The Gaze Tracking System with Natural Head Motion Compensation

Vidas RAUDONIS[1], Agnė PAULAUSKAITĖ-TARASEVIČIENĖ[2], Laura KIŽAUSKIENĖ[3]

[1]*Kaunas University of Technology, Department of Control Technologies*
Studentu 48-325, LT-51367 Kaunas, Lithuania
[2]*Kaunas University of Technology, Department of Business Informatics*
Studentu 56-442, LT-51424 Kaunas, Lithuania
[3]*Kaunas University of Technology, Department of Computer Engineering*
Studentu 50-214, LT-51368 Kaunas, Lithuania
e-mail: vidas.raudonis@ktu.lt, agne@ifko.ktu.lt, losta@ifko.ktu.lt

Received: March 2011; accepted: October 2011

Abstract. In this paper the gaze tracking system based on the adaptively changing threshold value of the gray level, which automatically detects the pupil position in two dimensional data is proposed. The system detects closed eye using normalized accumulative luminosity function. The detection of closed eye allows the confirmation of voluntary selected command in a more natural way. The presented technique allows recalibration without intervention or help of other assistive persons. Natural head motion is evaluated, employing signals, generated by orientation sensor. Within two experiments, the proposed system is tested at different behavior modes in respect of speed and accuracy of the system.

Keywords: eye tracking, artificial neural networks, adaptive threshold, head motion compensation, calibration.

1. Introduction

The input devices, that allow to interact with computers or other physical devices without actually touching them opens a new communication potential for people with physical impairments (such as amyotrophic lateral sclerosis, multiple sclerosis, cerebral palsy and etc). For this purpose the implemented systems use different actuating forces such as speech, brain signals or the eye gaze. During the last decade, the interest in such technology devices, so-called gaze tracking systems, has been rapidly growing. This communication system works using devices (e.g., video camera) that capture fast motions of user eye and estimate the gaze processing captured images. The system detects the position of the eye automatically and assigns appropriative commands to the external device or/and application program. This is a novel and perspective technique to control an application without a contact with a real object. The amount of the intensive investigation, shows that the assistive technologies have huge potential of practical usage in everyday life and more (Dzemydiene *et al.*, 2010; Heide and Trivino, 2010).

There are a number of methods for extracting the point of the gaze or the motion of the eye relative to the head from the captured eye or face image (Wu *et al.*, 2010; Grigaitis *et al.*, 2007). Gaze detection systems that are based-on special hardware and sensors are also explored (Cihan *et al.*, 2008; Fabio, 2008; Langner *et al.*, 2009; Shigeoka *et al.*, 2008). However, more handy and practical video-oculography based methods are considered in this work. The choice of the eye tracking method depends on the consideration of cost and application. For instance, low computational complexity tracking algorithm can be used by applying symmetry detector for rough pupil localization and triangulation for detecting the accurate pupil position (Kumar *et al.*, 2009). Experiments have shown that the algorithm was able to detect the pupil in 96% of cases and it is robust enough against drastic changes of ambient luminosity. However, the proposed system may by uncomfortable, because of its weight and size, since it consists of four video cameras and two servo drives that are mounted on the glasses. Another method to create the eye tracking algorithm is to combine feature-based and model-based approaches (Dongheng *et al.*, 2006). The device consists of scene and eye camera mounted on the glasses. Eye camera works in the near infrared (IR) light. Researches refer to this algorithm as "starburst" because of the way how the pupil features are obtained (Dongheng *et al.*, 2005). The ambient illumination occludes the Purkinje images. The pupil detection in IR images can be established using the algorithm of symmetric mass center (Long *et al.*, 2007). The pupil is detected in a binarized eye image that is acquired referring to the user defined threshold. Such predefined threshold can not adaptively change depending on the ambient IR luminosity. A slightly bigger search window than the actual pupil size can not capture the whole dynamical motion of the eye. The accuracy detection of the gaze vectors strongly depends on the position measurements' quality of corneal reflection (CR) and pupil centers. The task of the CR detection often becomes non-trivial, when the system works in an uncontrolled ambient illumination or the eye rotates to extreme positions. Gaze tracking technique based on the structured IR illumination overcomes the mentioned difficulties (Kolakowski and Pelz, 2007). The light source is designed from nine IR emitting diodes that are placed in certain distance from each other and directed to the user's eye. Although, such technique is capable of measuring a wide range of eye movement, the system blocks a large portion of the view field. The main disadvantage of the proposed algorithms is that they cannot be used outdoors during daytime due to the ambient infrared illumination. It is uncertain how such mentioned systems react to the different head poses or to the changes of calibration conditions.

Head mounted devices are a good option when accurate gaze detection is needed because they allow relatively free head movements. Most head mounted trackers use a second video camera, which captures the scene view and allows tracking objects of interest (Topal *et al.*, 2008; Sugita *et al.*, 2007). The angular rotation of the eye ball relative to the head orientation and the position of the head relative to the observable scene have to be measured in order to compute the accurate gaze vector. Yu and Eizenman (2004) have developed the gaze-point tracking methodology that uses features extracted from the scene camera and the homography algorithm with distortion compensation. The head location and orientation is determined according to the location of the objects in the scene. The presented point of gaze tracking system can be used to assess visual patterns with 0,9 angular accuracy. The methodology has been tested in the artificial environment, consequently the stability of the algorithm is uncertain in the outdoor.

All eye tracking systems must deal with frustrating problem of "Midas touch", i.e., the fixated gaze can have two meanings (Istance *et al.*, 2008). The combination of the gaze and the head tracking techniques discriminates more efficiently the voluntary selection of the command from the gaze fixation (Nonaka, 2003). The head gesture is employed to generate the decision, i.e., shaking, nodding and inclining of the head. The tree dimensional head orientation is measured using special device, which measures the position and orientation of certain active coil (transmitter) fixed on the head.

It is obvious, that the designed gaze tracking system must satisfy strong requirements, that come from the users of the target group: the system must be stable and work steadily in different lightning conditions; the user should be able to calibrate and recalibrate the system easily and independently; the system should be portable, flexible and as miniature as possible, i.e., easy to mount and move with electrical wheelchair; it should be possible to use the system for emailing, internet browsing, writing, gaming and etc. Most of the reviewed systems are applicable to certain cases, but still struggle with three main tasks: accurate pupil detection, compensation of natural head motions and "Midas touch" problem. In this paper the novel gaze tracking system is proposed that consist of three modules: pupil detection is based-on adaptive grayness thresholding, natural head motions are evaluated using non-linear technique and "Midas touch" task is solved using accumulative luminosity.

The rest of the paper is organized as follows. In Section 2 the gaze tracking system is introduced. All steps for system creation are described and explained: the development of pupil detection algorithm based on adaptive gray level threshold; the detection of closed eye based on accumulative luminosity function; cursor stabilizing and calibration and the natural head motion compensation. The description and investigation results of non-linear head motion compensation algorithm are described in Section 3. The testing, which evaluates proposed system executing two experiments, is presented in Section 4. Finally, Section 5 provides the concluding remarks of the study.

2. The Approach of the Gaze Tracking System

The identification of the eye pupil in the visual light is a complex task, because there are more features than in the infrared light, i.e., the reflection of the visual light from the ocular surface can mask the whole pupil. A human eye can't see the near infrared light, unlike the visual light. The external IR light source that does not cause eye discomfort can be used to illuminate the eye and gives mostly stable ambient IR luminosity. The eye image captured in the near infrared light always shows dark (black) pupil that is caused by the absorption of infrared light of the eye tissues. Thus, the task of pupil detection can be limited or simplified to the searching goal of the dark and round region in the image.

The constant adaptation of the pupil, the random variation of illumination condition and the natural motion of the head are the main challenges for gaze tracking algorithms.

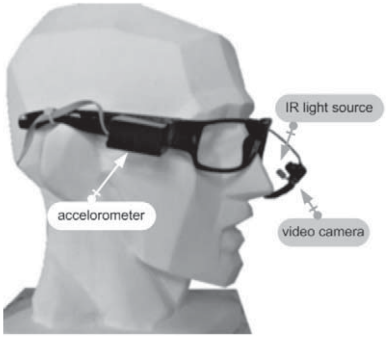


Fig. 1. Proposed monocular eye tracking system.

In this work we propose the algorithm, which is based on three fundamental processes: the evaluation of accumulative luminosity function, adaptive thresholding of grayscale images and the measurements of the head orientation. The luminosity function is used to separate the images of the closed and opened eye. The adaptive thresholding of grayscale images enables the precise detection of pupil in the different layers of gray color, regardless of how the lightening is changing. The diameter measurements of the dark region are compared with the limits of possible minimal and maximal pupil size. The control of the external device strongly depends on the calibration conditions and how they are maintained during the process.

The algorithm uses monocular video camera which is mounted on the glasses and directed to the user's eye. The user's eye is illuminated with one IR light diode. Head orientation in two directions is measured using accelerometer, which is attached to the side of the glasses. The signals from the accelerometer are used for the head pose compensation. The hardware of the proposed eye tracking system is shown in Fig. 1.

2.1. *Adaptive Threshold Based on the Real Time Measurements*

An adaptive property of the algorithm is obtained using iterative computation. The pupil is detected iteratively changing grayness level of the image and measuring geometrical properties of the observable object. The geometrical center, area and perimeter are measured in object recognition tasks.

The geometric center of the non-overlapping closed polygon by n vertices (x_i, y_i) is calculated using (1), (2) and (3). The area of the non-self intersecting polygon is written as:

$$A = \frac{1}{2}\sum_{i=0}^{N-1}(x_i y_{i+1} - x_{i+1} y_i). \tag{1}$$

The coordinates of the geometrical center C = (C_x, C_y) of the polygon, where

$$C_x = \frac{1}{6A}\sum_{i=0}^{N-1}(x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i), \quad x_N = x_0,\ y_N = y_0, \tag{2}$$

$$C_y = \frac{1}{6A}\sum_{i=0}^{N-1}(y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i), \quad x_N = x_0,\ y_N = y_0. \tag{3}$$

In (1), (2) and (3), the vertices are assumed to be numbered by the order of their occurrence along the polygon's perimeter, ant the vertex (x_N, y_N) is assumed to be the same as (x_0, y_0).

Proposed gaze tracking Algorithm 1 finds the accurate pupil center in the indexed image, which is indexed based on the adaptive gray level threshold. The algorithm is proceeded in two major steps. The rough pupil center is obtained in the first, and the accurate coordinates of the pupil are obtained in next step. The detection of the pupil is executed in the region of interest (sliding window) that is defined by three parameters *Length*, *Width* and the center coordinates (C_x, C_y). All logical indexing operations are executed in the region of interest. At the detection beginning, the center of eye image is considered as the starting position for the sliding window. All other positions are defined by located pupil center in the last frame. Values of all pixels higher than default grayness threshold Θ_0 are equalized to one, otherwise to zero. The threshold Θ is increased or reduced from Θ_0 according to the certain conditions defined by the current measured diameter r of the object. The threshold is increased by step $\Delta\Theta$, when r is smaller than the possible limits of the pupil size $[R_{\min} R_{\max}]$, otherwise it is decreased. The variation limits of the pupil size are taken according to analytical research (Pivrikas, 2006). The adaptation process is stopped when the number of iterations exceeds the limit, which is equal to $\mathrm{count}_{\mathrm{limit}}$. This parameter is set to 250, which covers almost all gray levels on the eye image. The threshold value is equalized to the default value Θ_0 and measurement r becomes equal to infinity in this case. If the threshold value exceeds the limits of the grayness $[\Theta_{\min}\Theta_{\max}]$, the detection of the rough pupil center (C_x, C_y) is stopped.

Algorithm 1. Pupil detection based on adaptive gray level threshold.

The rough center of pupil is extracted from the grayscale image of the eye $\Gamma(u, v)$, where $u \in N, v \in M$.

```
while $(\mathrm{flag} = 0)$ do
  $\mathrm{count} = \mathrm{count} + 1$
  $k = 0;$
  for $u \in N$ do
    for $v \in M$ do
      if $\Gamma(u, v) \leqslant \Theta$ then
        $k = k + 1;$
        $x_k = u, \qquad y_k = v;$
      end if
    end for
  end for
  $k \in K;$
  for $i \in K$ do
    for $j \in K$ do
```

```
            $d(i,j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$;
        end **for**
    end **for**
    $r = \max_{i \in K}(\max_{j \in K}(d(i,j)))$;
    $C_x, C_y \quad \Leftarrow \quad x(k), y(k)$
    **if** $R_{\min} \leqslant r \leqslant R_{\max}$ **then**
        flag $= 1$;
    **else if** $R_{\min} > r$ **then**
        $\Theta = \Theta + \Delta\Theta$;
    **else if** $R_{\max} < r$ **then**
        $\Theta = \Theta - \Delta\Theta$;
    end **if**
    **if** $\Theta_{\min} \leqslant \Theta \leqslant \Theta_{\max}$ OR $\text{count} > \text{count}_{\text{limit}}$ **then**
        flag $= 1, \qquad \Theta = \Theta_0, \qquad r = \infty$;
    end **if**
end **while**
```

The accurate pupil center $C' = [C'_x, C'_y]$ is obtained using the non-linear circle fitting method that improves the pupil center detection ($\sim$ 3–4 pixels) and increases the overall control accuracy. The additional actions executes the least squares method that fits the smallest surrounding circle on the selected points which belongs to the eye pupil. The coordinates $C = [C_x, C_y]$, are initial circle center for the circle fitting method and three-sigma rule on the candidate points x, y is used. The points that are three times farther from the midpoint than the dispersion are eliminated.

The explicit pupil center $\mathbf{C}' = (C'_x, C'_y)$ of the point cloud x_k, y_k is computed using non-linear least squares approach.

```
**if** r $\neq \infty$ **then**
**step 1** *Perform data filtration based-on three sigma rule on the candidate points*
    $d_i \quad \Leftarrow \quad \sqrt{(\bar{x} - x_i)^2 + (\bar{y} - y_i)^2}, \quad i \in K$;
    $\bar{d}, \sigma_d \quad \Leftarrow \quad d(i), \quad i \in K$;
    $n = 0$;
    **for** $i \in K$ **do**
        **if** $\bar{d} - 3\sigma_d \leqslant d(i) \leqslant \bar{d} + 3\sigma_d$ **then**
            $n = n + 1$;
            $x'_n = x_i, \qquad y'_n = y_i$;
        **end if**
    **end for**
**step 2** *Fit smallest surrounding circle to the filtrated data*
    $\mathbf{u} = (C'_x, C'_y, r'), \qquad \mathbf{X} = \begin{pmatrix} x'_1 & \dots & x'_n \\ y'_1 & \dots & y'_n \end{pmatrix}, \quad \mathbf{X} \in R^2$;
    $f_j(C'_x, C'_y, r') = \|\mathbf{C}' - \mathbf{X}_j\|^2 - r^2$;
    $C'_x, C'_y, r' \quad \Leftarrow \quad \sum_{j=1}^m f_j(\tilde{\mathbf{u}})^2 = \min_u \sum_{j=1}^m f_j(\mathbf{u})^2$;
**end if**
```

The computational actions of the proposed algorithm, taken for pupil detection, are shown in Fig. 2. The threshold Θ is iteratively changed and the distance between two extremities of the selected pixels is measured. These pixels usually appeared in opposite direction from each other. The pixels are marked with lighter color in Fig. 2. Θ is decreased from 130 to 120 gray level and the same measurements are applied to the new selected pixels. Gray threshold is reduced until the distance between extremities is in the possible variation range of the pupil size. The grayness value was reduced to 90 of the gray level in this case. If selected pixels do not satisfy the predefined condition then Θ is changed. The threshold is increased when the distance between two extremities is smaller than minimum limit of the pupil size variation and decreased when this distance exceeds predefined maximum The least square method is applied to selected pixels and the new pupil center C' is obtained.

The possibility of the proposed algorithm to change the threshold value adaptively overcomes several detection difficulties, such as, the variation of an ambient luminosity, constantly adapting pupil size and noise.

The resulting pupil detection examples are shown in Fig. 3. The black circle is corresponding to the fitted circle using least square method. The results show that the pupil detection can be efficient and accurate even on the image which is extreme noisy (Fig. 3e) or captured in non-uniform luminosity (Fig. 3a).

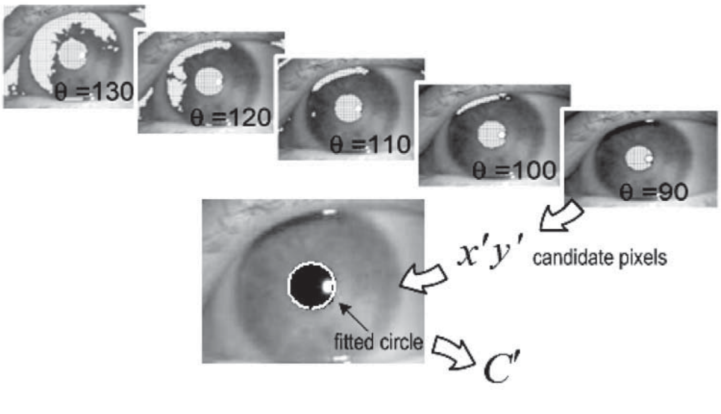


Fig. 2. The process of the adaptively changing threshold.

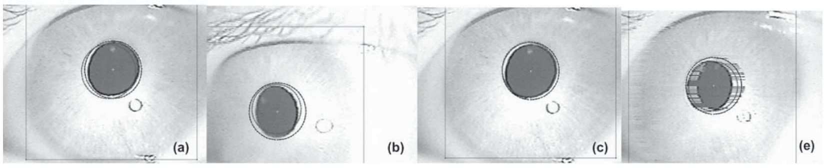


Fig. 3. The examples of the detected pupil, where candidate pixels are defined with red circle and fitted circle is colored in black.

2.2. *Detection of Closed Eye*

According to “Midas touch”, an additional signal (the image of the closed eye) has been introduced that allows confirming voluntary generated decision, i.e., pushing the button or activating of recalibration process. The accumulative luminosity function is used as a feature to distinguish the differences between the opened and closed eye images. The normalized accumulative luminosity is the distribution function of total brightness along the columns of the captured eye image. The normalized value of the total luminosity of the single column is evaluated summing all pixels in uth $(u \in N)$ column and normalizing it using (5). The resulting luminosity function varies between 0 and 1 (highest luminosity).

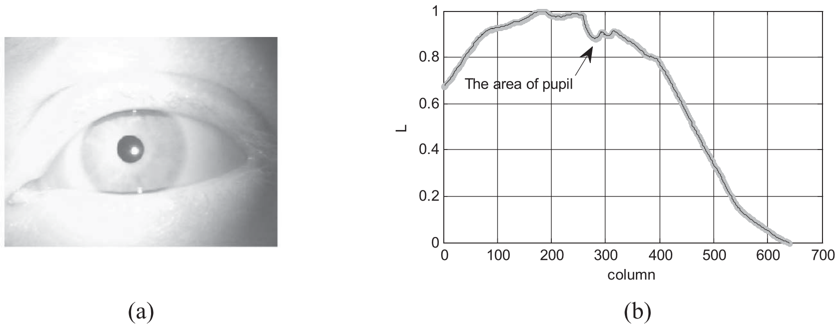
$$L(v) = \sum_{u=1}^{N} \Gamma(u,v), \quad u \in N,\ v \in M, \tag{4}$$

$$L^n(v) = \frac{L(v) - \min(L)}{\max(L) - \min(L)}. \tag{5}$$

The images of the closed and opened eyes have different functions of the accumulative luminosity, which can be used as discriminating two different classes feature. The comparison of the features can be evaluated using correlation. The correlation coefficient ς indicates the strength and direction of the relationship between the two signals. It has clear variation boundaries $[-1, 1]$. The experimental computation is executed in order to find proper similarity threshold ϑ between the template and the captured luminosity functions of the images.

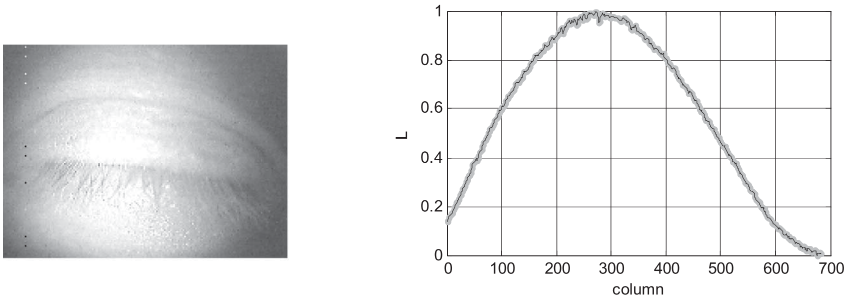
The examples of the luminosity functions of the closed and opened eye are shown in Figs. 5 and 4 accordingly. The function of the closed eye has a shape that is similar to the Gaussian distribution function. Such form is obtained, because the IR light is reflected by the skin surface. The luminosity function of the opened eye is different from the closed eye, because a part of the IR light is absorbed by the inner eye tissues.

The conditions of the ambient luminosity often vary randomly. The accumulative luminosity function of the image is changing during the time as well. The law pass filter is used to update the luminosity function during the operation time $t(L_t)$ and $t - 1(L_{t-1})$.



(a) (b)

Fig. 4. The accumulative normalized luminosity function (b), of the image with opened eye (a).



(a) (b)

Fig. 5. The accumulative normalized luminosity function (b), of the image with closed eye (a).

The proportions of each luminosity functions depend on the coefficients α and β. The sum of the coefficients must be equal to one. The law pass filter is expressed using (6).

$$\mathbf{L} = \alpha \cdot \mathbf{L}_t + \beta \cdot \mathbf{L}_{t-1}, \alpha + \beta = 1. \tag{6}$$

The luminosity function is unchanged during the time if the coefficient β is significantly higher than the coefficient α. The default luminosity function is used in this case and the gaze tracking algorithm does not react to the ambient luminosity variations. Such insensitivity can reduce the detection accuracy of the closed eye, and moreover, voluntary controlled selection. Otherwise, too high sensitivity to ambient luminosity changes can disable the whole algorithm, because it will try to adopt the adaptive thresholding method on the closed eye image.

The initial luminosity function is computed from the first n (∼30) frames, which is predefined by the user of the gaze tracking system. The detection of the closed eye is based on the similarity measure and certain threshold value. The image is classified as closed eye, when the similarity measure is higher than certain ϑ, and as opened eye, otherwise. The evaluation of the adaptive accumulative luminosity is executed in three steps:

```
step 1 compute initial luminosity of image
  $L_0 \Leftarrow \Gamma_0(u,v)$ step 2 compute the luminosity at time t
  $L_t \Leftarrow \Gamma_t(u,v)$
step 3 check update condition
  if $\varsigma > \vartheta$ then
    $\mathbf{L} = \alpha \cdot \mathbf{L}_t + \beta \cdot \mathbf{L}_{t-1}$
  end if
```

The selection of the similarity measure depends on the obtained accuracy and the processing rate selected during experimental investigation.

2.3. *Algorithm for Computer Mouse Control*

The calibration is the set of processes that establish relationship between the center coordinate of the pupil in the image, computer mouse position and concrete command. The changes of pupil center location are very small with respect to the computer screen size. The pupil can move about 120 pixels horizontally and about 100 pixels vertically and even a small shift of the pupil center (1 pixel) leads to the dislocation of the eye controlled cursor. Moreover, such sensitivity creates the so called jumping effect, which does not allow holding the cursor in a fixed position. The jumping effect is a big challenge for e-mouse control, which requires accurate evaluation of the gaze vector. To avoid this problem, such techniques as head motion compensation, cursor stabilizing and calibration of the system are used.

The cursor control problem in our paper is solved using certain calibration grid which stabilizes the cursor position on one point, if the pupil remains in one predefined region. The grid is acquired during the system calibration or recalibration process. For this process the user must observe four corner points of the computer screen. The coordinates of the pupil position when the gaze was directed to the corner point are used to obtain certain calibration field. The grid, which is build from N even sectors, where $N = \overline{1, n}$, is separately applied to the computer screen and the area of possible pupil trajectory.

Figure 6 shows the principle of the position assignation of the mouse cursor. Each area on the screen has a unique number that is equal to the number of the area on the eye image. The algorithm shown bellow assigns the center coordinates of the area to the cursor position having the same number as the area in which the pupil center $C' = \mathbf{P}$ appears on the eye image. In such manner the mouse cursor stays in one position despite the fact that small changes of the pupil position are still appearing within one region of the defined area. The jumping effect of the mouse cursor still remains, when these minor changes occur near the regional intersection. The Algorithm 2 is based on two main steps.

Algorithm 2. The interpolation of the cursor position on the screen

```
step 1 Divide the computer screen to N square regions
    $\mathbf{X}_{\text{scr}} = [x_1 \ldots x_N]$,        $\mathbf{Y}_{\text{scr}} = [y_1 \ldots y_N]$;
```

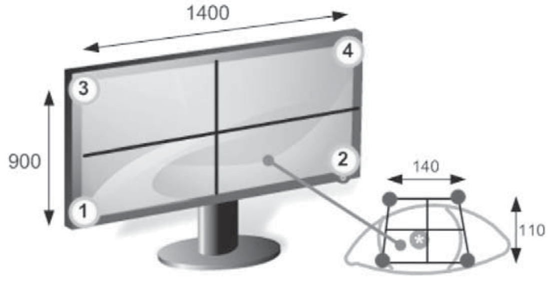


Fig. 6. The calibration field or grid.

```
step 2 Get cursor position according the pupil center P = $[x_{cp}, y_{cp}]$
  for $i \in N-1$ do
    for $j \in N-1$ do
      $Q_1^v = [\,X_{cf}^i\ Y_{cf}^j\,]$,    $Q_2^v = [\,X_{cf}^{i+1}\ Y_{cf}^j\,]$,    $Q_3^v = [\,X_{cf}^i\ Y_{cf}^{j+1}\,]$,
      $Q_4^v = [\,X_{cf}^{i+1}\ Y_{cf}^{j+1}\,]$;
      $ad = \det\left(\begin{bmatrix} Q_2^v - Q_1^v \\ P - Q_1^v \end{bmatrix}\right) / \|Q_2^v - Q_1^v\|$; // $ad$ – the point distance to lower
                                                    line of $N$ gaze region
      $vd = \det\left(\begin{bmatrix} Q_4^v - Q_3^v \\ P - Q_3^v \end{bmatrix}\right) / \|Q_4^v - Q_3^v\|$; // $vd$ – the point distance to upper
                                                    line of $N$ gaze region
      $Q_1^h = [\,X_{cf}^i\ Y_{cf}^j\,]$,    $Q_2^h = [\,X_{cf}^i\ Y_{cf}^{j+1}\,]$,    $Q_3^h = [\,X_{cf}^{i+1}\ Y_{cf}^j\,]$,
      $Q_4^h = [\,X_{cf}^{i+1}\ Y_{cf}^{j+1}\,]$;
      $kd = \det\left(\begin{bmatrix} Q_2^h - Q_1^h \\ P - Q_1^h \end{bmatrix}\right) / \|Q_2^h - Q_1^h\|$; // $kd$ – the point distance to left line
                                                    of $N$ gaze region
      $dd = \det\left(\begin{bmatrix} Q_4^h - Q_3^h \\ P - Q_3^h \end{bmatrix}\right) / \|Q_4^h - Q_3^h\|$; // $dd$ – the point distance to rigth
                                                    line of $N$ gaze region
      if ad > 0 AND vd < 0 AND kd < 0 AND dd > 0 then
        $x_{\text{cursor}} = 0.5 \cdot (\sum_{k=1}^{2} X_{\text{scr}}^k)$,    $y_{\text{cursor}} = 0.5 \cdot (\sum_{k=1}^{2} Y_{\text{scr}}^k)$;
      end if
    end for
  end for
```

The assignation of cursor position is realized in two main steps. First step divides the whole computer screen to N even regions, where each region is defined by its own position coordinates $\mathbf{X}_{\text{scr}}$ and $\mathbf{Y}_{\text{scr}}$. The coordinates $x_{\text{cursor}}, y_{\text{cursor}}$ of the cursor position are defined in the next step. These coordinates are assigned depending on which image area appears the pupil. The corner points of the calibration field are located at the peaks of the trapezoid, due to the fact that the eye is roughly spherical. Therefore, the calibration field is divided by vertical lines derived between points $Q_1^v \dots Q_4^v$ and horizontal lines derived between points $Q_1^h \dots Q_4^h$.

2.4. *Natural Head Motion Compensation*

The natural head motion has high impact on the control accuracy, because of the changing calibration conditions. Recalibration process must be executed in order to regain proper control using head mounted eye tracking device. The usage of the natural head motion compensation allows reducing the recalibration need.

The signal captured by orientation measuring device is used to design the technical facility for the eye tracking system. The device is attached to the eyeglasses together with

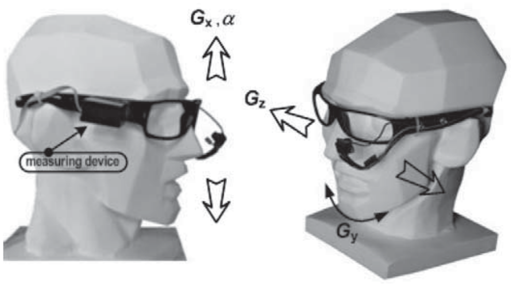


Fig. 7. Measurements of the leaned (G_z), rotated (G_y) and tilted (G_x, α) head.

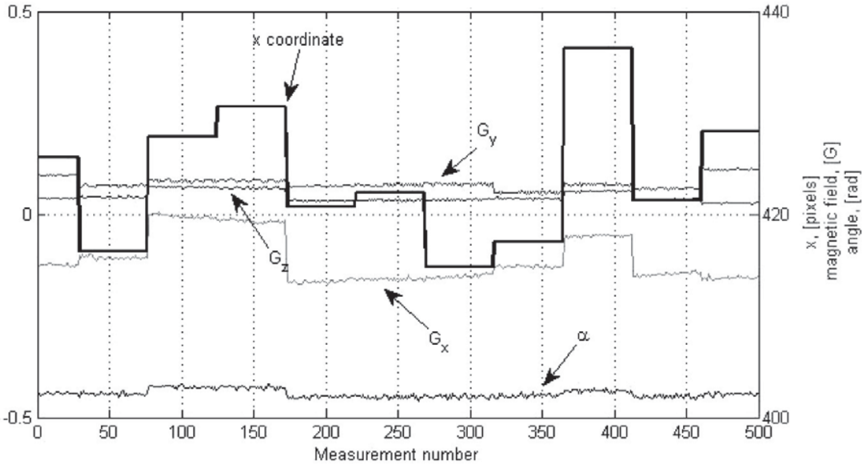


Fig. 8. The measurements of the head orientation and x coordinate of the first corner of the calibration field.

the video camera and can measure all head poses, that are shown in Fig. 7. The device consists of one axis accelerometer and three-axis magnetic field B sensor that measures the direction of the magnetic field in gauss G. The head movement compensation is realized using artificial neural networks, which finds the relationship between the measured direction of the magnetic field and the corners of the calibration field.

Instead of using a limited head model, the artificial neural network is applied to find the relation between the four calibration corners and the measured head orientation. The measurements example of the first corner coordinate x of the calibration field and angles is shown in Fig. 8. It is clearly visible that even relatively minor displacement of the head induces considerably large offset of the field coordinate. The relation between the measured angle values and the coordinates is non-linear, so a properly trained neural network could be able overcome the non-linearity and give desirable compensation accuracy.

The concept of the non-linear head motion compensation algorithm is shown in Fig. 9. Eight different neural networks are used instead of using one complex neural network with eight outputs. Each network has one output which generates the certain

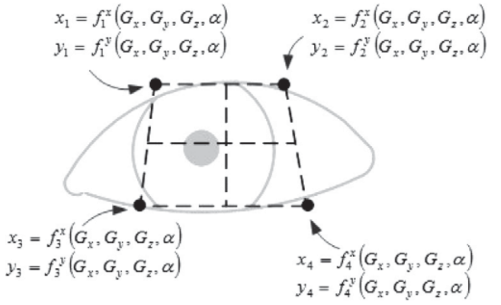

$x_1 = f_1^x(G_x, G_y, G_z, \alpha)$
$y_1 = f_1^y(G_x, G_y, G_z, \alpha)$
$x_2 = f_2^x(G_x, G_y, G_z, \alpha)$
$y_2 = f_2^y(G_x, G_y, G_z, \alpha)$
$x_3 = f_3^x(G_x, G_y, G_z, \alpha)$
$y_3 = f_3^y(G_x, G_y, G_z, \alpha)$
$x_4 = f_4^x(G_x, G_y, G_z, \alpha)$
$y_4 = f_4^y(G_x, G_y, G_z, \alpha)$

Fig. 9. The concept of the natural head motion compensation.

coordinate value based on the inputs. The neural network is described as a function $\zeta = f(G_x, G_y, G_z, \alpha)$, where ζ is x or y coordinate.

The ANN-based compensation algorithm can be considered as a “black box” model and it is essential advantage of the proposed algorithm. It finds the relation based on the training data not using certain predefined coefficients or parameters.

3. Selection of Optimal ANN Structure

Experimental testing with non-linear head motion compensation algorithm was executed with specially gathered data that consisted of: (1) the coordinates of the four corners of the calibration field; (2) magnetic field strength; (3) one-dimensional accelerometer’s signal. The sensors’ signals and coordinates were gathered during the calibration process, where a person was asked to observe four monitors’ corners time at a time. The orientation of the head was changed each time when new calibration started and kept in this position until calibration ended. $n = 61$ different head positions were measured and recorded in such way. 60 percent of the raw data was used for neural network training and the rest of the data – for the experimental testing. We took the view that the length and the width of the observable computer screen is equal to 20 cm and 30 cm respectively, and user sits in front of the computer screen.

The coordinates of the calibration field, obtained during the calibration process, were compared with the coordinates estimated using proposed non-linear head compensation algorithm. The experimental results of the compensation of the natural head motion are shown in Fig. 10. The calibration field that is obtained during the first system calibration is marked in a solid grey line, experimentally measured calibration field is marked in a dashed line, the resulting calibration field is presented with a dashed solid black line. The experimental results show that the non-linear compensation algorithm is efficient and accurate when head orientation changes from the initial orientation up to to 30 degrees into three different directions. The accuracy directly depends on the structure of the used neural network (Fig. 11).

During the experiments the neural network’s structure was alternated due to the number of neurons together with the transfer function in a hidden layer. Three different transfer functions were used: hyperbolic tangent sigmoid function (tansig), saturating linear

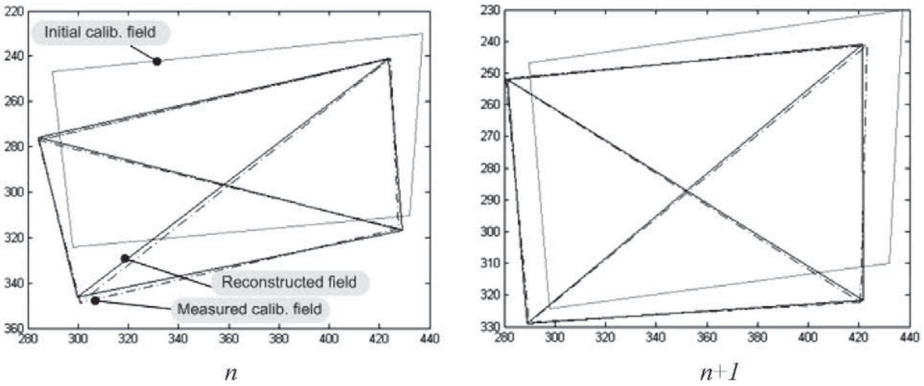


Fig. 10. The calibration fields in the different head orientations (n and $n+1$) obtained during experimental investigation.

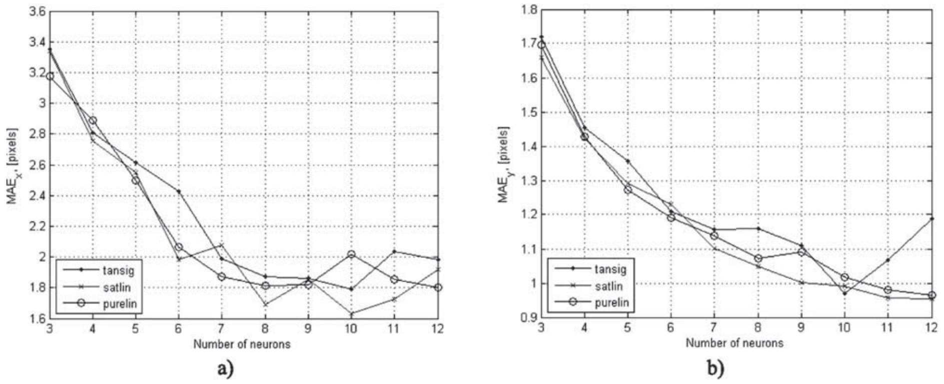
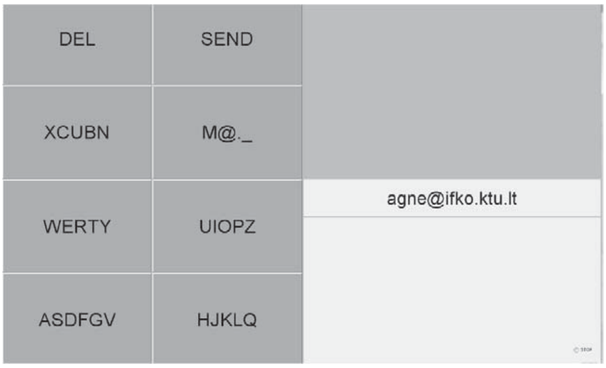


Fig. 11. Functional relationship between MAE ((a) of x coordinates, (b) of y coordinates) and the number of neurons in hidden layer of network.

function (satin) and linear transfer (purelin) function (see Fig. 11). The experimental investigation has shown that higher accuracy is obtained using relatively more neurons in a hidden layer. For instance, in the case of 3 neurons $MAE_{xy} = 4.8$, in the case of 8 neurons $MAE_{xy} = 2.75$, but in the case of 12 neurons $MAE_{xy} = 2.90$. The influence to the reconstruction accuracy of the transfer functions is insignificant. It follows, that for the tracking system testing application it is reasonable to use a neural network with 10 neurons in a hidden layer with saturating linear transfer function.

4. The Experiments with Real-Time Applications

Two different experiments were performed in order to evaluate the accuracy and the recognition rate of the proposed system. For evaluation of the recognition rate the text writing application was created. During this experiment, the average time needed to write



agne@ifko.ktu.lt

Fig. 12. The example of the text writing application.

the current word was considered. The grid environment with the numbered gaze regions was developed in the second experiment for determination of the proposed system accuracy, regarding that each action was limited in time.

During a week ten participants of different gander were participated in the experiments. At the beginning of each experiment, every participant has calibrated an eye tracking system for him self in order to properly operate the proposed application. Four calibration points were observed for a certain time. The 20 frames (about 1.5 s) were additionally taken to evaluate the accumulative luminosity function for closed eye detection. Whole calibration process lasted about 15 s in this case. Each participant was asked to perform both experiments in three different modes: “fixed” – keeping the head in fixed position; “not fixed” – constantly moving the head; “with compensation” – moving the head with turned on compensation function.

4.1. *Testing of System Control Speed*

The scientific research in the area of gaze tracking systems has shown that it is reasonable to use the text writing application for system testing (Aoki *et al.*, 2009). A similar application has been developed in our case as well. The participants, using the text writing application, were asked to write a word “EYETRACKER”. The classification rate and the time needed for establishing the task, was recorded during the experiment. The created writing application consists of three regions, i.e., regions of buttons, selected text and gaze tracking state (Fig. 12). The zooming functionality is used in this case, which expands the letters of the button when it is pushed. The mouse clicking operation is executed when user’s eye is closed. The application allows selecting one letter at a time, using only two mouse clicks. The selected letter is displayed in the text region. The recalibration procedure of the system is voluntary and freely executed; when user feels that system is lacking the accuracy.

The experimental performance results are depicted in Fig. 13. The best score can be achieved using the algorithm with the head movement compensation function, where the average time that is needed to write the word “EYETRACKER” is equal to 74.33 s, which

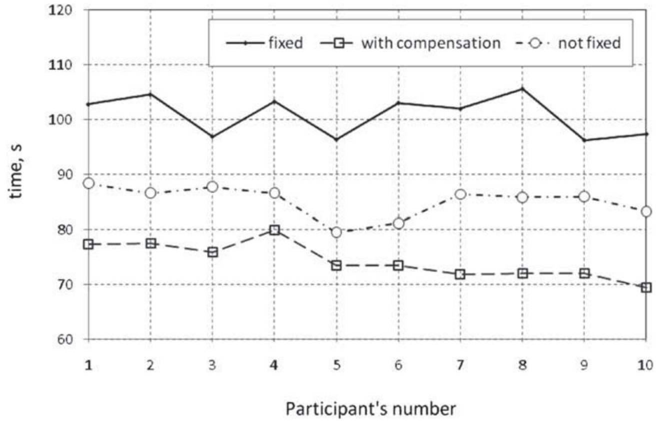


Fig. 13. The results of experiments with text writing application.

is 14% faster than "not fixed" mode and 35% faster than "fixed". About 20 commands per second are generated using proposed gaze tracking algorithm, because the algorithm is able to discriminate the processes of the selection and gazing. The tendency is noticed, that the users learn to use the eye tracking system more efficiently so the time needed for accomplishing the task has been shortened.

4.2. *Testing of System Control Accuracy*

The proposed eye detection system was developed using mini pinhole camera of resolution 640×480 at a frame of 20 fps. It has an IR-sensitive CCD chip and was modified with an IR filter. The system is implemented in MATLAB programming environment using image processing toolbox. In experiments, different grids of $5 \times 5,\ 10 \times 10,\ 15 \times 15$ and 20×20 were used and each was displayed on the screen so that the system has a resolution of 25, 100, 225 and 400 gaze regions accordingly.

To evaluate the accuracy of the system, a user study was performed with ten participants. The participants were located in front of the computer screen with resolution of 1400×900 pixels and asked to look at the five randomly selected gaze regions (darker regions in Fig. 14). The gaze point was recorded for $\sim$5 s (100 frames) for each chosen rectangle.

During the experiments the right hits are counted and used as a system evaluation parameter. The hit is assumed as correct, if the participant looks at the certain gaze region, directed by the application. The right hits dependency on the scale of the gaze region (grid) and the participant's head poise is presented in Fig. 15. The results show that the highest accuracy (98%) is obtained using the algorithm which includes the head movement compensation function. It can be seen, that using any of the three algorithms, the accuracy is sensitive to the decreasing grid size, and it is almost impossible (less than 8%) to make correct hits in the 20×20 gird gaze region.

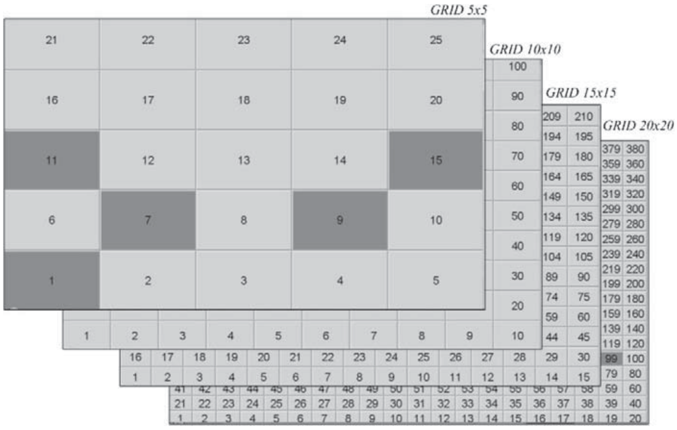


Fig. 14. The evaluation application of gaze tracking algorithm.

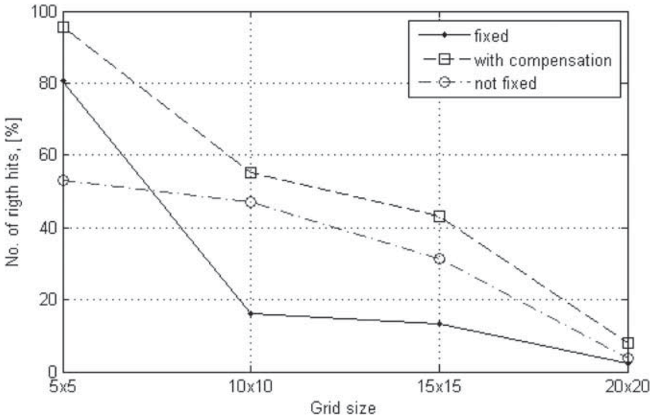


Fig. 15. The functional relationship between number of right hits and grid size.

5. Conclusions and Remarks

This paper proposes the new approach for the gaze tracking system involving the adapting thresholding value of the gray level, the normalized accumulative luminosity function and the algorithm of natural head motion compensation is proposed. The algorithm of pupil detection based on adaptive gray level threshold was created. The best tacking performance is reached, when rational default threshold value is selected from the range of $[80, \dots, 200]$ pixels intensively. The adaptive functionality of the proposed algorithm decreases the sensitivity of the system to the condition of constantly varying ambient lightness.

The correlation criterion is chosen for the detection of the closed eye, because it has clear variation limits and it is robust to the scale of the luminosity function. The highest accuracy of closed eye detection algorithm is reached when the correlation threshold Θ_C

is in the range of $[0.8, \ldots, 0.95]$. The normalized accumulative luminosity function can be used as additional signal for fixation of voluntary selected command.

The usage of non-linear compensation algorithm allows us to get efficient gaze tracking when the head orientation changes in three different directions. The testing of three different neural network's structures has shown that the type of transfer function in a hidden layer has insignificant influence to the accuracy and the higher accuracy is obtained using 10 neurons in a hidden layer.

The first experiment has shown that the presented eye tracking system can be successfully applied to the text writing purposes and the shortest time of writing can be achieved using the head movement compensation function. The second experiment has shown that the best results of accuracy can be reached using the same function as well. However it confirms that the proposed system allows accurate tracking in the small-scale grid of gaze regions (98% accuracy in 5×5 grid, 57% accuracy in 10×10 grid and less). According to such results we can conclude that people with physical impairments have possibilities to control dedicated applications (for typing text, e-mail reading) based on the proposed gaze tracking system.

6. Future Work

The performance and efficiency of the gaze tracking system can be increased using the combination of many signals, such as measurements of brain activity (EEG), measurements of muscle activity (EMG), measurements of 3D head orientation, voice or gesture recognition and evaluation of gaze direction. The method that combines all these signals must solve the data fusion task, because many of these signals are activated doing the same mental or physical action (task). Therefore, future research will involve the creation of sophisticated and robust methods that should make proper control decision despite the fact that some signals can be more or less erroneous than others in different cases.

The main disadvantage of the proposed eye tracking system is a lack of control resolution, i.e., the eye position is recognized in certain discrete position. Therefore, specially designed applications must be used instead of common Windows based applications (Internet explorer and etc.). Future work should involve the design of continuous (or analog) gaze tracking system, which enables smooth mouse cursor control. Such gaze tracking system must interpolate the pupil position between calibration points and associate this position with cursor. The interpolation can be executed using relatively simple polynomial equations or the computational intelligence, i.e., artificial neural networks or fuzzy logic.

References

Aoki, H., Hansen, J.P., Itoh, K. (2009). Learning gaze typing: what are the obstacles and, what progress to expect? *Universal Access in the Information Society*, 8(4), 297–310.

Cihan, T., Atakan, D., Omer, N.G. (2008). A wearable head-mounted sensor-based apparatus for eye tracking applications. In: *IEEE International Conference on Virtual Environments, Human–Computer Interfaces, and Measurement Systems*, pp. 136–139.

Dongheng, L., Winfield, D., Parkhurst, D.J. (2005). Starburst: a hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 79–99.

Dongheng, L., Babcock, J., Parkhurst, D.J. (2006). OpenEyes: a low-cost head-mounted eye-tracking solution. In: *Proceeedings of the 2006 Symposium on Eye Tracking Research and Applications (ETRA'06)*, pp. 95–100.

Dzemydiene, D., Bielskis, A.A., Andziulis, A., Drungilas, D., Gricius, G. (2010). Recognition of human emotions in reasoning algorithms of wheelchair type robots. *Informatica*, 21(4), 521–532.

Fabio, L.C. (2008). Class I infrared eye blinking detector. *Journal of Sensors and Actuators*, A148, 388–394.

Grigaitis, D., Bartkute, V., Sakalauskas, L. (2007). An optimization of system for automatic recognition of ischemic stroke areas in computed tomography images. *Informatica*, 18(4), 603–614.

Istance, H., Bates, R., Hyrskykari, A., Vickers, S. (2008). Snap clutch, a moded approach to solving the Midas touch problem. In: *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications (ETRA '08)*, pp. 221–228.

Kumar, N., Kohlbecker, S., Schneider, E. (2009). A novel approach to video-based pupil tracking. In: *IEEE International Conference on Systems, Man and Cybernetics (SMC'09)*, pp. 1255–1262.

Langner, F., Knauer, C., Jans, W., Ebert, A. (2009). Side scan sonar image resolution and automatic object detection, classification and identification. In: *OCEANS 2009-EUROPE*, pp. 1–8.

Li, F., Kolakowski, S., Pelz, J. (2007). Using structured illumination to enhance video-based eye tracking. In: *IEEE International Conference on Image Processing (ICIP)*, pp. 373–376.

Long, X., Tonguz, O., Kiderman, A. (2007). A high speed eye tracking system with robust pupil center estimation algorithm. In: *Annual International Conference of the Engineering in Medicine and Biology Society*, pp. 3331–3334.

Nonaka, H. (2003). Communication interface with eye-gaze and head gesture using successive dp matching and fuzzy inference. *Journal of Intelligent Information Systems*, 21(2), 105–112.

Pivrikas, V. (2006). *Vestibulometrija*. S. Jokuzio Publishing House, Klaipeda (in Lithuanian).

Shigeoka, T., Ninomiya, T., Muro, T., Miki, N. (2008). Wearable pupil position detection system utilizing dye-sensitized photovoltaic devices. *Sensors and Actuators*, A145–146, 103–108.

Sugita, T., Suzuki, S., Kolodko, J., Igarashi, H. (2007). Development of head-mounted eye tracking system achieving environmental recognition ability. In: *Annual Conference of Society of Instrument and Control Engineers (SICE)* , pp. 1887–1891.

Topal, C., Dogan, A., Gerek, O.N. (2008). A wearable head-mounted sensor-based apparatus for eye tracking applications. In: *IEEE Conference of Virtual Environments, Human–Computer Interfaces and Measurement Systems*, pp. 136–139.

Van der Heide, A., Trivino, G. (2010). Simulating emotional personality in human computer interfaces. In: *IEEE International Conference of Fuzzy Systems (FUZZ)*, pp. 1–7.

Wu, J., Ruan, Q., An, G. (2010). Exemplar-based image completion model employing PDE corrections. *Informatica*, 21(2), 259–276.

Yu, L.H., Eizenman, M. (2004). A new methodology for determining point-of-gaze in head-mounted eye tracking systems. *IEEE Transactions on Biomedical Engineering*, 51(10), 1765–1773.

V. Raudonis, is a doctor of informatics engineering at the Faculty of Electrical and Control Engineering from Kaunas University of Technology, Lithuania. He obtained masters degree in electrical and control technologies. His research interest is application of computation intelligence in human–computer interaction, computer vision, assistive technology and robotics. Together with colleagues from Lithuania he is participating in various research projects with industrial companies like Siemens, Beijer Electronics.

A. Paulauskaitė-Tarasevičienė is a doctor of informatics sciences, a lecturer in the Department of Business Informatics at Kaunas University of Technology. The field of research is formal specification, validation and simulation of dynamic structure adaptive systems, real time systems and theory of algorithms.

L. Kižauskienė is a doctor of informatics engineering and a lecturer at the Department of Computer Engineering in Kaunas University of Technology, Lithuania. The field of research is agent-based technology, embedded real-time systems, artificial intelligence, smart devices in ubiquitous computing environments.

Žvilgsnio sekimo programa, įvertinanti natūralius žmogaus galvos judesius

Vidas RAUDONIS, Agnė PAULAUSKAITĖ-TARASEVIČIENĖ, Laura KIŽAUSKIENĖ

Šiame darbe yra pristatyta žvilgsnio sekimo sistema, kuri automatiškai aptinka vyzdžio padėtį dvimatėje erdvėje, panaudojant adaptyviai besikeičiantį pilkumo slenkstį. Sistemoje užmerktos akies aptikimas yra realizuojamas panaudojant normalizuotą suminę šviesio funkciją, kuri leidžia užfiksuoti valingai pasirinktą komandą. Sistema taip pat leidžia atlikti sistemos perkalibravimą, kuris yra atliekamas be asistuojančio žmogaus pagalbos. Natūralūs galvos judesiai sistemoje yra įvertinami, panaudojant netiesinį klasifikatorių ir orientacijos jutiklius. Sistemos greitaveika ir tikslumas prie tam tikrų vartotojo elgsenos variacijų buvo išbandyti atliekant du eksperimentinius tyrimus.