev. **19**, 49–67 (1998)

9. Kocejko, T., Bujnowski, A., Wtorek, J.: Eye mouse for disabled. Conference on Human System Interactions, pp. 199–202 (2008)

10. Horng, W.B., Chen, C.Y., Chang, Y., Fan, C.H.: Driver fatigue detection based on eye tracking and dynamic template matching. In: Proceedings of of IEEE International Conference on Networking, Sensing and Control, pp. 7–12 (2004)

11. Królak, A., Strumiłło, P.: Fatigue monitoring by means of eye blink analysis in image sequences. ICSES **1**, 219–222 (2006)

12. Zhou, Z.H., Geng, X.: Projection functions for eye detection. Pattern Recognit. **37**(5), 1049–1056 (2004)

13. Kawaguchi, T., Hidaka, D., Rizon, M.: Detection of eyes from human faces by Hough transform and separability filters. In: Proceedings of International Conference on Image Processing, vol. 1, pp. 49–52 (2000)

14. Li, J.-W.: Eye blink detection based on multiple Gabor response waves. In: Proceedings of the 7th International Conference on Machine Learning and Cybernetics, pp. 2852–2856 (2008)

15. Bradski, G., Keahler, A., Pisarevsky, V.: Learning-based computer vision with intel's open source computer vision library. Intel Technol. J. **9**(2), 119–130 (2005)

16. Królak, A., Strumiłło, P.: Eye-blink controlled human–computer interface for the disabled. Advances Intell. Soft Comput. **60**, 133–144 (2009)

17. Yuille, A.L., Cohen, D.S., Hallinan, P.W.: Feature extraction from faces using deformable template. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 104–109 (1989)

18. http://www.a-s-l.co. Last visited May 2009

19. Magee, J.J., Scott, M.R., Waber, B.N., Betke, M.: EyeKeys: A real-time vision interface based on gaze detection from a low-grade video camera. Conference on Computer Vision and Pattern Recognition Workshop, CVPRW '04., pp. 159–159 (2004)

20. Yow, K.C., Cipolla, R.: Feature-based human face detection. Image Vis. Comput. **15**(9), 713–735 (1997)

21. McKenna, S., Gong, S., Raja, Y.: Modelling facial colour and identity with gaussian mixtures. Pattern Recognit. **31**(12), 1883–1892 (1998)

22. Lanitis, C.J., Taylor, T.F.C.: An automatic face identification system using flexible appearance models. Image Vis. Comput. **13**(5), 393–401 (1995)

23. Rowley, H.A., Baluja, S., Kanade, T.: Neural network-based face detection. IEEE Trans. Pattern Anal. Mach. Intell. **20**(1), 23–38 (1998)

24. Osuna, E., Freund, R., Girosi, F.: Training support vector machines: An application to face detection. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 130–136 (1997)

25. Rajagopalan, K., Kumar, J., Karlekar, R., Manivasakan, M., Patil, U., Desai, P., Poonacha, S., Chaudhuri: Finding faces in photographs. In: Proceedings of Sixth IEEE International Conference on Computer Vision, pp. 640–645 (1998)

26. Leinhart, R., Maydt, J.: An extended set of Haar-like features. In: Proceedings of International Conference on Image Processing, pp. 900–903 (2002)

27. Freund, Y., Schapire, R.E.: A short introduction to boosting. J. Jap. Soci. Artif. Intell. **14**(5), 771–780 (1999)

28. Yang G., Huang, T.S.: Human face detection in complex background. Pattern Recognit. **27**(1), 53–63 (1994)

29. Grauman, K., Betke, M., Lombardi, J., Gips, J., Bradski, G.R.: Communication via eye blinks and eyebrow raises: Video-based human–computer interfaces. Universal Access in the Information Society, **2**(4), 359–373 (2002)

30. MacKenzie, I.S., Ashtiani, B.: BlinkWrite: efficient text entry using eye blinks. Universal Access in the Information Society, Online First (2010)

31. Magee, J.J., Scott, M.R., Waber, B.N., Betke, M.: EyeKeys: A real-time vision interface based on gaze detection from a low-grade video camera. Conference on Computer Vision and Pattern Recognition Workshop, pp.159–159 (2004)

32. Safety of laser products—Part 1: Equipment classification and requirements (2nd ed.), International Electrotechnical Commission (2007)