A Low-Computational Approach on Gaze Estimation with Eye Touch System

Cihan Topal, Serkan Gunal, Onur Koçdeviren, Atakan Doğan, and Ömer Nezih Gerek

Abstract—Among various approaches to eye tracking systems, light-reflection based systems with non-imaging sensors, e.g., photodiodes or phototransistors are known to have relatively low complexity; yet, they provide moderately accurate estimation of the point of gaze. In this study, a low-computational approach on gaze estimation is proposed using Eye Touch system, which is a light-reflection based eye tracking system, previously introduced by the authors. Based on the physical implementation of Eye Touch, the sensor measurements are now utilized in low-computational least-squares algorithms to estimate arbitrary gaze directions, unlike the existing light reflection based systems, including the initial Eye Touch implementation, where only limited predefined regions were distinguished. The system also utilizes an effective pattern classification algorithm to be able to perform left, right and double clicks based on respective eye winks with significantly high accuracy. In order to avoid accuracy problems for sensitive sensor biasing hardware, a robust custom microcontroller-based data acquisition system is developed. Consequently, the physical size and cost of the overall Eye Touch system are considerably reduced while the power efficiency is improved. The results of the experimental analysis over numerous subjects clearly indicate that the proposed eye tracking system can classify eye winks with 98% accuracy, and attain an accurate gaze direction with an average angular error of about 0.93°. Due to its lightweight structure, competitive accuracy and low-computational requirements relative to video-based eye tracking systems, the proposed system is a promising human-computer interface for both stationary and mobile eye tracking applications.

Index Terms—Eye tracking, gaze estimation, human-computer interface, assistive technology.

I. INTRODUCTION

Eye tracking has become a key technology due to its potential in several applications, ranging from human-computer interface systems for people with and without disabilities to diagnosis of physiological, neurological and ophthalmologic problems in individuals, and mobile systems such as interfaces for wearable computers. In the literature, a variety of eye tracking systems have been proposed, which can be classified based on their physical structure or working style. In terms of the physical structure, they are grouped into head mounted systems or remote systems. Other categorizations could be due to being wearable or non-wearable, and being infrared-based or appearance-based. In terms of the working style, there are video-based eye tracking systems or systems that use other approaches (electro-oculography and photo-oculography) [1], [2].

Eye Touch is one of the light-reflection-based eye tracking systems, which is implemented by placing IR LEDs and IR sensors around an eyeglasses frame. The first experimental results of this tracking system were presented in [3]. Later on, a more efficient prototype was constructed and tested via simple sensor measurements in [4]. In both studies, Eye Touch was able to coarsely separate gaze directions on a grid of 3 × 4 on the screen. In other words, gaze positions were computed in a discrete manner rather than computing an actual gaze vector. Besides, the hardware of the tracking system was including a professional analog to digital converter, which keeps the cost of the overall system unaffordable.

In this paper, we further improved upon Eye Touch system so that a low-computational approach on gaze estimation with continuous resolution is achieved. Specifically, the following contributions can be listed with respect to our previous work [3], [4]:

○ The sensor measurements are utilized in low-computational least-squares based algorithms to estimate the point of gaze (PoG). Thus, continuous resolution is attained, which enables Eye Touch to point anywhere on the screen rather than a predefined grid.

○ The system is capable of performing left, right and double clicks using appropriate eye winks with the help of a state-of-the-art SVM classifier, rather than a simple nearest neighbor classification algorithm. In this way, accuracy and precision of the classification is improved.

○ A custom microcontroller-based data acquisition system is developed to accurately sample and digitize the analog sensor outputs. As a result, the physical size and the cost of Eye Touch system is considerably reduced.

○ An analysis is conducted to prove that Eye Touch is eye safe as far as IEC 825-1 standard is concerned.

The results of the experimental analysis over an extended set of subjects indicate that the proposed tracking system accurately classifies user events, and computes the point of gaze with a considerably high accuracy while requiring low-computational power and low-cost, as well.
works by detecting the current variations in the coil kindly mounted on a contact lens. When a special hardware that produces magnetic field is placed around the user, even very little saccades of the eye causes a sensible amount of current variations on the search coil [10]. This technique may not be suitable for portable applications, though it has evolved as one of the most precise eye movement measurement systems. There are studies about enhancing the compactness and accuracy of this system. In [11], in order to make the scleral search coil technique available for compact and portable applications, a modified version of the system that consists of a wireless search coil is proposed. In [12] and [13], the efficiency of the system is addressed. Furthermore, there is a more recent study in which a very compact planar transmitter for scleral search coil eye tracking systems is introduced [14].

Besides above approaches, there are also light-reflection based approaches, which employ non-imaging sensors such as photodiodes or phototransistors. With the employment of non-imaging sensors in the eye tracker design, considerable amount of reductions on the acquired data and hence on the required computational power can be obtained. These approaches simply illuminate the eye vicinity and collect the reflected light illumination with the carefully placed sensors around the eye. Thus, they obtain a one-dimensional vector including the information about the position of the eye. Note that imaging sensors employed in video-based methods acquire two-dimensional and larger amount of data. The method proposed herein can be considered in this class of eye trackers.

In [15], several differential reflection principles are described. Light-reflection based methods have been used in both research and commercial systems [16]-[18]. There is also a very sophisticated version of light-reflection based eye trackers, which illuminates the eye surface by employing a flying-spot laser as a substitute for light source [19]. Another system, the Owl, was originally developed in the 1980’s and renovated around 2005 [20]. It consists of a ring of LED emitters and photodiode sensors. The LEDs are triggered in a sequential order and the photodiodes measure the LED stimulation. Thus, coarse grain information about the position of the limbus can be extracted from the low dimensional data acquired, which can be processed in a very lightweight hardware. The prominent feature of Owl is that it uses the LEDs for feedback as well. However, the system has also a disadvantage, i.e., it is mounted right in front of the eye and substantially occludes the user’s sight. As its developers described, Owl is not suited for general gaze tracking, but it would serve better as a selection-driven tracker [20].

Among the other approaches, parallel to the technological developments in imaging devices, the most crowded subset of eye tracking literature formed around the video-based approaches, i.e. video-oculography (VOG) technique. Video-based eye tracking systems use one or more camera sensors (CCD or CMOS) in order to get the 2D image of eye whereas the light reflection methods acquire 1D data which is much more simpler to process. Typically,
computationally intensive image processing and pattern recognition methodologies are applied on this image to extract the related features from the eye image, i.e. pupil center, corneal reflection locations, etc [21]-[43]. Although, this type of eye trackers has become the default solution by achieving fairly high performance in terms of accuracy, they have drawbacks if low-complexity, hence low-power systems are desired.

In video-based eye tracking systems, the number of cameras for sampling the eye vicinity varies according to the purpose or the accuracy of the system. In some systems, monitoring one of the eyes may be sufficient, whereas in some other systems, both eyes may need to be tracked [21], [23], [40]-[43].

Likewise, camera resolutions and camera sampling frequencies are also important parameters that concern the efficiency and time-complexity of video-based eye tracking systems. In order to reach better accuracy levels, spatio-temporally higher resolution cameras might be preferred, necessitating additional hardware to meet the real-time processing requirements. Similarly, acquisition or conversion of the video signal to digital signal may also require its own authentic circuitry [22].

There are also remote video-based eye tracking systems which involve more sophisticated imaging hardware in order to provide high accuracy when operating far from the user. Although, for many users, remote video-based eye trackers are easier to use than intrusive systems, they require the user to keep his/her head in a particular area. Most of these video-based eye tracking systems benefit from differentiating bright and dark pupil images to detect pupil [23]-[25], [39], [42] and use Purkinje reflection [26], [40] to estimate the point of gaze. Detection of locations of pupil and glints enables to use mathematical models based on 3D geometry in order to estimate the gaze vector of the user [27], [41]-[43]. Besides the advantages, these systems also require explicit 3D calibrations of the employed cameras and light sources.

Apart from the computational and hardware problems of video-based eye tracking systems, they may also cause integration difficulties in specific applications such as mobile applications due to the problems including mounting the camera in an appropriate way and placing hardware capable of providing required computation power. With all pros and cons, video-based eye tracking systems offer the most competitive solutions for the era, hence, we present a more detailed comparison of Eye Touch system with the video-based systems in terms of accuracy and head pose invariance in Section III-D.

III. EYE TOUCH SYSTEM

Eye Touch system, as shown in Fig. 1, is composed of the following components: Eye Touch goggles, a data acquisition device, Eye Touch software which runs on a regular PC, and an external power supply. Each of these components will be introduced shortly.

Eye Touch system is designed to be capable of discovering eye clicking gestures and eye gaze direction in real-time. This capability is due to the following simple facts:

- Human eye has two main parts with respect to the color intensity distribution: sclera and iris-pupil. Whilst the sclera has mostly white color, the iris-pupil circle consists of darker tones such as brown, black, etc. This color distribution is fixed in all humans in infrared (IR) range.
- Different colors reflect the light in different amounts and around different wavelengths.
Considering these basic light reflection principles and the spherical structure of eye surface, it is possible to acquire the reflected light intensities from specific portions of the eye surface and then find out the current clicking gesture or eye gaze direction based on the measurements of reflected light intensities.

Eye Touch system provides only a relative eye-gaze direction with respect to the absolute head direction. Therefore, for the system to constitute an overall computer interface, a chin-rest must be used to avoid head movements.

### A. Eye Touch Goggles

Eye Touch goggles are an eyeglasses frame without any lenses as shown in Fig. 2. In our previous experimentation stages, two infrared light sensitive goggle prototypes were developed. The first prototype was reported in [3] and the second one in [4]. In this study, because of its proven higher performance, the second prototype has been used.

In order to detect the iris movements, the goggles are equipped with twelve infrared LEDs that illuminate both eyes and reduce the risk of ambient light to deteriorate the system performance, and with twelve infrared sensitive phototransistors that produce voltage values with respect to the collected amount of light in infrared spectrum. As it can be seen from Fig. 2, both left and right frame portions include six LEDs and sensors. It must be noted that the number of sensors (phototransistors) is a design parameter. There is a substantial relationship between the number of sensors and the system accuracy, which will be shown in Sec. VI. Furthermore, the sensors are surrounded by opaque cylindrical plastic covers, which eventually causes the sensors to depict light reflection from specific portions of the eye.

As far as the number of LEDs is concerned, it can be different than the number of sensors. Since the main purpose of having LEDs on the goggles is illuminating the eye vicinity homogeneously at a convenient brightness level, several of them are placed in order to prevent the illumination variances. In Appendix A we present useful information about the eye safety issues of infrared light emitted by the goggles.

The IR sensitive sensors used on the goggles are light-to-voltage optical sensors. These sensors respond to the light in 800 - 1100 nm wavelength range with a sharp peak at 940 nm. Since the IR LEDs used emit light at the wavelength of about 940 nm, the pairs match well.

Eye Touch goggles are connected to the data acquisition device over a parallel communication cable. This cable carries twelve channel analog voltage signals to the device as well as the power from the power supply for the sensors and LEDs. In its current form, the goggles weigh about 50 grams.

### B. Data Acquisition Device

Data Acquisition Device (DAQ), as shown in Fig. 3, is a custom-made microcontroller based embedded system. DAQ has been developed to be a bridge between the goggles, power supply and PC software (Fig. 1). We can summarize the DAQ’s main tasks as follows:

- Supplying regulated power to the goggles to enable sensitive measurements against noise.
- Collecting analog sensor measurements from the goggles and digitizing them to generate a feature vector.
- Sending feature vectors to our classification and gaze estimation software running on a PC.

One of the most important efficiency metrics for eye tracking systems is the operating rate that is defined to be the gaze vector computation rate per second. Operating rate of a system, on the other hand, mostly depends on its communication and computation requirements. The data communication requirements of the system is discussed herein, whereas computational requirements are discussed in Sec. V.

A video-oculography based eye tracking system typically uses a digital camera connected over a USB port to a PC. The maximum transmission speed of USB 2.0 is 480 Mbits/s or 60 MBytes/s (note that the actual bandwidth is lower in practice). The maximum theoretical operating rate of a video-oculography based eye tracking system can be computed based on this bandwidth value. Let’s consider a standard VGA camera with an RGB sensor that can acquire $640 \times 480$ frames. Each VGA frame consumes $640 \times 480 \times 3 \cong 0.9$ MBytes. As a result, the maximum operating rate of such a system can be found as $60 \div 0.9 = 66.7$ Hz. Considering the possibility of rapid eye movements for humans, this rate may impose limitations.

As compared to the video-oculography based systems, Eye Touch can reach higher operating rates, thanks to its small feature vector size. Recall from Section III-A that the goggles have twelve sensors. Let’s assume that each sensor output is converted by an ADC with a resolution of 12-bit (2 bytes). Then, each 12-dimensional feature vector consumes $12 \times 2 = 24$ bytes $\cong 0.024$ MByte. Therefore, the maximum operating rate becomes $60 \div 0.024 = 2.5$ KHz. This operating rate, when deployed, is well above what is typically required for an eye tracking system. Although Eye Touch system can reach high operating rates; we prefer to realize a simple DAQ prototype for experimental purposes to verify that the proposed approach is applicable. Our naive data acquisition device is capable of running in real-time with a sampling rate of 10 Hz. The cost of the goggles and the DAQ card is kept below $100 in total.

### C. Eye Touch Software

Eye Touch system is complemented by the software that runs on a PC. The software listens to DAQ device and handles some major tasks based on the information provided by the device. These tasks can be listed as the following:

- Providing a user interface for collecting data, which is shown in Fig. 4, and preprocessing the acquired data,
- Running the classification algorithm to determine the current action performed by the user (left click, right
invariance unless a relatively low accuracy (less than 1°) is acceptable. Both the head pose invariance and reasonable accuracy can be provided only if the amount of hardware is increased.

Considering the aforementioned comparisons, among EOG and light-reflection based eye tracking systems, Eye Touch is the only one that provides continuous resolution. As far as the head pose invariance is concerned, Eye Touch system cannot provide this feature unless an additional mechanism is employed. However, this drawback is also valid for most of the single camera video-based eye tracking systems [31]-[37]. Besides this fact, there are also many eye tracking applications in which the head pose invariance is not required. Wearable computers with head mounted displays (HMDs) and mobile eye tracking systems are examples of such applications. Considering the computational burden of video-based eye tracking systems, mobility would become a problem in terms of processing hardware and power efficiency. For mobile systems, estimating the point of gaze with a couple of simple sensors, as in the case of Eye Touch, becomes a useful solution.

Eye Touch is also able to detect click events (left, right and double clicks) accurately by utilizing a pattern classification method on the acquired sensor measurements. To our knowledge, the click operations based on the blink detection and their performances, e.g., accuracy and precision, have not been reported in the literature.

Consequently, Eye Touch combines the continuous resolution capability of video-based systems and low-computational framework of EOG and light-reflection based systems in a lightweight design. Finally, we can list the contributions that this study puts on top of the previous work as following:

- Unlike EOG and other light-reflection based systems, Eye Touch provides continuous resolution in gaze estimation.

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Head Pose</th>
<th>Accuracy (°)</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>12 photodiodes</td>
<td>×</td>
<td>0.9</td>
<td>Eye Touch System</td>
</tr>
<tr>
<td>1 camera</td>
<td>×</td>
<td>2 - 4</td>
<td>[31], [32]</td>
</tr>
<tr>
<td>1 camera</td>
<td>×</td>
<td>1 - 2</td>
<td>[33], [34], [35]</td>
</tr>
<tr>
<td>1 camera</td>
<td>×</td>
<td>0.5 - 1.5</td>
<td>[36], [37]</td>
</tr>
<tr>
<td>1 camera</td>
<td>√</td>
<td>1 - 3</td>
<td>[38], [39]</td>
</tr>
<tr>
<td>2 cameras</td>
<td>√</td>
<td>3</td>
<td>[40]</td>
</tr>
<tr>
<td>2 cameras</td>
<td>√</td>
<td>&lt; 1 - 2</td>
<td>[41]</td>
</tr>
<tr>
<td>3 cameras</td>
<td>√</td>
<td>0.7 - 1</td>
<td>[42]</td>
</tr>
<tr>
<td>4 cameras</td>
<td>√</td>
<td>0.6</td>
<td>[43]</td>
</tr>
</tbody>
</table>

Fig. 4. Graphical user interface and 12 gaze calibration points of Eye Touch software.
Fig. 5. Clusters of sensor voltage readings (V) belonging to 4 main events, i.e., gaze, left, right and double clicks.

- It offers a competitive gaze estimation performance as compared to video-based systems.
- It enables left, right and double click actions through highly accurate and precise blink detection.
- It provides a low-computational framework for user event detection and gaze estimation.

IV. SYSTEM TRAINING AND CALIBRATION

Before practically using Eye Touch system, the system must be prepared with respect to its current user by means of the system training and calibration which consists of three phases:

i. Data collection and preprocessing.
ii. Training for gaze/click classification.
iii. Calibration.

Each of these phases is elaborated below.

A. Data Collection and Preprocessing

System training and calibration starts with the data collection for which Eye Touch user interface in Fig. 4 is used. During the data collection, a user must steadily face the monitor (by a chin rest) and perform gazing and blink operations as commanded by on-screen directions. Specifically, user is first asked to sequentially look at sampling (training/calibration) points numbered from one to twelve that form a $3 \times 4$ grid on the screen. Finally, user is asked to perform left blink, right blink, and double blink gestures, all at a sufficiently long duration (about 2 seconds). Consequently, overall data collection phase lasts for about 30 seconds.

While the data collection is in progress, the goggles provide ten sensor readings for each sampling point or blink gesture, each of which is a 12-dimensional vectors composed of voltage values. In order to minimize the risk of the system being susceptible to some unintentional eye moves (unintentional blinks and/or unintentional gaze distractions), filtering the acquired data is required. Thus, during the preprocessing, we filter out the outlier vectors among these ten readings based on the following simple method: (i) We first compute the centroid of ten-vector cluster in 12-dimensional space. (ii) We eliminate four of them that lie farthest in Euclidean sense from the centroid, which leaves six readings for each sampling point to be used in the upcoming training and calibration phases. Effect of the outlier removal operation on performance of Eye Touch is empirically verified. The quantitative results of this verification is reported in Sec. VI.

B. Training for Gaze / Click Classification

The possibility of distinguishing click operations from normal gazing activity arises from the observation of different locus of feature data (readings from the sensors) in either case. It was visually observed in 3-dimensional space that the readings belonging to various sensor combinations form separate clusters for four different user events. The readings for one of these combinations are illustrated in Fig. 5. Considering this behavior of the sensor readings, the user events can be separated using appropriate hyper-planes or hyper-surfaces provided by a suitable pattern classification algorithm. Therefore, Eye Touch makes use of a classification algorithm to find out the current user event, which can be one of left click, right click, double click, or gaze.

Due to its popularity and proven performance, Support Vector Machine (SVM) classification algorithm is adopted for this work, with the selection of linear kernel [44]. It must be noted that this selection is just a design choice, and it does not affect the idea and the performance critically. During the training phase of SVM, the abovementioned preprocessed data is utilized. Once the training is over, the computed support vectors providing the maximum margins among four classes are stored to be used for the event classification in real-time operation.

C. Calibration

Once the classification algorithm determines that the current user event is gaze, Eye Touch uses a gaze estimation algorithm to further compute new cursor position in real-time. On the other hand, the estimation algorithm is built
upon linear mapping or nonlinear mapping, both of which require two mapping coefficient vectors to set forth the
cursor position. These vectors must be computed (or calibrated) beforehand for the real-time operation of the gaze
estimation algorithm. Thus, in this section, we first present
our motivation for using a linear/nonlinear mapping and
explain how the calibration is performed for each mapping
mechanism based on the preprocessed data.

In order to explain the rationale behind the proposed
gaze estimation approach, the sensor voltage readings from
three sensors (for the sake of 3D plot visualization) are illustrated for twelve different calibration points in Fig. 6.
The figure indicates that the clusters formed by the sensor
voltage readings do not not arbitrarily lie on the sensor output vector space. On the contrary, the distribution sur-
prisingly follows a pattern, revealing the possibility of a
geometric model that can be obtained through line and
curve fitting methods. Furthermore, it is interesting to note
that the clusters are formed in 3D space with respect to the
locations of calibration points in the screen coordinate
system (Fig. 4). Consequently, it is possible to estimate gaze directions from vector of sensor readings that may
correspond to arbitrary points among the training points
by means of a mathematic estimation method, i.e. least
squares estimation (LSE).

The eminent class-layout pattern on the sensor output
vector space could be explained by the physical shape
of an eyeball and the locations of sensors. The reflected
light from the eye ball is higher in white regions as compared
to that from the iris, and this difference almost linearly changes as the iris moves towards (or away from)
a particular sensor, corresponding to left, right, up or down
eye movements.

Since Eye Touch software knows the locations of cal-
ibration points in the screen coordinate system, and there is
a class-layout pattern on the sensor output vector space,
it is possible to construct a mapping from the sensor
voltage readings to the known gaze points. For this purpose,
two different mapping strategies (linear and nonlinear) are
tested in this work. The mapping coefficients (linear or
nonlinear) are calibrated during this phase, as explained
in the following sections.

1) Linear Mapping: Suppose that user is currently look-
ing at the coordinate $(x_{screen}, y_{screen})$ of the screen. These coordinate axes can be expressed as:

$$x_{screen} = c_x^T \cdot \vartheta = c_{x0} + c_{x1} f_1 + c_{x2} f_2 + \cdots + c_{x12} f_{12} \quad (1)$$
$$y_{screen} = c_y^T \cdot \vartheta = c_{y0} + c_{y1} f_1 + c_{y2} f_2 + \cdots + c_{y12} f_{12}. \quad (2)$$

where $c_x = [c_{x0}, c_{x1}, \ldots, c_{x12}]^T$ and $c_y = [c_{y0}, c_{y1}, \ldots, c_{y12}]^T$
denote the coefficient vectors for $x$ and $y$ coordinates, and
$\vartheta = [1 f_1 f_2 \ldots f_{12}]^T$ is the feature vector corresponding
to the sensor readings.

It was previously explained that during the data col-
clection phase, ten sensor readings, each of which is a
12-dimensional vector of voltage values, are made for
each calibration point. In order to eliminate unintentional
eye winks, out of these ten sensor readings, four outliers
are eliminated during the preprocessing phase, leaving six
readings to be used in the calibration of linear mapping.
For the simplicity of mathematical notations, consider $F_i$
as the matrix composed of the six filtered vectors that are
acquired when the user is gazing at $i^{th}$ calibration point
during the calibration:

$$F_i = \begin{bmatrix}
    f_{i, 1} & f_{i, 2} & \cdots & f_{i, 12} \\
    f_{i, 1} & f_{i, 2} & \cdots & f_{i, 12} \\
    \vdots & \vdots & \ddots & \vdots \\
    f_{i, 0} & f_{i, 2} & \cdots & f_{i, 12}
\end{bmatrix}_{6 \times 12} \quad (3)$$

Considering that there are 12 calibration points, the set of
equations for the $x$ coordinates of these calibrations points
can be represented as in Eq. 4:

$$\begin{bmatrix}
    x_1 \\
    x_2 \\
    \vdots \\
    x_{12}
\end{bmatrix} = \begin{bmatrix}
    c_{x0} \\
    c_{x1} \\
    \vdots \\
    c_{x12}
\end{bmatrix}_{12 \times 1} \begin{bmatrix}
    F_{72 \times 13} \\
    F_{72 \times 12}
\end{bmatrix}$$

where $\mathbf{1}_n$ denotes $n \times 1$ vector of unites, and $x_i = x_i \cdot \mathbf{1}_6$, where $x_i$ is the $x$ coordinate of the $i^{th}$
calibration point. Similarly, the set of equations for the $y$ coordinates of these calibrations points can be denoted as:

$$\mathbf{F}_{72 \times 13} \times c_y_{13 \times 1} = y_{72 \times 1} \quad (5)$$

The unknown set is composed of the vector of coeffi-
cients, $c_x$, which is of size 13. Since this value is less
than the number of equations, which is 72, a least squares
solution strategy based on singular value decomposition
(SVD) [45] is adopted. Finally, a vector ($c_y$) that ap-
proaches the above set of equations in the minimum squared
error sense is evaluated. Following the similar strategy for the y-axis, \( c_y \) vector is also computed, wrapping up the calibration phase for the linear mapping approach.

2) Nonlinear Mapping: The plot presented in Fig. 6 contains clues of a relation between the gaze direction and the sensor readings. It is, however, difficult to claim a perfectly linear relation. In order to test the possibility of a performance improvement, a nonlinear approach is tested. Obviously, more complicated nonlinearities than the ones in Eq. 6 and 7 could also be tested. However, the idea of the paper is to introduce the sensor based apparatus and its possible application to gaze direction estimation, so other (infinitely many) nonlinearities were left beyond the scope of this work.

In the nonlinear mapping adopted herein, \((x_{\text{screen}}, y_{\text{screen}})\) is denoted as:

\[
x_{\text{screen}} = c_{x_0} + c_{x_1} x_1 + c_{x_2} x_2 + \ldots + c_{x_{12}} x_{12} + c_{x_{13}} x_1^2 + c_{x_{14}} x_2^2 + \ldots + c_{x_{24}} x_{12}^2 \tag{6}
\]

\[
y_{\text{screen}} = c_{y_0} + c_{y_1} y_1 + c_{y_2} y_2 + \ldots + c_{y_{12}} y_{12} + c_{y_{13}} y_1^2 + c_{y_{14}} y_2^2 + \ldots + c_{y_{24}} y_{12}^2 \tag{7}
\]

Note that the number of parameters is doubled (from 12 to 24) in order to achieve coefficients of second order polynomials. Consequently, the set of equations for the \( x \) coordinates of twelve calibration points can be represented as follows:

\[
\begin{bmatrix}
16 
\begin{bmatrix}
F_1 
& F_1 F_1 
& F_1 & 1 
& F_2 
& F_2 F_2 
& F_2 & 1 
& \vdots 
& \vdots 
& \vdots 
& \vdots 
& 16 
\end{bmatrix}
& 12 
\end{bmatrix}
\begin{bmatrix}
c_{x_0} 
\vdots 
c_{x_{12}} 
\end{bmatrix}
= 
\begin{bmatrix}
x_1 
\vdots 
x_{12} 
\end{bmatrix}
\tag{8}
\]

where \( X, Y \) denotes the matrix whose entries \([X, Y]_{ij} = X_{ij}, Y_{ij}\). In a similar manner, the set of equations for their \( y \) coordinates become:

\[
F_{72 \times 25} \times c_{y_{25 \times 1}} = y_{72 \times 1} \tag{9}
\]

In a similar manner, we solve the equation system of nonlinear mapping based on SVD [45] method. As a result, similar to the linear mapping, the calibration is over once the two coefficient vectors are computed.

V. EVENT DETECTION AND GAZE ESTIMATION

After the system training and calibration is over, Eye Touch is ready for finding out eye clicking gestures and eye gaze direction in real-time. The real-time operation of Eye Touch is shown in Fig. 7.

According to this figure, our classification algorithm (or classifier) trained using the preprocessed data is at the first stage in the real-time operation. The classifier makes a decision on the current user event as follows:

- A new 12D feature is acquired from twelve sensors on the goggles.
- SVM classification for the unknown feature vector is carried out using the support vectors, which are previously computed and stored in the training phase.
- Finally, the unknown feature vector is assigned to a class (left click, right click, double click, or gaze) based on the outcome of SVM classification.

Eye Touch immediately handles the classes corresponding to left, right, or double click events. The gaze class, however, needs further computation of the gaze direction, which is the second stage in the real-time operation as shown in Fig. 6. Specifically, the estimator computes \((x_{\text{screen}}, y_{\text{screen}})\) as follows:

- 12D feature is further applied to the estimator.
- The gaze direction estimation based on the linear and nonlinear mapping is accomplished by means of equations (1)-(2) and (5)-(6), respectively. Thus, a dot product of two vectors, namely the mapping coefficient vector \(c_x, c_y\) and feature vector, is required to estimate \(x_{\text{screen}}\) and \(y_{\text{screen}}\), respectively.

Eye Touch software performs the required action, whether it is a mouse click or moving the cursor to its new position.

VI. EXPERIMENTAL RESULTS

To assess efficacy of the proposed tracking system, data are acquired from 50 distinct users under different lighting conditions (L) varying between 15 and 167 lux as described in Sec. IV-A. Amongst those users, 39 are male and 11 are female; 45 users have dark eye color and 5 users have light eye color. Consequently, 50 different datasets are constituted with different lighting, gender, and eye color attributes. Fig. 8 shows the physical structure of the prepared testbed and a user who participated in the experimental study.

A. Classification Performance

As we noted in Sec. III-C, Eye Touch system consists of two main stages, classification and, if a click gesture is not the case, then gaze estimation. Therefore, overall accuracy of the system depends on the performance of both of these stages. In this section, we present the accuracy and precision values obtained by SVM classification in the first stage. These values are obtained by employing 80%

### TABLE II

<table>
<thead>
<tr>
<th>Method</th>
<th>Gender</th>
<th>Eye Color</th>
<th>Ambient Luminosity</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>Male</td>
<td>Female</td>
<td>Brown</td>
<td>Other</td>
</tr>
<tr>
<td></td>
<td>1.60</td>
<td>1.66</td>
<td>1.59</td>
<td>1.87</td>
</tr>
<tr>
<td>Nonlinear</td>
<td>0.93</td>
<td>0.96</td>
<td>0.94</td>
<td>1.10</td>
</tr>
</tbody>
</table>

Where: Lower values indicate better performance.
of the collected sensor data for the training phase, and the remaining 20% for the test phase of SVM classifier.

Considering all four events, the accuracy values are between 91% and 98% whereas the precision values range from 94% to 98% as illustrated in Fig. 9. Furthermore, near perfect accuracy and precision levels especially for the gazing establishes a firm ground for the gaze estimation algorithm.

B. Gaze Estimation Performance

Gaze estimation performance of Eye Touch system is evaluated by comparing the actual calibration points against the estimated gaze locations for all the users enrolled within the study. The results of gaze estimation with nonlinear mapping for ten distinct users are illustrated in Fig. 10. Additionally, the gaze estimation accuracy of Eye Touch is presented for all 50 users in Fig. 11 for both linear and nonlinear mapping approaches. As expected, nonlinear mapping offered more accurate results for almost all users. More specifically, we have reached 0.58° as the best average error result with nonlinear mapping. The average angular gaze direction errors for all users were 1.61° in linear mapping, and 0.93° in nonlinear mapping. It was experimentally verified that if the outlier elimination is not applied, average gaze estimation errors would increase to 2.18° and 1.88° for linear and nonlinear mapping strategies, respectively.

More detailed results with respect to gender, eye color, and ambient luminosity are also presented in Table II. One can note from this table that Eye Touch estimated the gaze direction with an average error as low as 0.93° for male users and 0.96° for female users. Considering the eye colors of the users, an error of 0.94° was attained for the brown eye color whereas the error was 1.10° for the other eye colors. Finally, for different ranges of ambient luminosity, the errors ranging from 0.85° to 0.97° were obtained.

All of the abovementioned accuracies are satisfactory enough to compare Eye Touch with modern video-based systems. In majority of the video-based eye tracking studies [31]-[43], the reported accuracies are varying from 0.5° to 4.0° (see Table I) in terms of the angular gaze estimation error for both remote and head mounted systems. Considering its fair accuracy and low computational requirements, Eye Touch therefore proves itself to constitute a promising candidate for both mobile and stationary human computer interface applications even under varying conditions such as lighting, eye color, and gender of the user.

C. Best Feature Set

The Eye Touch goggles were designed to efficiently capture light reflectance from various portions of the human eye. Nevertheless, the design is ad-hoc, and it remains to be an interesting issue to know which of the sensors carry more information. To analyse this, the feature vector elements are tested for classification and gaze angle accuracy in a combinational manner. Particularly, the sequential forward floating selection technique [46] is employed to determine the best feature set minimizing the gaze estimation error in each dataset. Thus, relevancy or redundancy of the sensors is obtained.

Table III presents the selected feature subsets for some of the users enrolled in the experiments. It can be noticed that the selected feature subset contains different features for different users. Moreover, each individual feature is present in at least two datasets; therefore, none of the sensors can be eliminated from the system. This is valid for all 50 users. Consequently, it is concluded that each sensor has a particular contribution to the accuracy for distinct users.

D. Computational Performance

Running times of the classification and gaze estimation stages were also measured to verify that Eye Touch is suitable for real-time operation. The measurements taken on a desktop computer equipped with Intel Core2Duo 2.2 GHz processor and 2 GB of RAM indicate the following: SVM classification is completed in 0.15 msec; linear and nonlinear mapping based gaze estimation require 0.011 msec and 0.018 msec, respectively. Considering the DAQ implementation of Eye Touch system, the classification and gaze estimation operations are carried out in real-time with considerably low computational requirements.

VII. CONCLUSIONS

In this study, Eye Touch system, which is based on portable and low cost components in a wearable form, is proposed and implemented for gaze estimation and blink gesture detection for click actions. The elimination of classical video camera and corresponding high computational cost can be considered as an advantage of the proposed system. Besides, this work gives an insight and plausible
results of a novel infrared oculographic approach to the detection of gaze direction and eye winks. The experimental results indicate that Eye Touch constitutes a promising user interface alternative in certain circumstances where hand control may be inconvenient.

The proposed eye tracking system offers an adequate accuracy even under varying conditions such as lighting, eye color, and gender of the user. The best accuracy result we obtained during the experiments is an average angular deviation of 0.58° for one user. The average gaze angular accuracy for all users is 0.93°. Considering the cost of the entire system being below $100, Eye Touch is a promising candidate for a wide range of eye tracking applications.

An important finding of the study is on the efficiency of infrared sensors. It is observed that the set of 12 sensors contains a collective set of information, which is useful for arbitrary users. Therefore, no particular sensor was selected as “more important” or “redundant”.

Another useful characteristic of Eye Touch system is its convenience for left, right or double click operations (which corresponds to left blink, right blink, and all-blink) of the user in the assistive applications. In eye tracking systems, clicking operations are usually handled by some additional mechanisms such as simple switches. As it is shown in the previous section, Eye Touch is able to support all clicking operations with very high accuracy without any switch or gearing by the help of its authentic feature extraction method, i.e. the goggles. Since the reflected light readings significantly differ when the eyelids are closed, the classification of the click events become an easy task for people who are capable of deliberately blinking their eyes. Due to outlier elimination of short and natural blinks, these operations are not confused with the unintentional and natural blinks.

A restriction of the proposed system is about being unable to move the head freely in its current form. This drawback can be fixed by additional chin-rest mechanisms that many eye-tracking systems have already used. Interestingly, the style of using goggles for gaze detection becomes a fortunate advantage in head mounted displays (HMDs), where the vision is displayed on a pair of glasses, so the display and the head moves together. In such systems, with the incorporation of the proposed sensors, a natural eye tracking system can be achieved. In fact, HMDs are getting demanding in parallel with the improvements on mobile systems. With the incorporation of mobile multimedia devices, HMDs become a personal display, e.g. Google’s Project Glass [47]. In those equipments, controlling the system with gestures can be a tough and energy inefficient task for camera based eye tracking due to the mounting, occlusion and sensor (CCD, CMOS) energy consumption problems. In such systems, estimating the user gaze with computation intensive image processing algorithms requires significant computation power and may cause quick battery drains. On the contrary, with an approach similar to Eye Touch, the problem can be solved with an effortless and efficient manner.

As a conclusion, among few approaches to eye tracking, Eye Touch is proposed to become an alternative solution for mobile systems. In this particular work, a dedicated software was developed for the proposed hardware, and tested by several subjects. It is argued that the proposed tool can easily be placed on an eye-glasses frame. With the help of a chin-rest to stabilize and avoid head movement; it can be used to control computerized hardware without leaving your hands off your job. With possible incorporation of head motion compensation or by implementing the sensors on a head mounted display, Eye Touch can also be a very handful interface for those who need to deal with additional controls while doing their actual work. As a result, it is concluded that the idea of IR sensor utilization for eye position sensing is a promising attempt to accurately detecting the eye-gaze direction. In the future, it is planned to integrate the Eye Touch into a wearable computer having an HMD rather than using the system with a desktop computer. Therefore, Eye Touch can be a solution for novel and efficient user interfaces for various applications such as vehicle and wheelchair control for disabled.
APPENDIX A
EYE SAFETY ISSUES FOR GOGGLES

While six IR LEDs illuminate each eye homogeneously, Eye Touch goggles should be proved to be eye safe. That is, the goggles must be compliant with applicable safety standards IEC 825-1 and EN 60825-1 [48]. In order to show the compliance of the goggles with IEC 825-1, Accessible Exposure Limit (AEL) in mW/sr must be first computed for the IR LED (TSUS 5400) used in the goggles where AEL in mW/sr is defined to be [48]:

\[
AEL(mW/sr) = AEL(mW)/\Omega(sr)
\]  
(10)

Accessible Exposure Limit (AEL) in mW is defined to be:

\[
AEL(mW) = \frac{10^2}{\pi} \left[ 7 \times 10^{-4} \cdot 0.75 \cdot 10^{0.002(\lambda-700)} \cdot \left( \frac{\alpha}{\Omega_{min}} \right) \right]
\]  
(11)

where \( t \) is the exposure duration in seconds, \( \lambda \) is the wavelength in nanometers, \( \alpha_{min} \) is the apparent source angular subtense above which the source is considered an extended source in milliradians (mrad), and \( \alpha = 2000 \tan^{-1}(s/100) \) is the angle subtended that completely contains the apparent source size at a distance of 100 mm from the source in milliradians with \( s \) being the apparent source size in millimeters. For the IR LEDs, AEL is found to be 1.845 mW by taking \( t = 100 \text{ sec} \), \( \alpha = 950 \text{ nm} \), \( \alpha_{min} = 11 \text{ mrad} \), \( s = 2.90 \text{ mm} \) (from the datasheet of TSUS 5400 [49]) and \( \alpha = 29 \text{ mrad} \).

The solid angle \( \Omega(sr) \) is defined to be:

\[
\Omega(sr) = 2\pi(1 - \cos(\tan^{-1}(d/2r)))
\]  
(12)

where \( d = 7 \text{ mm} \) and \( r = 100(s/10 + 0.0046)^0.5 \text{ mm} \). For the IR LEDs, the solid angle is found to be 0.013 sr by taking \( s = 2.90 \text{ mm} \). Finally AEL becomes:

\[
AEL \text{ (mW/sr)} = 1.845 \text{ mW/0.013 sr} = 141 \text{ mW/sr}
\]  
(13)

In order for the goggles to be eye safe, the total radiant intensity due to six IR LEDs must be less than 141 mW/sr. During the normal operation of the goggles, each IR LED draws about 30 mA forward current, which results in a radiant intensity value of 10 mW/sr [49]. As a result, six IR LEDs produce a total radiant intensity value of 60 mW/sr, which is well under the allowed AEL for the goggles to be eye safe.

REFERENCES


Cihan Topal received the B.Sc. degree in Electrical and Electronics Engineering and M.S. degree in Computer Engineering, both from Anadolu University, in 2005 and 2008 respectively. He worked as a student intern in the Real-time Vision and Modeling Department at Siemens Corporate Research, Princeton, New Jersey in 2008. He is currently working at Anadolu University Computer Engineering Department, Turkey, towards his Ph.D. degree. His research interests are in the area of image processing, pattern recognition, computer vision and human-computer interaction.

Serkan Gunal received the B.S., M.S., and PhD. degrees in electrical and electronics engineering from Eskisehir Osmangazi University, Eskisehir, Turkey, in 1999, 2003, and 2008, respectively. He was a Research Assistant at the same university between 1999 and 2001. He then spent several years as an R&D Engineer at IT companies. In 2006, he joined the Department of Computer Engineering, Anadolu University, Eskisehir, Turkey, where he is currently an Assistant Professor. His research interests include pattern analysis and recognition.

Onur Koçdereven received a BSc degree in Electrical and Electronics Engineering department from Anadolu University, Eskisehir, Turkey in 2010. He worked as NGN engineer at a telecommunication company between 2010 and 2012. He is currently working on a R&D company specialized on medical devices. He has broad experience on embedded systems and mobile devices.

Atakan Doğan received his B.S. and M.S. degrees in electrical and electronics engineering from Anadolu University in 1995 and 1997, respectively, and his Ph.D. degree in electrical engineering from The Ohio State University in 2001. After receiving his Ph.D. degree, he worked as a post-doctoral researcher at The Ohio State University for nine months. He, then, joined the faculty at the Department of Electrical and Electronics Engineering, Anadolu University in 2002, where he is currently an associate professor. His current research interests include cluster and grid computing, parallel computing, parallel architectures, reconfigurable computing and embedded systems.

Ömer Nezih Gerek (S’89, M’98, SM’07) received BSc, MSc and PhD degrees in Electrical Engineering from Bilkent University, Ankara, Turkey in 1991, 1993 and 1998, respectively. During his PhD studies, he spent a semester in U. Of Minnesota as an exchange researcher. Following his PhD, he spent 1 year as a research associate at EPFL, Lausanne, Switzerland. Currently, he is a full professor of Electrical Engineering at Anadolu University, Eskisehir, Turkey. He is also a member of the Electrical, Electronics and Informatics Research Fund Group of the Scientific and Technological Research Council of Turkey. He is in the Editorial Board of Turkish Journal of Electrical Engineering and Computer Science and Elsevier: Digital Signal Processing. His research areas include signal analysis, image processing, and signal coding.