On-Screen Point-of-Regard Estimation Under Natural Head Movement for a Computer with Integrated Webcam

Stefania Cristina, Kenneth P. Camilleri
Department of Systems and Control Engineering
University of Malta, Malta
Email: stefania.cristina@um.edu.mt
kenneth.camilleri@um.edu.mt

Abstract—**Recent developments in the field of eye-gaze tracking by vidoeoculography indicate a growing interest towards unobtrusive tracking in real-life scenarios, a new paradigm referred to as pervasive eye-gaze tracking. Among the challenges associated with this paradigm, the capability of a tracking platform to integrate well into devices with in-built imaging hardware and to permit natural head movement during tracking is of importance in less constrained scenarios. The work presented in this paper builds on our earlier work, which addressed the problem of estimating on-screen point-of-regard from iris center movements captured by an integrated camera inside a notebook computer, by proposing a method to approximate the head movements in conjunction with the iris movements in order to alleviate the requirement for a stationary head pose. Following iris localization by an appearance-based method, linear mapping functions for the iris and head movement are computed during a brief calibration procedure permitting the image information to be mapped to a point-of-regard on the monitor screen. Following the calculation of the point-of-regard as a function of the iris and head movement, separate Kalman filters improve upon the noisy point-of-regard estimates to smoothen the trajectory of the mouse cursor on the monitor screen. Quantitative and qualitative results obtained from two validation procedures reveal an improvement in the estimation accuracy under natural head movement, over our previous results achieved from earlier work.**

Keywords–Point-of-regard estimation; Eye-gaze tracking; Iris center localization; Head movement tracking; Integrated webcam.

I. INTRODUCTION

The idea of estimating the human eye-gaze has been receiving increasing interest since at least the 1870s, following the realization that the eye movements hold important information that relates to visual attention. Throughout the years, efforts in improving eye-gaze tracking devices to minimize discomfort and direct contact with the user led to the conception of videooculography (VOG), whereby the eye movements are tracked remotely from a stream of images that is captured by digital cameras. Eye-gaze tracking by VOG quickly found its way into a host of applications, ranging from human-computer interaction (HCI) [1]–[3], to automotive engineering [4][5]. Indeed, with the advent of the personal computer, eye-gaze tracking technology was identified as an alternative controlling medium enabling the user to operate the mouse cursor using the eye movements alone [3].

Following the emergence and widespread use of highly mobile devices with integrated imaging hardware, there has been an increasing interest in mobile eye-gaze tracking that blends well into the daily life setting of the user [6]. This emerging interest led to the conception of a new paradigm that is referred to as *pervasive tracking*, which refers to the endeavor of tracking the eye movements continuously in different real-life scenarios [6]. This notion of pervasive eye-gaze tracking is multi-faceted, typically characterized by different aspects such as the capability of a tracking platform to permit tracking inside less constrained conditions, to track the user remotely and unobtrusively and to integrate well into devices that already comprise imaging hardware without necessitating hardware modification.

Nevertheless, this new paradigm brings challenges that go beyond the typical conditions for which classical video-based eye-gaze tracking methods have been developed. Despite considerable advances in the field of eye-gaze tracking as evidenced by an abundance of methods proposed over the years [7], video-based eye-gaze tracking has been mainly considered as a desktop technology, often requiring specific conditions to operate. Commercially available eye-gaze tracking systems, for instance, are usually equipped with high-grade cameras and actively project infra-red illumination over the face and the eyes to obtain accurate eye movement measurements. In utilizing specialized hardware to operate, active eye-gaze tracking fails to integrate well into devices that already comprise imaging hardware, while its usability is constrained to controlled environments away from interfering infra-red sources. On the other hand, passive eye-gaze tracking that operates via standard imaging hardware and exploits the appearance of the eye without relying on specialized illumination sources for localization and tracking, provides a solution that promises to integrate better into pervasive scenarios.

Nonetheless, utilizing existing passive eye-gaze tracking methods to address the challenges associated with pervasive tracking, such as the measurement of eye movement from low-quality images captured by lower-grade hardware, may not necessarily be a suitable solution. For instance, existing shape-based methods that localize the eye region inside an image frame by fitting curves to its contours, often require images of suitable quality and good contrast in which the boundaries between different components such as the eyelids, the sclera and the iris are clearly distinguishable [8]–[11]. Similarly,

feature-based methods that search for distinctive features such as the limbus boundary [12][13], necessitate these features to be clearly identifiable. Appearance-based methods relying on a trained classifier, such as a Support Vector Machine (SVM), to estimate the 2D point-of-regard directly from an eye region image without identifying its separate components, have been reported to perform relatively well on lower-quality images as long as the training data includes images of similar quality as well [14]. However, this performance usually comes at the cost of lengthy calibration sessions that serve to gather the user-dependent data that is required for training [15][16].

Moreover, recent attempts to track the eye-gaze on mobile platforms by existing eye-gaze tracking methods [17][18] have reported undesirable constraints such as a requirement for close-up eye region images [19], lengthy calibration sessions [17] and stationary head poses [15][20][21] often with the aid of a chin-rest [15][20]. The capability of a tracking platform to allow head movement during gaze estimation, in particular, is an important aspect in the context of pervasive eye-gaze tracking, permitting the user to move naturally without constraining the tracking conditions. Methods that do not cater for head movement during tracking and calibration often estimate an image-to-screen mapping function that is valid for a single head pose alone, hence requiring a stationary head pose during and following calibration [15][20][21]. While this may be considered as a workable solution for short-term use, it does not provide for a comfortable setup over longer periods of time. Head movement compensation is, therefore, required in order to lift the constraint of maintaining a stationary head pose, permitting small head displacement if the movement is measured by a simple head marker [22][23] or higher degrees-of-freedom if the head pose is calculated in 3-dimensional space [9][24][25]. In this regard, several eye-gaze tracking methods that cater for head movement via an appearance-based approach generally populate a training dataset with images captured under different head orientations [26][27], often at the expense of presenting the user with a larger set of calibration targets resulting in a data collection session that is considerably prolonged [26]. Feature-based approaches for head pose estimation, on the other hand, generally follow a model-fitting approach whereby a face model is fit to specific face feature landmarks allowing the estimation of the head rotation angles [19][28][29]. The accuracy of these approaches is often contingent upon accurate tracking of several facial features, which is in turn susceptible to feature distortion and self-occlusion during head rotations [19].

In light of the challenges associated with pervasive tracking, we propose a passive eye-gaze tracking method to estimate the point-of-regard (POR) on a monitor screen from lower-quality images acquired by an integrated camera inside a notebook computer, while approximating any natural, typically small, head rotations performed by the user during tracking. To localize the iris center coordinates from low-resolution eye region images while the user sits at a distance from the monitor screen, we propose an appearance-based method that localizes the iris region by its intensity values. In addition, our method ensures that the iris region can be located at different angles of eyeball and head rotation and under partial occlusion by the eyelids, and can be automatically relocated after this has been entirely occluded during blinking. Following iris localization, the iris center coordinates extracted earlier are mapped to a POR on the monitor screen via linear mapping functions that are estimated through a brief calibration procedure. Linear mapping functions for the head movement are also estimated during calibration, ensuring that any natural head rotations performed by the user during tracking are handled according to the resulting face region displacement inside the image space. Kalman filters handling the iris and head movement separately are assigned to each of four screen quadrants, characterised by different mapping functions, in order to improve upon the noisy POR estimates computed as a function of the iris and head movement. This serves to smoothen the trajectory of the mouse cursor on the monitor screen.

The details of the proposed passive eye-gaze tracking method are described in Section II. Section III presents and discusses the experimental results. A comparison between the results achieved by our method and those reported by relevant state-of-the-art methods is provided in Section IV, while Section V draws the final remarks that conclude the paper.

II. METHOD

The following sections describe the stages of the proposed method, starting off with eye region detection and tracking up to the estimation of the POR onto the monitor screen.

A. Eye Region Detection

The estimation of the POR on the monitor screen requires that the eye region is initially detected inside the first few image frames. Searching for the eye region over an entire image frame can be computationally expensive for a real-time application and can lead to the occurrence of several false positive detections. Therefore, prior to detecting the eye region, the bounding box that encloses the face region is detected first such that this constrains the search range for the eye region, reducing the searching time as well as the possibility of false positives. The eye region is subsequently detected within the area delimited by the boundaries of the face region.

Given the real-time nature of our application, we chose the Viola-Jones algorithm for rapid detection of the face and eye region [30]. Within the Viola-Jones framework, features of interest are detected by sliding rectangular windows of Haar-like operators over an image frame, subtracting the underlying image pixels that fall within the shaded regions of the Haar-like operators from the image pixels that fall within the clear regions. Candidate image patches are classified between positive and negative samples by a cascade of weak classifiers arranged in order of increasing complexity. Every weak classifier is trained to search for a specific set of Haar features by a technique called boosting, such that each stage processes the samples that pass through the preceding classifier and rejects the negative samples as early into the cascade as possible to ensure computational efficiency.

The face and eye region detection stages in our work utilize freely available cascades of classifiers that come with the OpenCV library [31], which had been previously trained on a wide variety of training images such that detection generalizes well across different users. Since the training data for these classifiers was mainly composed of frontal face and eye region samples, the user is required to hold a frontal head pose for a brief period of time until the face and eye regions have been successfully detected. In case multiple candidates are detected by the face region classifier, the proposed method chooses the candidate that is closest to the monitor screen characterized by the largest bounding box, and discards the others.

B. Eye Region Tracking

To allow for small and natural head movement during tracking without requiring the uncomfortable use of a chin-rest, the initial position of the eye region detected earlier needs to be updated at every image frame to account for its displacement in the x- and y-directions. While performing eye region detection on a frame-by-frame basis would be a possible solution to estimate the eye region displacement through an image sequence, such an approach would be sub-optimal in terms of computational efficiency for a real-time application. Therefore, assuming gradual and small head displacement, the eye region is tracked between successive image frames by template matching, using the last known position of the eye region inside the previous image frame to constrain the search area inside the next frame.

A template image of the eye region is captured and stored following earlier detection of this region by the Viola-Jones algorithm. The template image is then matched to the search image inside a window of fixed size, centered around the last known position of the feature of interest. Template matching utilizes the normalized sum of squared differences (NSSD) as a measure of similarity, denoted as follows,

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

where *T* denotes the template image and *I* denotes the search image. A NSSD value of zero represents a perfect match between the template and search image, whereas a higher value denotes increasing mismatch between the two images. This permits the identification of the new position of the feature of interest, which is specified by the location inside the search image that gives the minimum NSSD value after template matching.

C. Iris Center Localization

The movement of the eyes is commonly represented by the trajectory of the iris or pupil center in a stream of image frames [7], and hence the significance of localizing the iris or pupil center coordinates after the eye region has been detected. Given the small footprint of the eye region inside the image space, we opt to localize the iris center coordinates rather than the pupil, since the iris occupies a larger area inside the eye region and can be detected more reliably.

While there exist different methods that permit localization of the iris region inside an image frame, not all of these methods are suitable for localizing the iris region from low-resolution images, especially if fine details such as the contours of different components of the eye [8]–[11] need to be clearly distinguishable. We propose an appearance-based method that segments the iris region via a Bayes' classifier to localize it. The Bayes' classifier is trained during an offline training stage to classify between iris and non-iris pixels based on their red channel value in the RGB color space. During tracking, intensity values of pixels residing within the eye region are classified as belonging to the iris region if their likelihood exceeds a pre-defined threshold value, θ:

$$\frac{p(x_r(i,j) \mid \varpi_{iris})}{p(x_r(i,j) \mid \varpi_{non-iris})} \geq \theta \quad (2)$$

where $p(x_r(i,j) \mid \varpi_{iris})$ denotes the class-conditional probability of observing a red-band measurement at pixel *(i, j)* knowing it belongs to the iris class, while $p(x_r(i,j) \mid \varpi_{non-iris})$ denotes the class-conditional probability of observing the same red-band measurement at pixel *(i, j)* knowing it belongs to the non-iris class. The resulting binary image contains a blob of pixels that belongs to the iris region, whose center of mass is taken to represent the iris center coordinates. In case the eyebrow is also mistakenly classified as belonging to the iris region due to the resemblance in color with dark irises, the blob of pixels that is closer to the center of the eye region is considered to represent the iris.

The Bayes' classifier had been previously used for skin region segmentation in images [32], but to our knowledge it has never been adopted to the problem of iris region localization for eye-gaze tracking until our work of [1]. Preliminary results have shown this method to be suitable in localizing the iris region from low-quality images, owing especially to the fact that the proposed localization method depends upon statistical color modeling rather than geometrical information. Another advantage that is also related to its independency from geometrical information is the ability to locate the iris region at different angles of rotation and under partial occlusion by the eyelids. The main downside of this method is its susceptibility to illumination variations, which problem is however alleviated by training the Bayes' classifier on iris and non-iris pixels acquired under different illumination conditions.

D. POR Estimation

Having determined the iris center coordinates, the final stage seeks to estimate the user's POR on the monitor screen from the iris displacement while approximating any head rotations performed by the user during tracking.

For simplicity, and since we expect small eyeball and head rotation angles within the width and height of the monitor screen at close range, the eyeball and head rotations performed

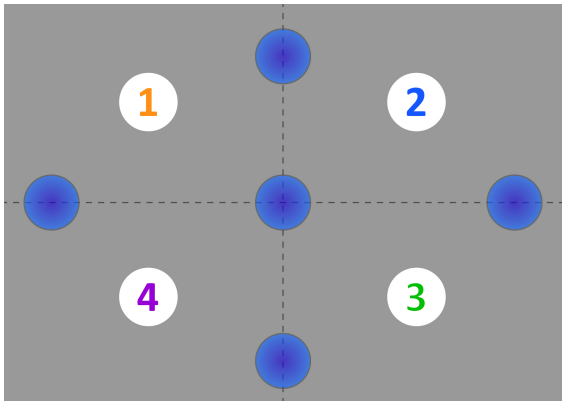


Figure 1. Strategically placed calibration points divide the screen display into four separate quadrants.

by the user during tracking are approximated by planar displacement of the respective iris center and face region inside the image space. The on-screen displacement, $\Delta\mathbf{x}_{s_g}$, of the user's POR may therefore be defined as a function of the iris displacement, $\Delta\mathbf{x}_{s_e}$, and head displacement, $\Delta\mathbf{x}_{s_h}$, on the monitor screen as follows,

$$\Delta\mathbf{x}_{s_g} = \Delta\mathbf{x}_{s_e} + \Delta\mathbf{x}_{s_h} \tag{3}$$

In order to estimate the on-screen POR displacement, we seek to determine image-to-screen mapping relationships for the iris and head components. The assumption of planar iris center displacement permits us to define a linear mapping relationship between the image and screen coordinates as follows,

$$(\mathbf{x}_{s_e}^{(3)} - \mathbf{x}_s^{(1)}) = \frac{(\mathbf{x}_s^{(2)} - \mathbf{x}_s^{(1)})}{(\mathbf{x}_{i_e}^{(2)} - \mathbf{x}_{i_e}^{(1)})}(\mathbf{x}_{i_e}^{(3)} - \mathbf{x}_{i_e}^{(1)}) \tag{4}$$

where $\mathbf{x}_s^{(1)}$ and $\mathbf{x}_s^{(2)}$ denote the screen coordinates of two calibration points respectively, whereas $\mathbf{x}_{i_e}^{(1)}$ and $\mathbf{x}_{i_e}^{(2)}$ denote the corresponding iris center coordinates inside the eye region which are estimated while the user fixates at the two calibration points maintaining the head stationary. During tracking, the mapping function in (4) computes the displacement in screen coordinates between the new POR $\mathbf{x}_{s_e}^{(3)}$ and the calibration point $\mathbf{x}_s^{(1)}$, following the estimation of the displacement in image coordinates between the new iris center location $\mathbf{x}_{i_e}^{(3)}$ and the previously estimated $\mathbf{x}_{i_e}^{(1)}$. In order to compensate for the assumption of planar iris movement, the monitor screen is divided into four separate quadrants by strategically placed calibration points as illustrated in Figure 1, such that each quadrant is assigned different parameter values that best describe the linear mapping between the image-to-screen coordinates.

Similarly, a linear image-to-screen mapping relationship for the head displacement is defined as follows,

$$(\mathbf{x}_{s_h}^{(3)} - \mathbf{x}_s^{(1)}) = \frac{(\mathbf{x}_s^{(2)} - \mathbf{x}_s^{(1)})}{(\mathbf{x}_{i_h}^{(2)} - \mathbf{x}_{i_h}^{(1)})}(\mathbf{x}_{i_h}^{(3)} - \mathbf{x}_{i_h}^{(1)}) \tag{5}$$

requiring the estimation of image coordinates, $\mathbf{x}_{i_h}^{(1)}$ and $\mathbf{x}_{i_h}^{(2)}$, while the user fixates at the corresponding on-screen calibration targets, $\mathbf{x}_s^{(1)}$ and $\mathbf{x}_s^{(2)}$, by the head pose alone maintaining the eyes stationary. In order to reduce the calibration effort and re-use the iris mapping relationship estimated earlier for the visual targets in Figure 1, the user is requested to perform head rotations in the horizontal and vertical directions while fixating at a single calibration target. Since the user's POR on the monitor screen is maintained fixed during the head movement, Equation (3) reduces to,

$$\Delta\mathbf{x}_{s_e} + \Delta\mathbf{x}_{s_h} = 0 \tag{6}$$

such that the on-screen relationship between the iris and head displacement may be defined as follows,

$$\Delta\mathbf{x}_{s_e} = -\Delta\mathbf{x}_{s_h} \tag{7}$$

Inside the image space, $\Delta\mathbf{x}_{i_e} \propto \Delta\mathbf{x}_{i_h}$ up to a scale factor S which models the relationship between the different rotation radii of the eyeball and the head around their respective axis. As the user performs arbitrary head movement during calibration, tuples of coordinates $\mathbf{x}_{i_e}$ and $\mathbf{x}_{i_h}$ are collected allowing the scale factors S_x and S_y, in the horizontal and vertical directions, to be estimated as follows,

$$S_x = \frac{\Delta x_{i_h}}{\Delta x_{i_e}} \qquad S_y = \frac{\Delta y_{i_h}}{\Delta y_{i_e}} \tag{8}$$

where displacements, $\Delta\mathbf{x}_{i_e}$ and $\Delta\mathbf{x}_{i_h}$, are estimated with respect to initial eye and head poses. Equation (5) may hence be re-defined as,

$$(\mathbf{x}_{s_h}^{(3)} - \mathbf{x}_s^{(1)}) = \frac{(\mathbf{x}_s^{(2)} - \mathbf{x}_s^{(1)})}{\mathbf{S} \circ (\mathbf{x}_{i_e}^{(2)} - \mathbf{x}_{i_e}^{(1)})}(\mathbf{x}_{i_h}^{(3)} - \mathbf{S} \circ \mathbf{x}_{i_e}^{(1)}), \tag{9}$$

$$\mathbf{S} = \begin{pmatrix} S_x \\ S_y \end{pmatrix}$$

where $\mathbf{x}_{s_h}^{(3)}$ denotes the new POR determined by the head movement alone, corresponding to a new head region location $\mathbf{x}_{i_h}^{(3)}$ inside the image space. Having determined the iris, $\Delta\mathbf{x}_{s_e}$, and head displacement, $\Delta\mathbf{x}_{s_h}$, following Equations (4) and (9), respectively, the on-screen POR displacement may be estimated from Equation (3).

To alleviate the issue of noisy iris center and head displacement measurements from low-quality images, and hence smoothen the trajectory of the mouse cursor on the monitor screen after mapping the iris and head movement to a POR, we propose to employ Kalman filtering to improve upon these noisy measurements. Indeed, the Kalman filter is an algorithm that recursively utilizes noisy measurements observed over time to produce estimates of desired variables that tend to be more accurate than the single measurements alone [33]. We define separate Kalman filters to handle the respective iris and

head displacement for our specific application of smoothing the mouse cursor trajectory as follows:

State Vector: We define the state vector $\mathbf{x}_{k+1}^{(e)}$ as,

$$\mathbf{x}_{k+1}^{(e)} = [\Delta x_{s_e} \quad \Delta y_{s_e}]_{k+1}^T \tag{10}$$

comprising the horizontal iris centre displacement, $\Delta x_{s_e} = (x_{s_e}^{(3)} - x_s^{(1)})$, and similarly for the vertical iris centre displacement displacement, Δy_{s_e}. Similarly, for the other Kalman filter, we define the state vector $\mathbf{x}_{k+1}^{(h)}$ as,

$$\mathbf{x}_{k+1}^{(h)} = [\Delta x_{s_h} \quad \Delta y_{s_h}]_{k+1}^T \tag{11}$$

where Δx_{s_h} and Δy_{s_h} denote the horizontal and vertical on-screen head displacement.

Transition Matrix: Assuming the eye and head movement during tracking to consist of fixation periods and smooth movement between one visual stimulus and another, we represent the transition matrices $A^{(e)}$ and $A^{(h)}$ by simple linear models of the ideal mouse cursor trajectory during constant velocity movement as follows,

$$A^{(e)} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{12}$$

and,

$$A^{(h)} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{13}$$

Measurement Vector: In our work, the measurement vector $\mathbf{z}_{k+1}^{(e)}$, attributed to the Kalman filter that handles the iris movement, holds the estimated image displacement of the iris center coordinates as follows,

$$\mathbf{z}_{k+1}^{(e)} = [\Delta x_{i_e} \quad \Delta y_{i_e}]_{k+1}^T \tag{14}$$

where Δx_{i_e} and Δy_{i_e} represent the horizontal and vertical image displacement respectively. Similarly, the measurement vector, $\mathbf{z}_{k+1}^{(h)}$, belonging to the Kalman filter that handles the head movement is defined as follows,

$$\mathbf{z}_{k+1}^{(h)} = [\Delta x_{i_h} \quad \Delta y_{i_h}]_{k+1}^T \tag{15}$$

where Δx_{i_h} and Δy_{i_h} denote the horizontal and vertical head displacement inside the image space.

Measurement Matrix: The measurement matrix defines the relationship that maps the true state space onto the measurement. In our work, the values that populate the measurement matrices can be derived from Equations (4) and (9), such that these matrices map the screen coordinates onto the image coordinates. The measurement matrix corresponding to the Kalman filter that handles the iris displacement is defined as,

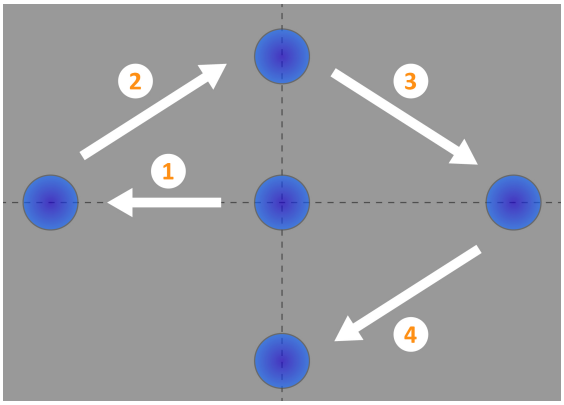


Figure 2. Five visual stimuli were displayed in succession on the monitor screen during a brief calibration procedure in order to collect image-screen coordinate pairs.

$$H_{k+1}^{(e)} = \begin{bmatrix} \frac{(x_{i_e}^{(2)} - x_{i_e}^{(1)})}{(x_s^{(2)} - x_s^{(1)})} & 0 \\ 0 & \frac{(y_{i_e}^{(2)} - y_{i_e}^{(1)})}{(y_s^{(2)} - y_s^{(1)})} \end{bmatrix} \tag{16}$$

while the measurement matrix attributed to the Kalman filter that handles the head movement is defined as,

$$H_{k+1}^{(h)} = \begin{bmatrix} \frac{S_x(x_{i_e}^{(2)} - x_{i_e}^{(1)})}{(x_s^{(2)} - x_s^{(1)})} & 0 \\ 0 & \frac{S_y(y_{i_e}^{(2)} - y_{i_e}^{(1)})}{(y_s^{(2)} - y_s^{(1)})} \end{bmatrix} \tag{17}$$

Measurement Noise and Process Noise: The measurement noise is represented by vector, $\mathbf{v}_{k+1} = [v_{k+1}^x v_{k+1}^y]$, characterized by standard deviations σ_{v_x} and σ_{v_y} in the x- and y-directions, respectively, and similarly the process noise is represented by vector, $\mathbf{w}_{k+1} = [w_{k+1}^x w_{k+1}^y]$, characterized by standard deviations σ_{w_x} and σ_{w_y} in the respective x- and y-directions. The process noise corresponding to the iris movement is taken to represent the characteristics inherent to the visual system itself, such that the standard deviations $\sigma_{w_x}^{(e)}$ and $\sigma_{w_y}^{(e)}$ are therefore set to a low value to model the small, microsaccadic movements performed by the eye during periods of fixation. Similarly, the standard deviations $\sigma_{w_x}^{(h)}$ and $\sigma_{w_y}^{(h)}$ characterizing the process noise corresponding to the head movement are also set to a low value. Values for the standard deviations, $\sigma_{v_x}^{(e)}$ and $\sigma_{v_y}^{(e)}$, corresponding to the iris displacement and standard deviations, $\sigma_{v_x}^{(h)}$ and $\sigma_{v_y}^{(h)}$, corresponding to the head movement that adequately smooth the mouse cursor trajectory after the estimation of noisy image measurements were found experimentally.

Separate Kalman filters are assigned to every screen quadrant, with each filter being characterized by a different measurement matrix corresponding to the screen quadrant for which it is responsible. During tracking, all Kalman filters are updated online to produce an estimate of the POR following

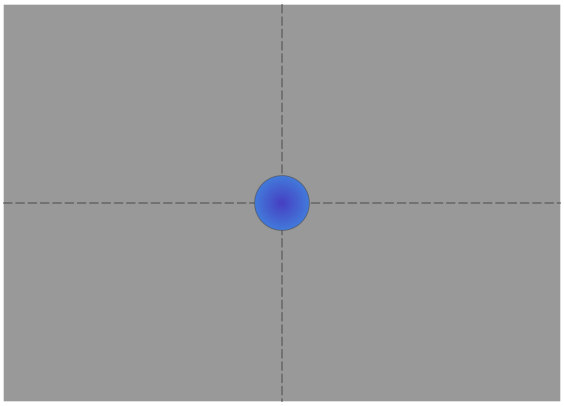


Figure 3. An additional calibration target was displayed at the center of the monitor screen while the participants performed horizontal and vertical head rotations, maintaining a fixed gaze point.

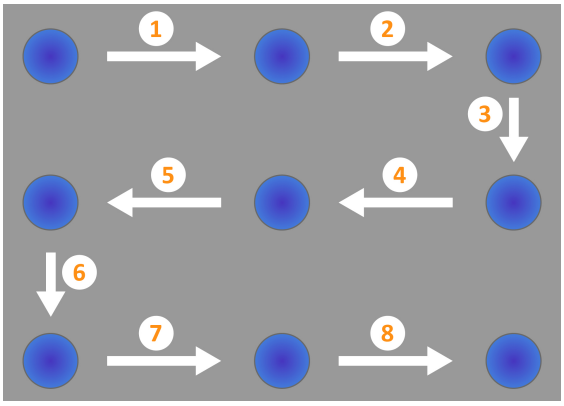


Figure 4. A validation session consisting of nine visual stimuli displayed in succession served to calculate the error of the estimated PORs.

the estimation of the iris center coordinates, such that the on-screen position of the mouse cursor can subsequently be updated according to the Kalman filter estimate that corresponds to the quadrant of interest. In updating the Kalman filters at every time step, we ensure a smooth hand over between one filter and another as the mouse cursor trajectory crosses over adjacent screen quadrants.

III. Experimental Results and Discussion

To evaluate the proposed eye-gaze tracking method, a group of nine female and male participants having a mean age of 35.1 and standard deviation of 12.7, were recruited for an experimental session. All participants were proficient computer users without any prior experience in the field of eye-gaze tracking, except for one participant who was already accustomed to the technology. The experimental procedure was carried out on a 15.6" notebook display while each participant was seated inside a well-lit indoor environment at an approximate distance of 60 cm from the monitor screen and the camera. At this distance, combined eyeball and head rotations were carried out within a $\pm 15^{\circ}$ range, corresponding to the width and height of the monitor screen. Image data was acquired by the webcam that was readily available on-board the notebook computer.

Following detection and tracking of the eye region, and iris center localization, each participant was requested to sit through a brief calibration procedure that served to estimate the mapping functions required to transform the iris center and head displacement inside the image space into a POR on the monitor screen. During the calibration procedure, the participants were first instructed to fixate at five visual stimuli appearing in succession on the monitor screen as shown in Figure 2, and requested to maintain a stationary head pose while pairs of image-screen coordinates were collected. Subsequently, an additional calibration target was displayed at the center of the monitor screen as shown in Figure 3, during which the participants were instructed to perform horizontal and vertical head rotations while maintaining their gaze fixed onto the visual target. The five visual stimuli displayed successively during calibration were positioned strategically in order to divide the screen into four separate quadrants, as illustrated in Figure 1. A different mapping function was estimated for each quadrant according to the relationship between the image and screen coordinates collected earlier.

Every participant was then requested to sit through two validation procedures that served to calculate the error between the estimated POR and ground truth data. The first validation procedure consisted of nine visual stimuli which were evenly spread throughout the monitor screen and displayed in succession as shown in Figure 4. The participants were allowed natural head movement, and instructed to move the mouse cursor as close to each visual stimulus as possible and hold its position for a brief period of time such that the on-screen coordinates of the mouse cursor were recorded, as shown in Figure 7 for participant 7. Tables I and II display the mean absolute error (MAE) and standard deviation (SD) in pixels for each validation target in the x- and y-directions. The second validation procedure served to evaluate the accuracy of head pose compensation for a fixed POR on the monitor screen. To this end, the participants were requested to maintain their gaze fixed upon a single visual stimulus displayed on the monitor screen and perform arbitrary head rotations in the horizontal and vertical directions while the mouse cursor coordinates were recorded. Table IV presents the MAE and SD in pixels calculated between the cursor position and validation target in the x- and y-directions. Qualitative results of the iris segmentation method are presented in Figure 6. The proposed method was coded in C using the OpenCV library [31] and was capable of executing in real-time at 18 fps on an Intel Core i5 notebook computer, at 2.60 GHz.

An analysis of the results in Tables I and II reveals that for the majority of the validation targets, the MAE and SD values in pixel units for the x-coordinates of the estimated PORs exceed the error values corresponding to the y-coordinates across all participants. One of the main sources for this consistent discrepancy in error was found to relate to inaccuracies in the estimation of the iris center coordinates, as indicated by the iris segmentation results in Figures 6(a-e). As shown in the binary images, dark colored pixels belonging to the eyelashes are often erroneously included with the iris region

TABLE I. MEAN ABSOLUTE ERROR (MAE) AND STANDARD DEVIATION (SD) IN PIXELS, OF THE ESTIMATED ON-SCREEN POR X-COORDINATES FOR EACH VALIDATION TARGET.

Participant	Validation Target								
	1	2	3	4	5	6	7	8	9
	x-Coordinate (MAE (pix), SD (pix)))								
1	(18.32, 13.09)	(37.96, 28.02)	(20.81, 11.74)	(28.12, 18.30)	(6.19, 3.05)	(34.36, 5.27)	(9.03, 6.74)	(10.88, 5.08)	(34.09, 13.42)
2	(27.12, 15.40)	(12.91, 13.36)	(38.92, 5.41)	(8.66, 5.46)	(9.03, 8.06)	(47.25, 21.30)	(6.99, 6.00)	(18.11, 12.28)	(37.96, 13.04)
3	(22.00, 13.13)	(17.91, 9.14)	(20.06, 11.27)	(9.95, 7.75)	(10.66, 8.37)	(22.47, 8.10)	(7.67, 5.03)	(20.65, 8.01)	(33.64, 9.38)
4	(20.38, 8.80)	(37.17, 24.06)	(37.61, 6.18)	(32.55, 20.05)	(55.94, 38.20)	(30.23, 0.67)	(12.47, 3.70)	(26.58, 5.93)	(34.91, 7.94)
5	(10.66, 2.14)	(27.80, 14.63)	(58.44, 25.70)	(20.68, 13.11)	(13.64, 3.18)	(38.37, 4.46)	(6.98, 6.23)	(33.85, 18.78)	(36.80, 15.25)
6	(7.72, 4.46)	(20.16, 3.37)	(23.90, 6.73)	(4.16, 2.28)	(25.54, 7.13)	(27.10, 8.28)	(10.81, 12.31)	(15.61, 3.53)	(20.73, 8.51)
7	(29.85, 7.43)	(32.49, 7.69)	(40.82, 14.18)	(11.53, 6.15)	(13.95, 3.89)	(49.05, 10.89)	(13.57, 7.82)	(9.22, 6.09)	(19.35, 4.86)
8	(17.67, 4.95)	(8.07, 0.96)	(17.22, 4.36)	(30.95, 11.36)	(15.37, 15.08)	(23.41, 5.55)	(6.01, 2.38)	(6.10, 3.96)	(27.41, 4.34)
9	(14.22, 10.92)	(17.90, 9.79)	(25.75, 15.29)	(20.49, 17.14)	(17.32, 9.49)	(26.92, 23.10)	(18.88, 8.10)	(10.71, 5.05)	(47.01, 22.97)
Mean	(18.66, 8.92)	(23.60, 12.34)	(31.50, 11.21)	(18.57, 11.29)	(18.63, 10.72)	(33.24, 9.74)	(10.27, 6.48)	(16.86, 7.63)	(32.43, 11.08)

TABLE II. MEAN ABSOLUTE ERROR (MAE) AND STANDARD DEVIATION (SD) IN PIXELS, OF THE ESTIMATED ON-SCREEN POR Y-COORDINATES FOR EACH VALIDATION TARGET.

Participant	Validation Target								
	1	2	3	4	5	6	7	8	9
	y-Coordinate (MAE (pix), SD (pix)))								
1	(17.58, 5.76)	(17.06, 7.60)	(11.75, 2.27)	(26.60, 20.07)	(16.26, 9.50)	(10.80, 7.35)	(31.84, 16.23)	(6.63, 6.80)	(7.29, 6.97)
2	(11.48, 8.46)	(49.17, 14.13)	(7.75, 2.02)	(13.42, 8.65)	(9.96, 6.17)	(33.21, 19.31)	(22.28, 5.19)	(23.80, 13.06)	(14.37, 8.63)
3	(16.31, 13.45)	(11.97, 9.08)	(16.03, 7.98)	(5.64, 3.65)	(13.16, 11.03)	(13.38, 8.06)	(20.53, 7.51)	(12.20, 7.84)	(10.36, 5.93)
4	(20.03, 8.98)	(30.14, 11.08)	(34.84, 8.96)	(29.59, 7.31)	(19.27, 8.11)	(10.69, 5.13)	(13.87, 6.49)	(18.14, 9.72)	(22.98, 2.82)
5	(4.53, 2.87)	(7.50, 2.91)	(15.12, 7.08)	(12.06, 5.85)	(22.44, 7.50)	(16.74, 1.55)	(44.73, 28.11)	(31.64, 16.13)	(21.39, 18.39)
6	(8.58, 4.55)	(11.50, 4.93)	(29.49, 21.50)	(3.67, 2.43)	(6.34, 3.06)	(20.62, 14.87)	(31.06, 19.01)	(14.28, 10.16)	(11.27, 7.39)
7	(4.58, 3.52)	(26.90, 2.01)	(27.49, 3.28)	(21.19, 5.79)	(4.06, 3.06)	(3.87, 2.03)	(6.97, 4.39)	(22.06, 4.14)	(15.09, 7.48)
8	(56.07, 5.24)	(31.06, 1.83)	(16.35, 2.11)	(35.15, 4.03)	(11.01, 9.04)	(7.63, 4.49)	(23.50, 3.04)	(8.12, 5.36)	(11.75, 4.61)
9	(20.52, 13.49)	(3.54, 3.39)	(10.14, 6.11)	(12.34, 9.41)	(13.60, 6.50)	(37.78, 33.79)	(23.47, 16.18)	(14.08, 7.00)	(11.11, 11.15)
Mean	(17.74, 7.37)	(20.98, 6.33)	(18.77, 6.81)	(17.74, 7.47)	(12.90, 7.11)	(17.19, 10.73)	(24.25, 11.79)	(16.77, 8.91)	(13.96, 8.15)

during segmentation due to their close resemblance in color to dark brown irises. This erroneous inclusion of pixels on either side of the iris region serves to shift the center of mass of the segmented blob of pixels horizontally towards the inner or outer eye corners, away from the true iris center. The horizontal displacement of the iris center gives rise to errors in the estimated PORs, and it was found that even a seemingly trivial error of a few pixels in the estimation of the iris center coordinates inside the image frame could produce a significant POR error at an approximate distance of 60 cm from the monitor screen.

While the majority of the validation targets displayed a higher error in the horizontal component of the estimated PORs, the results for validation target 7 clearly indicate the contrary, that is higher MAE and SD values for the y-coordinates of most participants. The discrepancy in error for this particular validation target relates to its lower left on-screen position as shown in Figure 4, requiring the user to gaze downwards and towards the outer edge of the monitor screen. At this instance, the tracked iris region displaces towards the outer eye corner and becomes partially occluded underneath the upper and lower eyelids, as shown in Figures 6(f-j). The reduced visibility of the true shape of the iris region shifts the localized iris centre downwards, reducing the accuracy of the corresponding POR. Another source of error that reduces the accuracy of the POR y-coordinates in general is the less than ideal positioning of the webcam at the top of the monitor screen, in relation to the positioning of the eyes as the user sits in front of the display. Indeed, commercial systems usually place the tracking device below the monitor screen in order to capture a better view of the visible portion of the eyeball that is not concealed below the eyelid. Being situated at the top of the screen, the webcam that is utilized in our work captures a smaller portion of the iris especially when the user gazes downwards, partially occluding the iris region below the eyelid and potentially introducing an error in the estimation of the iris center coordinates. It is, nonetheless, worth noting that since the iris region is localized by its photometric appearance rather than the shape, the proposed method for iris region segmentation was equally capable of detecting the iris region and hence follow its displacement under partial occlusion by the eyelids.

TABLE III. MEAN ABSOLUTE ERROR (MAE) AND STANDARD DEVIATION (SD) OF THE ERROR IN PIXELS AND VISUAL ANGLE, OF THE ESTIMATED ON-SCREEN POR COORDINATES FOR THE FIRST VALIDATION STAGE.

Participant Number	x	y	x	y
	(MAE (pix), SD (pix))		(MAE (°), SD (°))	
1	(22.20, 11.63)	(16.20, 9.17)	(0.53, 0.28)	(0.39, 0.22)
2	(22.99, 11.15)	(20.60, 9.51)	(0.55, 0.27)	(0.49, 0.23)
3	(18.34, 8.91)	(13.29, 8.28)	(0.44, 0.21)	(0.32, 0.20)
4	(31.98, 12.84)	(22.17, 7.62)	(0.76, 0.31)	(0.53, 0.18)
5	(27.47, 11.50)	(19.57, 10.04)	(0.65, 0.27)	(0.47, 0.24)
6	(17.30, 6.29)	(15.20, 9.77)	(0.41, 0.15)	(0.36, 0.23)
7	(24.43, 7.67)	(14.69, 3.97)	(0.58, 0.18)	(0.35, 0.09)
8	(16.91, 5.88)	(22.29, 4.42)	(0.40, 0.14)	(0.53, 0.11)
9	(22.13, 13.54)	(16.29, 11.89)	(0.53, 0.32)	(0.39, 0.28)
Mean	(22.64, 9.93)	(17.81, 8.30)	(0.54, 0.24)	(0.42, 0.20)

In order to put the error values tabulated in Tables I and II into context, the mean MAE and SD values per participant across all validation targets were subsequently calculated and converted to visual angle at a distance of 60 cm between the user and the monitor screen, as presented in Table III. Given that the resolution of the monitor screen is equal to 1366×768 pixels, a mean MAE value of (22.64, 17.81) in pixel units constitutes less than 2% and 3% of the screen

resolution in the horizontal and vertical directions, respectively. Furthermore, at a distance of 60 cm away from the monitor screen, a mean error of (22.64, 17.81) pixels was found to correspond to an error of (0.54°, 0.42°) in visual angle. It is also worth noting that in approximating any head movement performed by the user during tracking, we have also obtained an improvement of around 37% and 59% in visual angle corresponding to the horizontal and vertical directions respectively, over our previous result of (1.46°, 0.71°) obtained through our earlier work without head movement compensation [1]. This improvement further substantiates the benefit of handling head movement during tracking rather than constraining the user to a stationary head pose, where small head displacements performed by the user due to fatigue may adversely affect the accuracy of the estimated PORs if the computed mapping function is valid for a single head pose alone. Indeed, plotting the calculated on-screen iris and head displacements performed by participant 7 during the validation procedure in Figure 5, reveals that the users naturally tend to perform combined eyeball and head movement during tracking in order to move the mouse cursor position close to the validation targets. The low quantitative error obtained by the proposed method may also be corroborated qualitatively as shown in Figure 7, displaying a small distance between the estimated PORs on the monitor screen and the ground truth validation targets. If the average on-screen icon is taken to have an average size of 45×45 pixels, the mean error that is achieved through the proposed method can be considered to be within the footprint of the average on-screen icon and therefore applicable to an HCI scenario.

A second validation procedure aimed to evaluate the accuracy of head movement approximation, by requesting the user to perform horizontal and vertical head rotations while maintaining the gaze fixed upon an on-screen validation target. Visually, the mouse cursor was expected to maintain a fixed on-screen position close to the validation target during this experimental procedure. In response to the head rotations performed during this validation stage, the mouse cursor was noticed to displace in the direction of the head movement and promptly return to its starting position in response to a counter-rotation of the eyeball without a change in the user's gaze. This resulted in mouse cursor trajectories as shown in Figure 8, for three horizontal and vertical head rotations performed by participant 7. As shown in Figure 9, the return positions of the mouse cursor corresponding to a total of five different head rotations performed by the participant were close to the ground truth validation target, hence ensuring that a fixed POR was also maintained on the monitor screen. Plotting the calculated on-screen iris and head displacements for these head rotations in Figure 10, together with the resulting mouse cursor coordinates, reveals that at every head rotation the estimated iris and head displacements reach equivalent amplitudes in opposite directions in agreement with Equation (7). The MAE and SD values across all participants, presented in Table IV in pixels and visual angle calculated at a distance of 60 cm from the monitor screen, reveal low error values between the return positions of the mouse cursor and the validation target. These results indicate the effectiveness of the head pose compensation algorithm in maintaining a consistent POR corresponding to a fixed gaze point in the presence of head rotations.

TABLE IV. MEAN ABSOLUTE ERROR (MAE) AND STANDARD DEVIATION (SD) IN PIXELS AND VISUAL ANGLE, OF THE ESTIMATED ON-SCREEN POR COORDINATES DURING ARBITRARY HEAD ROTATIONS.

Participant Number	x (MAE (pix), SD (pix))	y (MAE (pix), SD (pix))	x (MAE (°), SD (°))	y (MAE (°), SD (°))
1	(45.30, 59.53)	(64.02, 19.20)	(1.08, 1.42)	(1.53, 0.46)
2	(36.56, 11.07)	(16.52, 9.70)	(0.87, 0.26)	(0.39, 0.23)
3	(13.27, 7.07)	(24.96, 16.87)	(0.32, 0.17)	(0.60, 0.40)
4	(48.97, 9.66)	(12.28, 4.53)	(1.17, 0.23)	(0.29, 0.11)
5	(16.93, 8.85)	(10.33, 6.91)	(0.40, 0.21)	(0.25, 0.16)
6	(13.06, 4.65)	(21.44, 5.21)	(0.31, 0.11)	(0.51, 0.12)
7	(29.02, 2.46)	(24.13, 3.05)	(0.69, 0.06)	(0.58, 0.07)
8	(42.24, 23.47)	(28.01, 10.23)	(1.01, 0.56)	(0.67, 0.24)
9	(17.37, 6.20)	(23.12, 13.28)	(0.41, 0.15)	(0.55, 0.32)
Mean	(29.19, 14.77)	(24.98, 9.89)	(0.70, 0.35)	(0.60, 0.24)

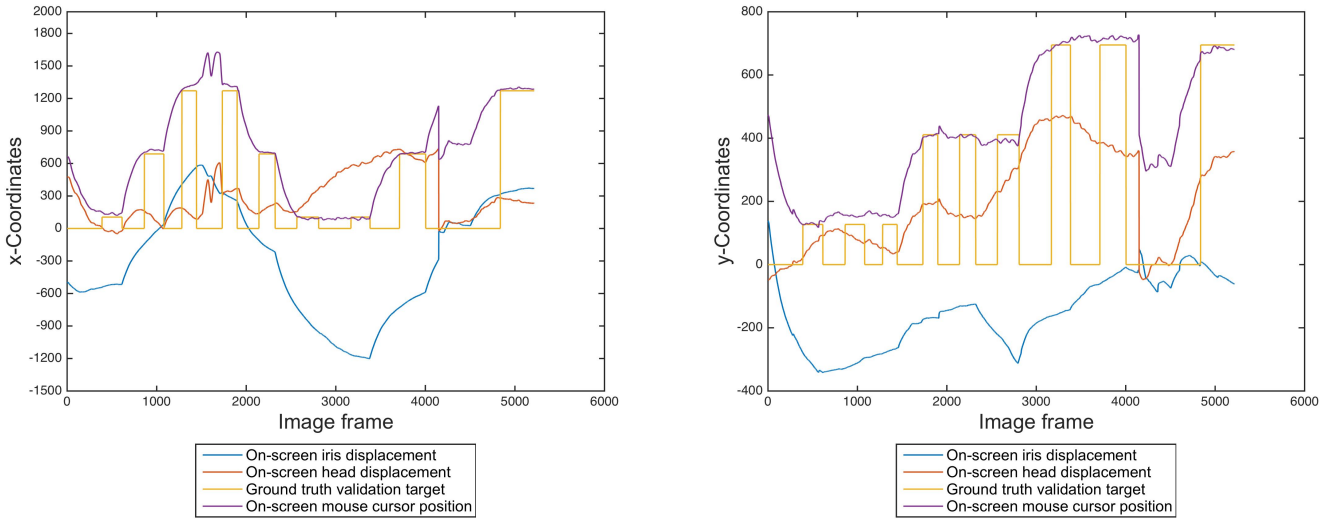
IV. COMPARISON WITH THE STATE-OF-THE-ART

In order to put the results achieved by the proposed method into context, a comparison with the results reported by other state-of-the-art methods [9][15]–[17][21][24][28] has been provided in Table V. Relevant state-of-the-art methods that estimate a POR on a monitor screen under conditions similar to ours have been chosen. These include methods that allow free head movement [9][16][17][21][24][28] and utilize generic imaging hardware, such as webcams [9][15][17][21][24], for image frame acquisition. Table V presents a comparison of gaze estimation results in visual angle, specifying the number of subjects considered during data collection, the number of calibration points required for POR estimation, the range of head rotation angles performed by the participants, and the distance at which the participants sit from the camera and visual targets.

The results in Table V reveal that our method achieves a POR estimation performance that is comparable to [21][1] or better than the results reported by relevant state-of-the-art methods in [9][15]–[17][21][24][28]. It is also worth noting that our method achieves this performance by employing fewer calibration points than the methods of [15]–[17][21][24][28], in the absence of training prior to POR estimation unlike the method of [16] and without constraining the user to a stationary head pose by the use of a chin-rest unlike the method of [15]. These characteristics allow for a method that addresses the challenges associated with pervasive eye-gaze tracking in less constrained conditions, by estimating a POR on a monitor screen from images acquired by an integrated webcam inside a notebook computer, under natural head movement, following a brief calibration procedure that does not require prolonged user-cooperation.

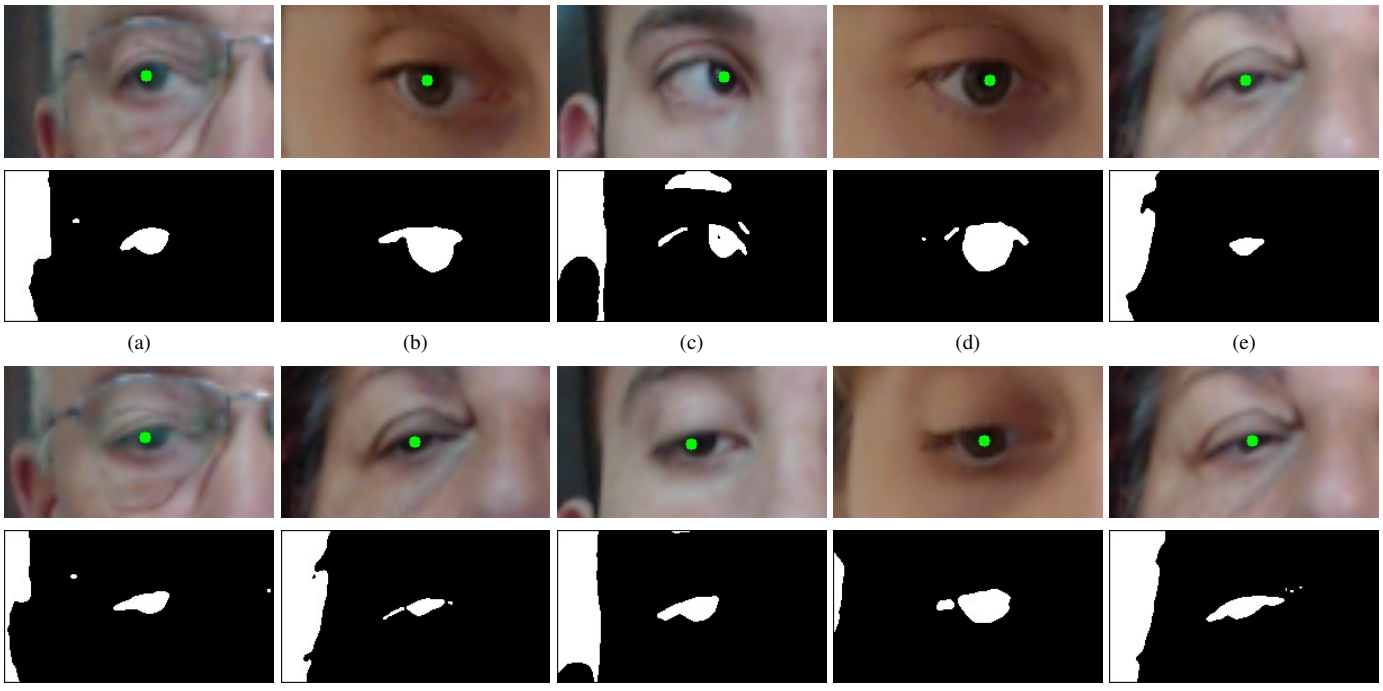
V. CONCLUSION

In this paper, we have proposed a passive eye-gaze tracking method to estimate the POR on a monitor screen from low-quality image data acquired by an integrated camera inside a notebook computer, while approximating any head rotations performed by the user during tracking. Following eye



(a) (b)

Figure 5. Calculated on-screen iris and head displacements, with respect to reference head and eye poses, during horizontal (a) and vertical (b) head rotations performed by participant 7 throughout the first validation procedure.



(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Figure 6. The proposed method for iris region segmentation was capable of localizing the iris center coordinates marked in green, at different eyeball rotations (a-e) and under partial occlusion of the iris region by the eyelids (f-j).

TABLE V. COMPARISON OF THE MEAN ABSOLUTE ERROR (MAE) AND STANDARD DEVIATION (SD) RESULTS OBTAINED BY OUR METHOD WITH RESULTS REPORTED BY RELEVANT STATE-OF-THE-ART METHODS. [1,2,3,4] REFER TO RESULTS REPORTED BY THE SAME METHOD, EMPLOYING DIFFERENT CALIBRATION PATTERNS CONTAINING VARYING AMOUNTS OF CALIBRATION POINTS.

Method	Yaw (MAE/°, SD/°)	Pitch (MAE/°, SD/°)	Number of Subjects	Number of Calibration Points	Range of Head Rotation (Yaw/°, Pitch/°, Roll/°)	User-Camera Distance (cm)	User-Target Distance (cm)
Our method	(0.54, 0.24)	(0.42, 0.20)	9	6	(±15, ±15, 0)	60	60
[9]	1.9	2.2	11	5	N/A	75	75
[15]	2.60 (±3.43)	2.61 (±2.45)	5	48	None	75	75
[16]	1.90		4	4000 training samples/user	N/A	N/A	N/A
[17]	3.55 (±0.42)		11	16	N/A	58.42 (±6.98)	58.42 (±6.98)
[21][1]	0.59 (±0.38)		7	33	N/A	50 - 60	50 - 60
[21][2]	0.63 (±0.39)		7	23	N/A	50 - 60	50 - 60
[21][3]	0.69 (±0.41)		7	18	N/A	50 - 60	50 - 60
[21][4]	0.97 (±0.57)		7	9	N/A	50 - 60	50 - 60
[24]	5.06		N/A	12	(⩽21.2, ⩽9.75, 0)	N/A	N/A
[28]	5.3	7.7	5	Arbitrary head rotations	N/A	220	240

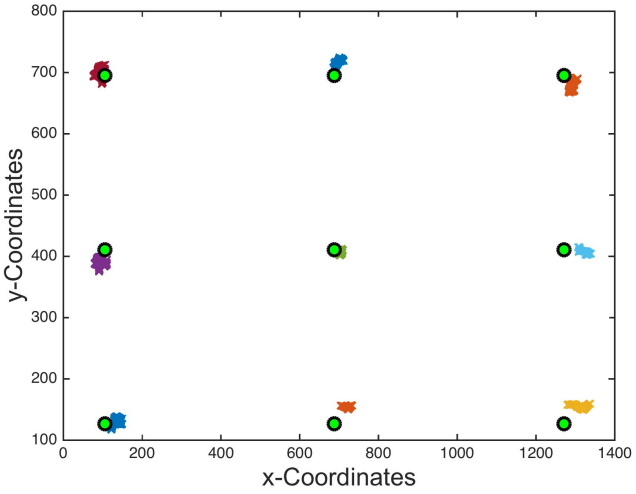


Figure 7. Validation result for participant 7, showing the displayed visual stimuli (green) and the estimated on-screen PORs (colored).

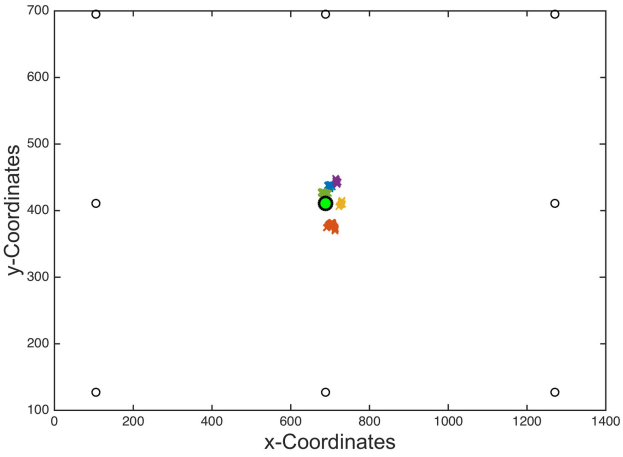


Figure 9. Return positions of the mouse cursor close to the ground truth validation target, corresponding to a total of five different head rotations performed by participant 7.

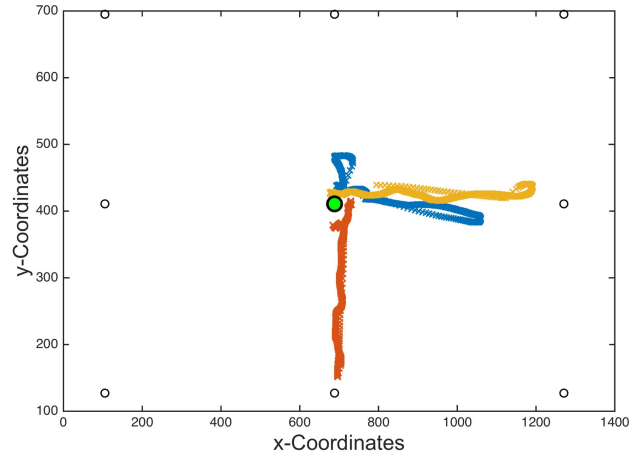
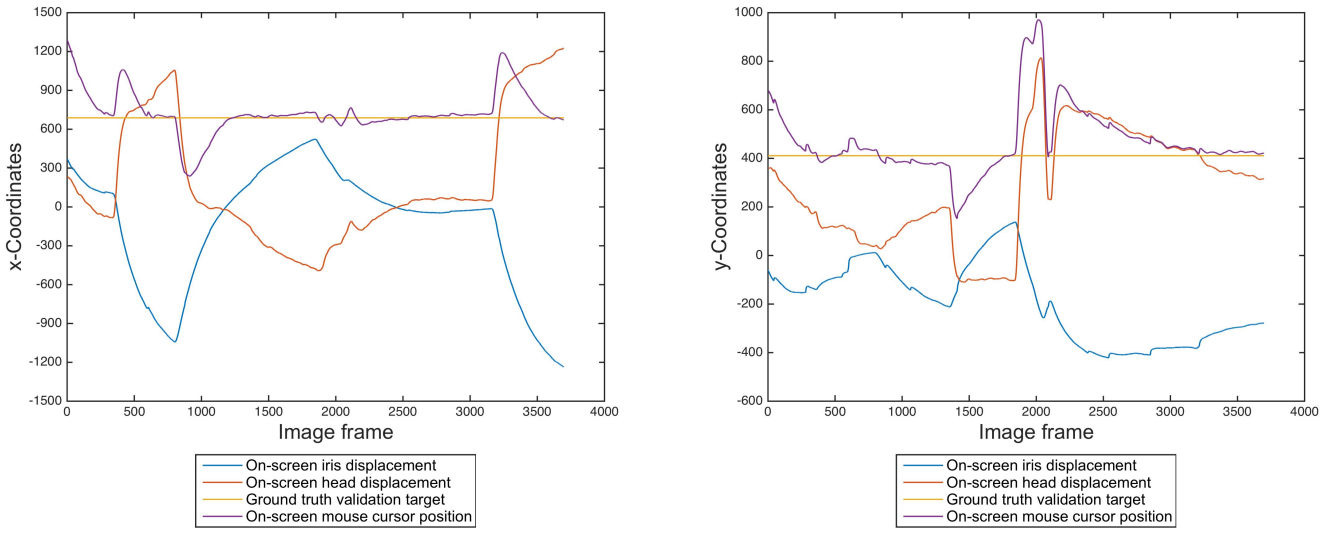


Figure 8. Mouse cursor trajectories in response to three of five head rotations performed by participant 7, keeping a fixed gaze point.

region detection and tracking, we proposed an appearance-based method which allows the localization of the iris center coordinates from low-resolution eye region images. In view of the constraints imposed when working on a computer, approximate linear mapping functions for the iris and head displacement inside the image space were computed during a calibration procedure, in order to map the image information to a POR on the monitor screen. The POR estimates, computed as a function of the iris and head movement, were improved by Kalman filters to smoothen the mouse cursor trajectory.

Two validation procedures were carried out in order to evaluate the proposed POR estimation method with head movement approximation. The experimental results for the first validation procedure revealed a noticeable discrepancy between the error in the x- and y-directions of the estimated PORs, with the error in the x-direction being the dominant between the two. The source of this error was observed to relate to incorrect segmentation of pixels belonging to artifacts, such as the eyelashes, along with the iris region producing a



(a) (b)

Figure 10. Calculated on-screen iris and head displacements, with respect to reference head and eye poses, having the same amplitude in opposite directions, during horizontal (a) and vertical (b) head rotations maintaining a fixed gaze point.

horizontal shift away from the true iris center. Nonetheless, the proposed method for iris region segmentation was capable of detecting and hence following the displacement of the iris region under partial occlusion by the eyelids, permitting the estimation of the POR in less than ideal conditions. It is noteworthy to mention that the proposed method achieved a low mean MAE of (0.54°, 0.42°), a significant improvement over our previous result of (1.46°, 0.71°) obtained through our earlier work without head movement compensation [1].

The second validation procedure aimed to evaluate the accuracy of head movement approximation. In response to horizontal and vertical head rotations without a change in gaze, the mouse cursor was noticed to displace in the direction of the head movement and promptly return to its starting position, hence retaining a fixed POR on the monitor screen. A low mean MAE of (0.70°, 0.60°) was achieved across all participants, indicating a small distance between the return positions of the mouse cursor and the validation target.

Future work aims to consider the non-linearities associated with eyeball and head rotations, approximated by planar image displacement of the respective iris center and head region in this work, in order to permit a wider range of eyeball and head rotations during tracking.

ACKNOWLEDGMENT

This work forms part of the project *Eye-Communicate* funded by the Malta Council for Science and Technology through the National Research & Innovation Programme (2012) under Research Grant No. R&I-2012-057.

REFERENCES

[1] S. Cristina and K. P. Camilleri, "Cursor control by point-of-regard estimation for a computer with integrated webcam," in The 8th International Conference on Advanced Engineering Computing and Applications in Sciences (ADVCOMP), 2014, pp. 126–131.

[2] R. J. K. Jacob, "What you look at is what you get: eye movement-based interaction techniques," in Proceedings of the SIGCHI conference on Human factors in computing systems: Empowering people, 1990, pp. 11–18.

[3] J. L. Levine, "An eye-controlled computer," in IBM Thomas J. Watson Research Center Res. Rep. RC-8857, Yorktown Heights, N.Y., 1981.

[4] S. J. Lee, J. Jo, H. G. Jung, K. R. Park, and J. Kim, "Real-Time Gaze Estimator Based on Driver's Head Orientation for Forward Collision Warning System," IEEE Transactions on Intelligent Transportation Systems, vol. 12, 2011, pp. 254–267.

[5] M. Shahid, T. Nawaz, and H. A. Habib, "Eye-Gaze and Augmented Reality Framework for Driver Assistance," Life Science Journal, vol. 10, 2013, pp. 1571–1578.

[6] A. Bulling, A. T. Duchowski, and P. Majaranta, "PETMEI 2011: The 1st International Workshop on Pervasive Eye Tracking and Mobile Eye-Based Interaction," in Proceedings of the 13th International Conference on Ubiquitous Computing: UbiComp 2011, 2011.

[7] D. W. Hansen and Q. Ji, "In the Eye of the Beholder: A Survey of Models for Eyes and Gaze," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, 2010, pp. 478–500.

[8] R. Valenti, A. Lablack, N. Sebe, C. Djeraba, and T. Gevers, "Visual Gaze Estimation by Joint Head and Eye Information," in 20th International Conference on Pattern Recognition, Aug. 2010, pp. 3870–3873.

[9] R. Valenti, N. Sebe, and T. Gevers, "Combining Head Pose and Eye Location Information for Gaze Estimation," IEEE Transactions on Image Processing, vol. 21, 2012, pp. 802–815.

[10] W. Haiyuan, Y. Kitagawaa, and T. Wada, "Tracking Iris Contour with a 3D Eye-Model for Gaze Estimation," in Proceedings of the 8th Asian Conference on Computer Vision, Nov. 2007, pp. 688–697.

[11] T. Moriyama, T. Kanade, J. Xiao, and J. F. Cohn, "Meticulously Detailed Eye Region Model and Its Application to Analysis of Facial Images," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, 2006, pp. 738–752.

[12] F. Timm and E. Barth, "Accurate Eye Centre Localisation by Means of Gradients," in Proceedings of the Sixth International Conference on Computer Vision Theory and Applications, Mar. 2011, pp. 125–130.

[13] E. G. Dehkordi, M. Mahlouji, and H. E. Komleh, "Human Eye Tracking Using Particle Filters," International Journal of Computer Science Issues, vol. 10, 2013, pp. 107–115.

[14] J. Mansanet, A. Albiol, R. Paredes, J. M. Mossi, and A. Albiol, "Estimating Point of Regard with a Consumer Camera at a Distance," Pattern Recognition and Image Analysis, 2013, pp. 881–888.

[15] W. Sewell and O. Komogortsev, "Real-time eye gaze tracking with an unmodified commodity webcam employing a neural network," in Proceedings of the 28th International Conference Extended Abstracts on Human Factors in Computing Systems, 2010, pp. 3739–3744.

[16] R. Stiefelhagen, J. Yang, and A. Waibel, "Tracking Eyes and Monitoring Eye Gaze," in Proceedings of the Workshop on Perceptual User Interfaces, Oct. 1997, pp. 98–100.

[17] C. Holland and O. Komogortsev, "Eye Tracking on Unmodified Common Tablets: Challenges and Solutions," in Proceedings of the Symposium on Eye Tracking Research and Applications, Mar. 2012, pp. 277–280.

[18] K. Kunze, S. Ishimaru, Y. Utsumi, and K. Kise, "My Reading Life - Towards Utilizing Eyetracking on Unmodified Tablets and Phones," in Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing, 2013, pp. 283–286.

[19] W. Z. Zhang, Z. C. Wang, J. K. Xu, and X. Y. Cong, "A method of gaze direction estimation considering head posture," International Journal of Signal Processing, Image Processing and Pattern Recognition, vol. 6, no. 2, 2013, pp. 103–111.

[20] E. Demjen, V. Abosi, and Z. Tomori, "Eye tracking using artificial neural networks for human computer interaction," Physiological Research, vol. 60, 2011, pp. 841–844.

[21] T. Okabe, F. Lu, Y. Sugano, and Y. Sato, "Adaptive linear regression for appearance-based gaze estimation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 36, no. 10, 2014, pp. 2033–2046.

[22] K. Kim and R. S. Ramakrishna, "Vision-based eye-gaze tracking for human computer interface," in IEEE International Conference on Systems, Man, and Cybernetics, vol. 2, 1999, pp. 324–329.

[23] N. M. Piratla and A. P. Jayasumana, "A neural network based real-time gaze tracker," Journal of Network and Computer Applications, 2002, pp. 179–196.

[24] H. Wang, C. Pan, and C. Chaillou, "Tracking eye gaze under coordinated head rotations with an ordinary camera," in 9th Asian Conference on Computer Vision, vol. 5995, 2010, pp. 120–129.

[25] R. Ronsse, O. White, and P. Lefevre, "Computation of gaze orientation under unrestrained head movements," Journal of Neuroscience Methods, vol. 159, no. 1, 2007, pp. 158–169.

[26] B. L. Nguyen, Y. Chahir, M. Molina, C. Tijus, and F. Jouen, "Eye gaze tracking with free head movements using a single camera," in Symposium on Information and Communication Technology, 2010, pp. 108–113.

[27] R. Dhyawala, J. Dhariwal, and N. Nain, "Eye gazing with low resolution web-cam images using artificial neural network," in Proceedings of the International Conference on Advances in Computer, Electronics and Electrical Engineering, 2012, pp. 443–447.

[28] H. Yamazoe, A. Utsumi, T. Yonezawa, and S. Abe, "Remote gaze estimation with a single camera based on facial-feature tracking without special calibration actions," in Proceedings of the 2008 Symposium on Eye Tracking Research and Applications, 2008, pp. 245–250.

[29] Y. Matsumoto and A. Zelinsky, "An algorithm for real-time stereo vision implementation of head pose and gaze direction measurement," in Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition, 2000, pp. 499–504.

[30] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Jul. 2001, pp. 1231–1238.

[31] "OpenCV," 2014, URL: http://http://opencv.org/ [accessed: 2014-04-21].

[32] V. Vezhnevets, V. Sazonov, and A. Andreeva, "A Survey on Pixel-Based Skin Color Detection Techniques," in Proceedings of the GraphiCon, 2003, pp. 85–92.

[33] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," Journal of Basic Engineering, 1960, pp. 35–45.