Building a Personalized, Auto-Calibrating Eye Tracker from User Interactions

Michael Xuelin Huang, Tiffany C.K. Kwok, Grace Ngai, Stephen C.F. Chan, Hong Va Leong
Department of Computing, Hong Kong Polytechnic University
Hong Kong, China
{csxhuang,csckkwok,csgngai,csschan,cshleong}@comp.polyu.edu.hk

ABSTRACT
We present PACE, a Personalized, Automatically Calibrating Eye-tracking system that identifies and collects data unobtrusively from user interaction events on standard computing systems without the need for specialized equipment. PACE relies on eye/facial analysis of webcam data based on a set of robust geometric gaze features and a two-layer data validation mechanism to identify good training samples from daily interaction data. The design of the system is founded on an in-depth investigation of the relationship between gaze patterns and interaction cues, and takes into consideration user preferences and habits. The result is an adaptive, data-driven approach that continuously recalibrates, adapts and improves with additional use.

Quantitative evaluation on 31 subjects across different interaction behaviors shows that training instances identified by the PACE data collection have higher gaze point-interaction cue consistency than those identified by conventional approaches. An *in-situ* study using real-life tasks on a diverse set of interactive applications demonstrates that the PACE gaze estimation achieves an average error of 2.56º, which is comparable to state-of-the-art, but without the need for explicit training or calibration. This demonstrates the effectiveness of both the gaze estimation method and the corresponding data collection mechanism.

Author Keywords
Gaze estimation; implicit modeling; data validation; gaze-interaction correspondence

ACM Classification Keywords
H.1.2 [Models and Principles]: User/Machine Systems-Human factors; I.5.m [Pattern Recognition]: Miscellaneous

INTRODUCTION

Gaze information, as a reflection of human attention and cognition, has great potential applications in a large number of domains, including diagnostics, crowdsourcing, market research and education. The potential of gaze-aware systems for daily human-computer interaction and social interaction has increased with the growing pervasiveness of camera systems. Therefore, there has been, and continues to be, much research in gaze estimation [8].

With few exceptions, most gaze estimation methods require calibration and non-periodical re-calibration to accommodate lighting and head pose variances [26], which is obviously cumbersome and inconvenient in real use [22]. Since there is likely a strong correlation between eye gaze and interaction cues, such as cursor and caret locations, it seems to make sense that the mapping between gaze features and the gaze point can be collected implicitly from normal computer interactions and used to recalibrate or retrain gaze estimation models. However, while a number of studies have demonstrated a correlation between gaze and cursor [10][15], there have been few efforts in using noisy daily interaction data for webcam gaze learning. One notable exception is that of Sugano et al. [22], which collects mouse clicks for incremental gaze learning.

Given the massive amounts of time spent interacting with the computer, we believe it makes sense to explore the use of regular human computer interactions with mouse and keyboard for gaze learning. This can potentially make available a much larger amount of data, which accelerates the learning process and improves performance. Specifically, we are interested in data that can be obtained unobtrusively using conventional off-the-shelf equipment.

Most previous work [6][13][22] makes the assumption that users are looking at where they click, or, to put it more broadly, that users are looking at the interaction cue at the moment that the interaction is triggered. However, this assumption may be not valid in real-use situations, due to factors such as eye blink, mind-absence, response delay, individuality and task difference. We therefore propose to apply behavior-informed and data-driven approaches to identify reliable training instances from daily-use interaction data and webcam video.

To the best of our knowledge, there is no prior work that seeks to automatically identify and validate noisy interaction and webcam video data for gaze model learning. The exception is Huang et al [12], which addresses gaze-

related feature extraction and tentative data validation, but with more restrictive and rigid constraints.

This paper presents PACE, a Personalized, Automatically Calibrating Eye-tracking system, which is designed to enable high-performing gaze estimation in standard computing systems with no additional equipment. We (1) conduct an in-depth experimental study to investigate and quantify the gaze-interaction consistency across different behaviors; (2) propose an unobtrusive, adaptive, interaction-informed method that identifies the gaze-interaction alignment in daily computer use; and (3) demonstrate the effectiveness of PACE in multi-person evaluations across diverse interactive tasks.

RELATED WORK

The best-performing gaze estimation systems are model-based approaches [28], which use specialized equipment such as multiple cameras, light sources, or infrared, but the costs and equipment requirements make them infeasible for wide-range application on a large scale. In contrast, appearance-based approaches [1][16][18][21][23][24] that estimate gaze from single images are more versatile and easily deployed. However, they are sensitive to noise from lighting conditions, head pose, and individuality. Hansen and Ji [8] present an extensive review of such approaches.

Generalizing From Limited Data

Making good use of limited data is key to improving performance while reducing the calibration effort. Williams et al. [23] developed a semi-supervised method to use unlabeled calibration data. Their method requires users to follow an animated spot on the screen. Lu et al. [16] introduced compensations to correct the head pose biases for gaze estimation. Their calibration requires the user to rotate his/her head while fixating on each calibration point. To reduce the amount of calibration needed, Lu et al. [17] synthesized training samples for unseen head poses from multiple reference images where the user's head position changes while the eye rotation is held constant.

Although these methods reduce the amount of total calibration data, they all require a tedious, explicit initialization procedure. In addition, human error during calibration, such as eye blinks [18] or distracted saccades, may cause unexpected performance drop. It is not difficult to see that a method that implicitly collects good training data can also be used in conjunction with the above approaches to further improve gaze modeling.

Implicit Collection of Incremental Data

Some approaches bypass the cumbersome and lengthy calibration phase by implicitly collecting data from daily computer usages. One popular solution uses a saliency model that assumes that the user is more likely to look at the salient region of an image or video frame. Sugano et al. [21] applied the saliency map of video frames to estimate gaze based on images captured by a monocular camera. The problem with this approach is that the consistency between image saliency and real gaze location is often influenced by the attributes of visual stimuli, such as complexity and semantics. Apart from these uncertainties, the computation saliency models often do not match the actual human gaze movement [14]. Alnajar et al. [1] therefore made a different assumption that infers the calibration of a new user based on previously-collected gaze data from a group of individuals. This method makes use of inter-personal similarity for visual attention. However, the correlation between visual attention and image conspicuity is also affected by differences between individuals.

There has been some work into adopting interaction information to facilitate gaze learning. Hornof et al. [9] suggested a strategy that looks for interactions with known fixation points for run-time recalibration of the eye tracking model. Zhang et al. [27] further identified probable fixation locations to account for instances that cannot be clearly mapped to a known fixation point. Their work, however, relies on an infrared eye tracker to detect eye fixations, and knowledge of the visual context, including target locations and layout irregularity.

Sugano et al. [22] proposed an alternative model which collected mouse click points as ground truth data to incrementally update the gaze model. Jacob [13] used click data to correct gaze tracking results. Similarly, Fares et al. [6] proposed to use mouse-click data as dynamic local calibration data. These approaches all assume that the location of the click point is the location of the user's gaze. However, in unconstrained real-use situations, this assumption may not always hold.

Investigating Gaze-Cursor Correlation

Gaze-cursor consistency is a perennially popular topic of study, especially for web browsing behaviors. Chen el al. [4] suggested that there is a strong correlation between gaze and saccade-like mouse movement. Rodden et al. [19] reported strong alignments between gaze and cursor during active mouse usages, including using the cursor as a reading aid (in both horizontal and vertical directions) and to mark particular results. Guo et al. [7] proposed a set of mouse features to identify the moments with strong gaze and cursor alignment during browsing. They achieved an average accuracy of 77%, 3% higher than the baseline. Liebling et al. [15] showed that gaze and mouse coordination contain complex and nuanced characteristics in real-life scenarios. Huang et al. [11] found that there is a certain correlation between gaze and cursor, but with substantial variation whereby the distance between the eye gaze and the mouse click location is smallest one second before the click occurs for one third of the subjects, in a "cursor lags behind gaze" phenomenon [10]. These findings suggest that the conventional hypothesis that "gaze is well approximated by cursor" may be naïve. It also shows that temporal alignments varied significantly across individuals, which argues for a personalized approach.

Interaction event	Human intention	*Potential* gaze pattern
Mouse click	Link or button selection	Fixation on the mouse cursor
Mouse double-click	Word selection in document editing	Fixation on the mouse cursor
Mouse button up after drag	Paragraph selection in document editing	Fixation on the mouse cursor
Keyboard letter key down	Word typing	Fixation/smooth pursuit on the typing caret

Table 1. Examples of gaze patterns from common interaction behaviors.

INVESTIGATING CONSISTENCY BETWEEN GAZE AND INTERACTION

In order for PACE to be feasibly integrated into real-world computing systems, the gaze model must be robust to natural head movement, not require explicit calibration, and not require the use of specialized equipment. Our approach is therefore to unobtrusively identify and collect training instances from daily interaction data. We start by evaluating the assumed correspondence between gaze and interaction from previous work [6][13][22]. Our findings will guide the identification of reliable data for gaze learning.

We hypothesize that the moment of strongest gaze-interaction correlation depends on the nature of the interaction, the context, user habits and preferences. Some activities require a more explicit demonstration of human intention, which results in a higher gaze-cursor consistency. For example, the positions of the eye gaze and the mouse cursor are better aligned during a mouse click event, compared to when the mouse cursor is being used as a reading aid [10]. The context also affects the correlation. It is easy to see that clicking to select a single character in a paragraph of text would require a more purposeful and precise gaze than clicking on photo thumbnails to scroll through photo albums. Some tasks may actually require or encourage the user to look at a part of the screen away from the cursor, e.g. pausing the playback of a video at a precise moment during video editing.

Experiment Setup

Our investigation focuses on four commonly encountered interaction activities, as shown in Table 1. We use the Tobii EyeX tracker to provide us with the "ground truth" of the user's gaze point. The Tobii tracker uses infra-red technology with 60 Hz tracking frequency and an accuracy within 1 ^{o}visual angle (corresponding to 30-50 pixels on our 22" monitor at 1680×1050 resolution and a reading distance ≈500~800mm), and can be considered to be state-of-the-art.

We recruited 31 subjects (16 female, ages 20-30 yrs, mean 25.1, standard deviation 2.5) for this study. The subjects were university students and staff. 24 of them are capable of touch-typing, at least for the letter keys.

Subjects were asked to work naturally, which meant free movement of head and body. They were allowed to change the chair position and height, but to keep their head-to-monitor distance within the valid range (≈500~800mm) of the monitor. Three experiments were designed to generate the necessary interaction behaviors:

Correlation between visual attention and click targets: The first experiment requires the subject to click on targets of different shapes and sizes. Observations of common mouse usage patterns show that mouse clicks usually involve (1) long slim targets, (2) small targets and (3) large targets.

To obtain data on (1) and (2), a list of academic papers with long titles (≥ 3/4 of the screen width) was prepared in advance. The subjects were asked to search for each paper on Google Scholar, and to click on the hyperlinks for the title and authors for each paper. Since Google abbreviates authors' first names, author hyperlinks are usually short and button-like (small targets), and the hyperlinks with the paper titles give long slim targets. The large targets were obtained by asking the subjects to search for and click on photos in Flickr that they found interesting. The width of the photos occupied around one third of the screen.

Correlation between visual attention and mouse drag actions: The second experiment considers a different kind of mouse activity – dragging. Subjects were asked to select sentences from a given PDF document by dragging the mouse. To ensure that they would actually pay attention to what they were doing, they were required to select complete sentences or phrases (ending with a period or comma.)

Correlation between visual attention and keyboard usage: For the third experiment, subjects were asked to type a short paragraph into a text file. They could type anything they wanted, as long as it was a syntactically correct paragraph that made sense semantically. We collected only letter keys ("a"-"z", "A"-"Z", spaces), because most people, even those who can touch-type, are actually only able to touch type the letter keys, not the number or function keys.

Each of the experiment subjects was required to generate at least 50 instances of each interaction activity. The occurrence of a clicking event was defined as the press of the left mouse button, dragging events as the release of the left mouse button, and keypresses as the depression of a letter key. Each interaction event triggered a screenshot that was saved for data validation and event classification. In total, we collected 1915 clicking events on long slim targets, 2344 on small targets, 1955 on large targets, 2029 dragging events and 4863 typing events.

Evaluation of the Correspondence Assumption

To investigate the correlation between visual attention and interaction event, we study the 3 seconds of data from the Tobii tracker *preceding* each interaction event. The focus on pre-interaction behavior is informed by previous work

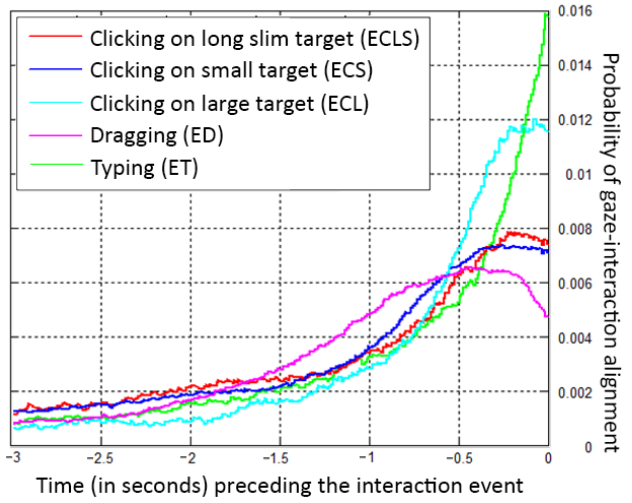

Figure 1. Probability of gaze-interaction alignment – i.e. the likelihood that visual attention is spatially located at the interaction event, as a function of time *preceding* the event.

[10], which reports that in general, the position of the cursor lags behind the gaze point, not the other way around.

For simplicity, we will use the following abbreviations when referring to events of: clicking on small targets (ECS), long slim targets (ECLS), large targets (ECL), dragging (ED) and typing (ET).

The eye tracker returns the position of the user's gaze on the screen as a temporal sequence of (*x*, *y*) screen coordinates. To allow for inherent error from the eye tracker, we choose a small distance threshold γ (=60 pixels), which matches the equipment error. The position of the user's gaze and the location of the interaction event are considered to be *aligned* when their *displacement*, or the distance between them, is less than this threshold, *i.e.* $D(\boldsymbol{g}_t(t_p), \boldsymbol{c}) < \gamma$, where $D(\boldsymbol{g}_t(t_p), \boldsymbol{c})$ indicates the Euclidean distance between the tracker-measured gaze point $\boldsymbol{g}_t$ and the interaction point $\boldsymbol{c}$, and t_p represents the time preceding an interaction event.

Figure 1 shows the probability of *gaze-interaction* (i.e. gaze-cursor or gaze-caret) *alignment* as we approach the moment of interaction, *i.e.* $Pr(D(\boldsymbol{g}_t(t_p), \boldsymbol{c}) < \gamma)$. The *x*-axis shows the time in seconds *before* the interaction event. As expected, the probability of the gaze-interaction consistency generally increases as we get closer to the interaction event. However, the moment of highest probability for gaze-interaction alignment does not necessarily occur at the moment of the interaction. For example, the likelihood distribution of gaze-interaction alignment peaks at t_p=-0.01s for ET, -0.07s for ECL, -0.20s for ECLS, -0.25s for ECS, and -0.43s for ED. In particular, for mouse drag events (ED), the probability of gaze-interaction alignment *falls off significantly* in the moments just before the interaction event happens. It would seem that typing events (ET) are the only ones in which the assumption that "the user is looking where he/she is interacting" is consistently upheld.

Figure 2 presents the *gaze-interaction displacement* *i.e.* $D(\boldsymbol{g}_t(t_p), \boldsymbol{c})$ or the distance between the location of the user's gaze and the eventual location of the interaction event. It can be seen that the displacement (mainly: the grey and blue regions) generally decreases as we get closer to the time of the interaction event, but the distributions are quite dissimilar. Unsurprisingly, the displacement is largest for ECL (clicking on large targets). However, the interaction activity with the second largest displacement is ET (typing), which also has a large range of displacement values (wide blue and grey regions). This implies that even though

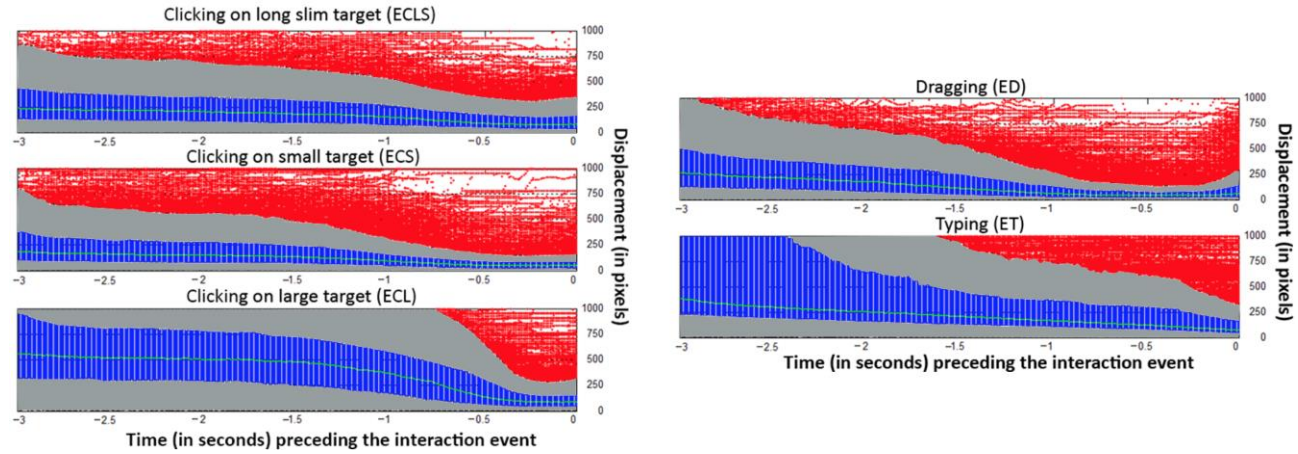

Figure 2. Displacement between gaze and interaction cues (cursor/caret) as a function of pre-interaction event time for different interaction activities. The *x*-axis indicates the time before the event, the *y*-axis shows Euclidean distance between location of event (i.e. where the mouse is actually clicked or where the character appears on the screen) and gaze coordinates collected by eye tracker. The green line shows the median distance. The blue region shows values that fall within $[\boldsymbol{p}_{25}, \boldsymbol{p}_{75}]$, *i.e.* the 25th and 75th percentiles. The grey area indicates the range of the data points that are not considered as outliers. The red points are the individual outliers, defined as values located beyond $[2.5\boldsymbol{p}_{25} - 1.5\boldsymbol{p}_{75},\ 2.5\boldsymbol{p}_{75} - 1.5\boldsymbol{p}_{25}]$.

Figure 1 suggests that people are *generally* looking at the caret when they type, there is still much variation across different events, and hence, using the raw typing-informed data for gaze model training would introduce much noise and error into the system.

The distribution of the outliers (the red regions) is also of interest. Figure 2 shows that there is much variation in the user's gaze. Furthermore, the *range* of these locations is very great, ranging to 1000 pixels away from the interaction position. This further corroborates our hypothesis that the correspondence assumption is not valid in real-use situations. This means that a naïve use of raw interactional-informed data for gaze model learning is not likely to produce optimal results.

Inspecting the non-outlier data reveals some interesting findings. For ED (dragging), a U-shape distribution starts to form around one second before the interaction event. This indicates that in most cases, subjects start looking elsewhere before the drag event is complete. A similar phenomenon happens during ECLS. This suggests that if we could identify the point of least displacement, this would potentially be a better indicator of eye gaze than simply collecting the data at the moment of the event.

Another interesting finding comes from the *densities* of the displacement distributions across the different behaviors. Although the *range* of the displacements, *i.e.* the upper boundaries of the grey region, is fairly large, 75% of the data stays within half of the range, as evidenced by the fact that the upper boundary of the blue regions lies close to the middle of the grey regions. Similarly, the green median line lies below the middle of the blue region for almost all behaviors, which indicates that the data is very compactly distributed. This is especially true for typing events (ET). This implies that the majority of the data exhibits strong gaze/interaction consistency, even in real-use situations.

In addition to the data analysis, observations of the subjects' behavior during the experiment and the post-experiment interviews also reveal some interesting insights and provide possible explanations for the data distribution.

Clicking: We observed two distinct ways in which people usually click on long slim links. Some people start moving the cursor towards a link only after reading its context and deciding to click on it. In this case, the user's gaze tends to stay close to the last word they read and that is usually also where they click. However, sometimes the user perceives that there is a high probability that a certain link would be relevant, even before reading it. In these cases, they often move the cursor to hover over the link before actually reading the words. Once they finish reading, they click the mouse without moving it again. In these cases, the displacement between gaze and click could be quite unpredictable, depending on where the cursor hovers and where the link ends.

Dragging: Dragging generally results in high consistency between gaze and interaction-informed data. Since the nature of the action requires precision, people are more likely to spend more time and care to ensure that the context selection is correct. However, by the time a drag has been completed, users may already be looking for the next target, such as the "highlight" button. This may be the main reason that the probability of gaze-cursor alignment drops as we get closer to the moment of the event as shown in Figure 1 and that the corresponding large displacement causes the U-shape distribution as shown in Figure 2. The context is also important. When the selected sentence ends a paragraph, the PDF viewing application automatically snaps the end of the drag to the end of the paragraph. Therefore, subjects are often more careless when selecting such sentences, thus creating a large distance displacement.

Typing: For users who can touch type, their gaze normally follows the caret, *i.e.* the location of the character being typed. However, we observed that when they are thinking hard, many (11 out of 24 in our case) touch-typers look elsewhere on the screen while continuing to type. For users with limited touch-typing skills, the gaze switches between the monitor and keyboard. Both of these behaviors create large displacement, and explain the presence of the large number of outliers for this interaction activity.

IDENTIFYING GOOD INTERACTION DATA FOR EYE TRACKING

Our behavior study suggests that interaction data corresponds with the gaze to a certain degree, which implies that it is feasible to use everyday user interactions as data to build a gaze tracking system. However, the findings also show that the moment of best alignment varies across different interaction activities, the context of each interaction, and the individuality of the users. This argues for a more refined approach to using interaction-informed data for training. It also suggests that a *personalized* approach would be the most effective; since it would automatically adapt to user preferences and constraints.

We hypothesize that it is possible to use *knowledge of common gaze patterns* and *analysis of visual signals* to identify the point at which user gaze and interaction event are most likely to be aligned. In other words, we postulate that there are periods when user gaze-interaction event alignment is likely, and it is possible to detect these periods in an automated fashion.

Overall System Flow and Methodology

Figure 3 gives an overview of our Personalized Automatically Calibrating Eye-tracking (PACE) approach. A standard webcam captures video of the user's head and shoulders while mouse movements and keystrokes are tracked by the system. Two tracking models extract the gaze feature vector $\boldsymbol{v}$ from face and eye landmarks identified in the frames of the video stream.

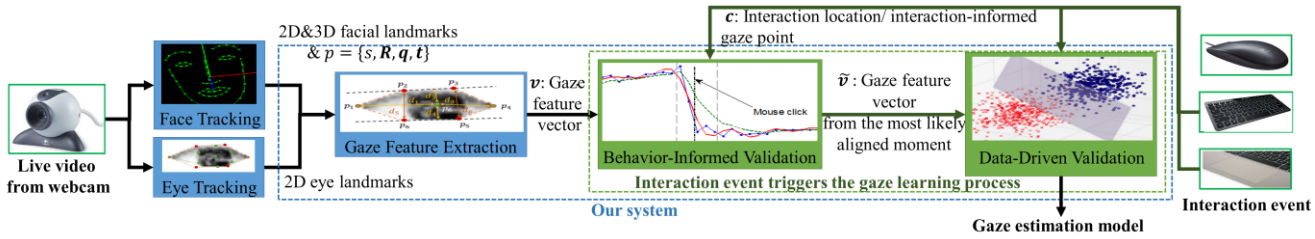

Figure 3. Overview of the PACE methodology: combining interaction data and webcam video for eye gaze modeling.

Upon the trigger of an interaction event $\boldsymbol{c}$, gaze feature vectors from the 3-second window preceding the interaction are sent to a *behavior-informed validation* engine and a *data-driven validation* engine. The behavior-informed validation engine selects one vector $\tilde{\boldsymbol{v}}$ that corresponds to the moment when the user's gaze is most likely to be aligned with the interaction event. The data-driven validation engine then further checks the validity of $\tilde{\boldsymbol{v}}$, based on the previous training samples. If $\tilde{\boldsymbol{v}}$ passes both of these validation steps, $[\tilde{\boldsymbol{v}}, \boldsymbol{c}]$ will be used as training data to update the gaze estimation model. Otherwise the data is retained for re-evaluation after the next update.

Extracting Gaze Features from Video

To obtain accurate locations of the facial landmarks and head pose information, we use Constrained Local Models (CLM) [20] to obtain 3D vertices of 66 facial landmarks. Procrustes alignment [5] is used to normalize the rigid transformation and Principal Component Analysis (PCA) to approximate the non-rigid deformation of the given face images. Supervised Descent Method (SDM) is then used to optimize 48 facial landmarks [25] to improve the localization accuracy and facilitate head pose estimation.

To model the user's eyes, we follow Huang et al. [12] and use facial landmarks on the iris contour and eyelid corners to calculate the location of the pupil center inside the eye image. This gives 3 eye features representing the eye yaw and pitch rotation, and the degree of openness of the eye.

The facial and eye features are combined into a 12-feature gaze vector $\boldsymbol{v} = [s, \boldsymbol{R}, \boldsymbol{t}, \boldsymbol{e}_r, \boldsymbol{e}_l]^T$, where $s, \boldsymbol{R}, \boldsymbol{t}$ are the head pose features and $\boldsymbol{e}_r, \boldsymbol{e}_l$ the features from right and left eyes, respectively.

Using Human Behavior to Inform Data Validation

Human gaze patterns can be categorized into four behaviors: *fixation*, *smooth pursuit*, *saccade* and *blink* [8]. Fixation indicates a stationary gaze. Smooth pursuit denotes relatively slow gaze movements, while saccades are eye movements that rapidly direct towards a stationary target.

Figure 4 shows the change in an example feature signal (eye yaw) around a mouse click. The gaze pattern contains 2 short fixations, 2 saccades and 1 smooth pursuit. The black dashed line indicates the moment of the mouse click. The figure shows that the user's gaze was originally focused on one point (1st fixation), then rapidly moved towards a short link (1st saccade), which took approximately 1 second to read (smooth pursuit). She then clicked on the link and her attention shifted (2nd saccade) elsewhere (2nd fixation). This behavior is consistent across multiple instances, with the mouse click usually occurring at the end of the smooth pursuit, the beginning of the 2nd saccade or even the beginning of the 2nd fixation. This clearly illustrates how the user's fixation is not always located at the point of the interaction event.

The behavior-informed validation in PACE identifies moments at which the user's gaze aligns with the cursor/caret when interaction activity is triggered. Based on knowledge of human gaze patterns and our observations, this is most likely during periods of fixation or smooth pursuit. For simplicity, we refer to these periods as $\boldsymbol{S}_s$ and the state of the gaze feature during these periods as being "stationary with small trend", or "stationary". The problem then is to determine $\boldsymbol{S}_s$ automatically from the webcam data.

Signal Smoothing and Filtering

Based on our previous findings, we focus on the user signals collected in the 3 seconds prior to the interaction event. Linear interpolation is used to resample the signal to 100Hz, giving us 300 samples per feature per interaction event. Since the raw webcam signals contain much high-frequency visual noise, a standard low-pass filter is used to remove some of the noise prior to analysis [2].

Figure 4 demonstrates the effects of filtering in the frequency and temporal domains. While both filtering methods remove much of the high-frequency noise, filtering in the temporal domain results in the loss of some critical information, such as the dynamic overshoot glissade that occurs before the 1st rapid saccade. We therefore filter in the frequency domain to remove the high frequency

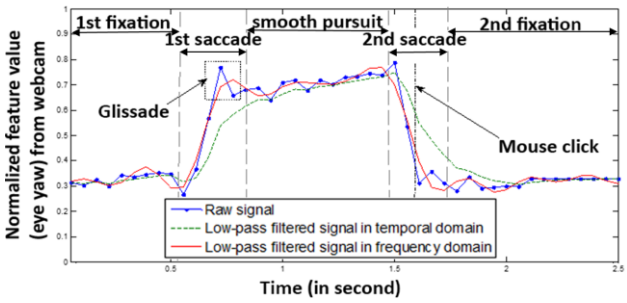

Figure 4. Raw and filtered webcam signals from a sample mouse click event (user is reading from left to right).

Input: Matrix $\boldsymbol{F}$ of gaze feature vectors from the 3-second window before the interaction event.	
Output: The stationary period $\boldsymbol{S}_s$ and the feature vector $\tilde{\boldsymbol{v}}$ at the most likely moment of gaze-interaction alignment.	
1	Calculate the low-pass frequency-domain filtered signal $\hat{\boldsymbol{f}}_j$ for each gaze feature signal $\boldsymbol{f}_j$
2	Iteratively search for the stationary period using an incremental threshold $\boldsymbol{\varepsilon} \in [\boldsymbol{r}/1000, \boldsymbol{r}/100]$, where $\boldsymbol{r} \in \Re^{n\times 1}$ is the range of each feature over the window.
3a	Initialize the overall stationary period with the frame index $\boldsymbol{S} = [1, \ldots, m]$.
3b	Calculate the overall stationary period $\boldsymbol{S}$ according to Equation (1); set S_i to 0 if the condition is not satisfied.
3c	Backward search $\boldsymbol{S}$ for the first series of consecutive frame indices $\boldsymbol{S}_s$ whose corresponding duration is longer than the minimum fixation duration (80ms) [8].
3d	If t_s, the last moment of $\boldsymbol{S}_s$, occurs within 0.5 seconds before the event, then break. Else increment $\boldsymbol{\varepsilon}$ by $\boldsymbol{r}/1000$.
	End
4	Perform line fitting for each feature signal $\hat{\boldsymbol{f}}_j$ in $\boldsymbol{S}_s$ to approximate the final gaze feature vector $\tilde{\boldsymbol{v}}$.

Table 2. Adaptively Extracting a Stationary Feature Vector.

temporal jitter while maintaining the shape of the main component with minimal distortion.

Extracting a stationary feature vector

To identify the stationary period $\boldsymbol{S}_s$ and the corresponding estimated feature vector $\tilde{\boldsymbol{v}}$, our approach adaptively searches for candidate periods occurring close to the point of the event that do not exhibit high signal variance.

To identify these candidate periods, PACE uses a novel adaptive method that searches for a *relatively stationary* period close to the event. The algorithm is similar to that of Huang et al [12], which uses a threshold to determine if the signal change is small enough to be considered "stationary", but their approach requires a truly fixed gaze and will fail if the user's gaze is not truly still, which would be problematic for real-world contexts where the eye is rarely truly stationary.

Table 2 presents our algorithm. Basically, we analyze the changes between consecutive frames in the 3-second window prior to the interaction event. An adaptive threshold $\boldsymbol{\varepsilon}$, which is based on the range of each feature, is used to identify candidate frames whose feature vectors satisfy the following condition:

$$\prod_{j=1}^{n} H(\varepsilon_j{}^2 - \dot{\hat{f}}_{ij}^2) = 1 \qquad (1)$$

where $H(x) = (1 + \mathrm{sgn}(x))/2$ is the Heaviside step function and $\dot{\hat{f}}_{ij}$ is the i^{th} frame value of the derivative of $\hat{\boldsymbol{f}}_j$ with respect to the sample time. These frames are considered to be potentially within the stationary period.

After all frames in the window have been tested, a backward search is performed to locate $\boldsymbol{S}_s$. Linear regression is used to approximate the gaze feature vector $\tilde{\boldsymbol{v}}$ corresponding to the last moment of $\boldsymbol{S}_s$, which under our assumptions is also the moment when the user's gaze is most likely to be aligned with the interaction event. If no fixation or smooth pursuit is detected at all during the 3-second time window, that interaction event is considered to be not suitable for training and is discarded.

Data-driven validation

The behavior-informed validation looks for the stationary period corresponding to the received interaction event. However, there is a possibility that even though the user's gaze is fixated on something, the gaze point may not be anywhere near the location of the interaction event – for example, when the user is watching a movie with the mouse pointer poised over the "pause" button, or typing a chat message while reading the previous incoming responses. To accommodate these types of interactions, PACE uses data-driven validation as an additional layer of validation to determine the goodness of the feature vector $\tilde{\boldsymbol{v}}$ and the corresponding assumed interaction-informed gaze point $\boldsymbol{c}$ based on previously validated data.

We use random forest [3] to build the gaze regression models for both x- and y-coordinates. The gaze feature vectors $\tilde{\boldsymbol{v}}$ are used as features and the corresponding interaction points $\boldsymbol{c}$ as the "truth". Each model generates 100 trees and each tree considers 4 random gaze features. The initial model is trained on the first 100 interaction instances in which the moment of the interaction occurs within a fixation period.

When the most-recently collected feature vector $\tilde{\boldsymbol{v}}$ is passed through the gaze model, it outputs a webcam estimated gaze point $\boldsymbol{g}_w$. If $D(\boldsymbol{g}_w, \boldsymbol{c}) \leq \lambda$, ($\lambda$ = 1/12 of the screen diagonal length in our work), the instance $[\tilde{\boldsymbol{v}}, \boldsymbol{c}]$ is considered validated and can be used as training data. For efficiency, the gaze model is updated in a batch mode; upon the collection of 150 valid instances of new training data, the random forest regression model is retrained on all the validated data. To make full use of all potential data, instances that fail the current validation are also retained and re-evaluated after each update of the gaze model.

In summary, PACE uses a dual-level validation. First, by looking for the stationary gaze features corresponding to an interaction event, the behavior-informed validation uses knowledge of gaze movement patterns and user interaction to identify the moment when user gaze and interaction point are most closely aligned. The data-driven validation further applies prior knowledge of the particular user to account for user individualities and contextual differences.

Correctness of Assumptions on User Gaze Patterns

The PACE approach makes some assumptions about the characteristics of user gaze patterns to identify good data points that can be used to train a gaze model. We are interested in whether these data points really correspond to moments when user gaze aligns with interaction event.

We use the eye gaze coordinates collected by the Tobii EyeX tracker as the gold standard for this evaluation. For each interaction event, we compare its location $\boldsymbol{c}$ with the Tobii-measured gaze coordinates $\boldsymbol{g}_t$ at the moment identified by the behavior-informed validation mechanism as the point of closest alignment. The error of our PACE method, $error_{PACE}$, is calculated as the displacement (Euclidean distance) between $\boldsymbol{g}_t$ and $\boldsymbol{c}$. We also calculate $error_{naive}$, which is the error that would result if we take the naïve assumption and simply choose the moment of the interaction event as the moment of best alignment, as in previous work [22]. As another point of reference, we compute $error_{min}$, which is the minimum possible error that we could achieve if we somehow knew the exact moment of best gaze-interaction alignment in the 3-second window; and $error_{minf}$, which is the minimum error achieved after discarding the obvious errors, defined as points where the displacement is larger than 1/12 of the screen diagonal length (≈240 pixels). These usually correspond to instances in which the user is clearly not looking at the interaction point – for example, when he/she is looking for a key on the keyboard.

Figure 5 compares the performance achieved by the three different approaches on the data described in the previous experiment. The color bars indicate the displacement values, and the circles denote the percentage of data that is retained after outliers are removed.

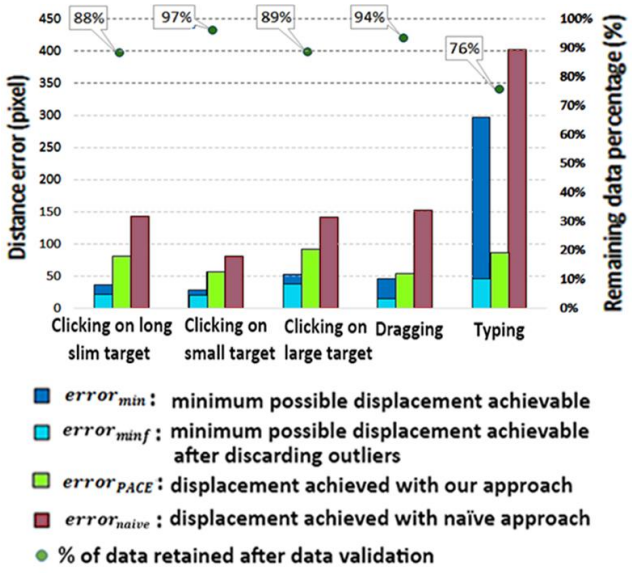

Figure 5. Displacement between gaze points identified with different approaches and location of corresponding interaction events.

The results are encouraging. As expected, $error_{min}$ (blue) can be very small, with an average of 41.1 pixels over all mouse interactions. For keyboard events, $error_{min}$ is much larger at 401.8 pixels. This is caused by the instances in which the user was not looking at what was being typed at all – i.e. the gaze was either wandering about the screen, or he/she was looking at the keyboard. When outliers are discarded, the keyboard event error decreases to 86.7 pixels. Discarding outliers for all events brings the error down to 27.8 pixels. However, this is the best achievable result, which is extremely difficult to achieve in practice.

In comparison, $error_{naive}$ (red) from the naïve approach is larger than $error_{PACE}$ (green) from our approach, across all interaction behaviors. The reason is obvious when one considers the U-shape pattern seen in most of the interaction behaviors (Figure 2), as the displacement falls to a minimum before the event, and then actually rises again just before the event. For example, with mouse drags, using the point at the event moment gives $error_{naive}$ of 153.3 pixels, while $error_{PACE}$, at 53 pixels, approaches the lowest possible $error_{min}$ (46 pixels). On average, $error_{naive}$ is 184.2 pixels while $error_{PACE}$ is 73.6 pixels.

It is also interesting to consider the amount of data that is retained after the two-layer validation process. We find that on average, 88.8% of the data is retained. The exception is typing, which has a low retention rate (76%), which is due to the large number of outliers. Incidentally, both the unfiltered $error_{min}$ and $error_{naive}$ are very large for typing interactions. However, our method is able to successfully identify these problematic data points, hence achieving a small $error_{PACE}$, which is close to that of mouse events. This is promising as keypress events are usually more numerous than mouse events, and hence it makes sense to find a means to include them as interaction-informed data.

Our results suggest that the proposed validation mechanism can effectively and precisely identify the reliable interaction-informed data, significantly outperforming the method based on the conventional assumption.

EVALUATION IN REAL-USE CONTEXTS

To evaluate the accuracy and effectiveness of the PACE system, we recruited 10 subjects (university students, 6 female, aged 20-33) for a focus study. Subjects were asked to choose at least 3 of the following tasks for the data collection: browsing websites, coding in Visual Studio, writing in Notepad, creating a figure using Microsoft Paint, and playing a shooting game (the House of the Dead). These tasks were chosen to cover a diverse range of common user interaction activities and applications, and to contain diverse interaction types. For example, some of the tasks will involve relatively dense keypresses while others contain mainly mouse events, like clicking and dragging.

The experiments were run on an i7-2600MHz PC with 4GB RAM, a 22" monitor and a standard off-the-shelf webcam

Methods	Error	Calibration	Data Required / Method Used
PACE	**2.56 °**	**Implicit, Automatic**	**mouse/keyboard interactions**
Sugano et al.[22]	4 º5 °	Implicit	click
Lu et al.[16]	2 º3 °	Explicit	video
Lu et al.[17]	2 º3 °	Explicit	image synthesis

Table 3. Performance of our approach, compared with state-of-the-art appearance models that allow free head motion.

capable of recording at 30fps. Running PACE on this setup achieves a frame rate of 22fps and performing a model update with 1500 data points takes less than 500ms. The gold standard eye gaze position is measured by the Tobii EyeX tracker. Approximated by our face tracker, the head-to-monitor distance ranges from 372~892mm (mean: 663mm; SD: 35.2mm); and head-to-camera pitch from -9.6 º~56.6 º(mean: 14.2 º, SD: 8.9 º).

At least 1500 events were collected from each subject. Each mouse click and press of a letter key logs the gaze feature data from the preceding 3 seconds. Subjects were allowed to pause and continue the experiment as needed, even over multiple days if necessary. They were also free to adjust head pose, body posture and chair position/height. The experiment lasted from 2 to 18 hours, depending on how long it took for the needed interaction events to be generated. On average, the subjects took around 4 hours to generate the required number of interaction points.

Table 3 shows the performance achieved by our method, as compared against similar state-of-the-art appearance-based methods that rely on webcam signals. The performance is measured as the average error over all collected instances and subjects. It is encouraging to see that our method achieves an average error of 30.9mm (*i.e* 2.56 º visual error, calculated using the approach in Sugano et al [22]), which is comparable to state-of-the-art approaches that require explicit calibration. This is a promising result, given that PACE (1) does not require explicit calibration and will

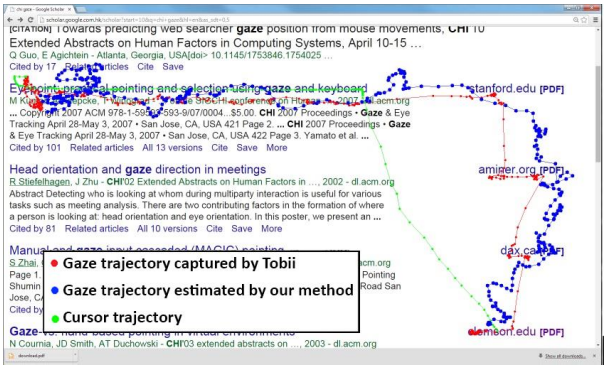

Figure 6. Trajectories of user gaze as estimated by PACE (blue) and as captured by the Tobii EyeX eye tracker (red). The cursor trajectory (green) is included for reference.

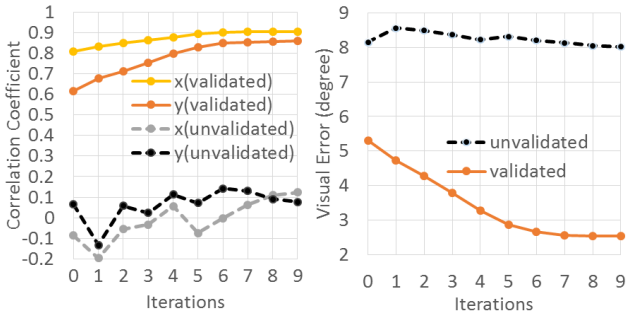

Figure 7. Comparison of PACE and naïve models. Change in performance (Correlation and Visual Error) as data increases. Each iteration consists of 150 interaction events.

automatically update itself to account for changes in light and posture variance, even across multiple sessions spanning over multiple days, (2) uses conventional off-the-shelf equipment that is commonplace in work environments, and (3) is tested using real applications and activities.

Figure 6 shows a graphical example of the performance of our system during a browsing activity. The blue line shows the eye movement estimated by PACE, while the red line denotes the true trajectory, measured by the Tobii tracker. It is seen that the PACE eye positions closely approximate that from the Tobii for the majority of the time, without using any additional equipment.

It is informative to consider the improvement in performance as the amount of data increases. We train two models using the same random forest algorithm. One is trained on data collected under the naïve assumption. The other is the PACE model, trained on data identified through the 2-step validation process. The models are updated and retrained every 150 interaction events (mouse clicks or keypresses). This means that the naïve model gets 150 new data instances per iteration, but the PACE model will get fewer instances, as the data-driven validation will invariably filter out some unreliable data points.

Figure 7 compares the performance (correlation and visual error) of the two models. Along the *x*-axis, each point represents one iteration of 150 interaction events. The performance of the naïve model fluctuates considerably – the visual error hovers around 8 º, and the correlation never increases beyond 0.2. The performance of PACE, on the other hand, improves monotonically as additional training data is provided. The correlation reaches an impressive 0.90 and 0.85 for the *x*- and *y*-coordinates respectively, and the visual error drops steadily to 2.56 º. This is further evidence that shows that our validation mechanism is effective as well as necessary for collecting interaction-informed training data in real-use situations.

CONCLUSION

This paper describes PACE, a Personalized, Automatically-Calibrating Eye-tracking system that can be integrated into

standard interactive computing systems. The assumptions behind PACE are informed by an in-depth study on the relationship between eye gaze and interaction location for several common types of interactive behaviors. Based on the results of the study, we then develop a novel approach that automatically identifies the moment of best gaze-interaction alignment, and a further data validation mechanism that accounts for user differences and context.

Experimental evaluations demonstrate that PACE can effectively extract good training data from daily interaction activities to build a reliable eye tracker with automatic updating and re-calibration. The performance thus achieved is comparable to those from similar state-of-the-art methods. However, PACE has the advantage that it automatically updates and re-calibrates itself and is therefore able to adjust to variances in conditions over multiple days and sessions.

In future work, we plan to probe further into different types of interaction behaviors, and also to take the semantics of the interaction (e.g. the functionality of the button that was clicked, or the type of the key that was pressed) and/or the history of the interaction into account. We also plan to investigate the impact of focus and attention on human eye gaze patterns and the performance of the eventual gaze tracking system. In addition, since it is also not difficult to see how the underlying data validation method could be combined with other gaze estimation techniques, we also plan to investigate the effectiveness of collecting training data in a similar way for appearance-based techniques using webcam video, or even for model-based techniques that work on infrared devices.

ACKNOWLEDGMENTS

We would like to thank all the experiment subjects for their help. We also thank the anonymous reviewers for their constructive comments and suggestions. This work was partially supported by grants PolyU 5235/11E and PolyU 5222/13E from the Hong Kong Research Grants Council.

REFERENCES

1. Alnajar, F., Gevers, T., Valenti, R. and Ghebreab, S. 2013. Calibration-Free Gaze Estimation Using Human Gaze Patterns. 2013 IEEE International Conference on Computer Vision (Dec. 2013), 137–144.
2. B. A. Shenoi 2005. Introduction to Digital Signal Processing and Filter Design. Wiley.
3. Breiman, L. 2001. Random forests. Machine Learning. 45, (2001), 5–32.
4. Chen, M.C., Anderson, J.R. and Sohn, M.H. 2001. What can a mouse cursor tell us more? CHI ’01 extended abstracts on Human factors in computing systems - CHI '01 (New York, New York, USA, 2001), 281.
5. Cootes, T.F., Edwards, G.J. and Taylor, C.J. 2001. Active appearance models. IEEE Transactions on Pattern Analysis and Machine Intelligence. 23, 6 (Jun. 2001), 681–685.
6. Fares, R., Fang, S. and Komogortsev, O. 2013. Can we beat the mouse with MAGIC? Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI ’13 (New York, New York, USA, 2013), 1387.
7. Guo, Q. and Agichtein, E. 2010. Towards predicting web searcher gaze position from mouse movements. Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems - CHI EA ’10 (New York, New York, USA, 2010), 3601.
8. Hansen, D.W. and Ji, Q. 2010. In the eye of the beholder: a survey of models for eyes and gaze. IEEE transactions on pattern analysis and machine intelligence. 32, 3 (Mar. 2010), 478–500.
9. Hornof, A.J. and Halverson, T. 2002. Cleaning up systematic error in eye-tracking data by using required fixation locations. Behavior research methods, instruments, & computers : a journal of the Psychonomic Society, Inc. 34, (2002), 592–604.
10. Huang, J., White, R. and Buscher, G. 2012. User see, user point: gaze and cursor alignment in web search. Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems - CHI ’12 (New York, New York, USA, 2012), 1341.
11. Huang, J., White, R.W. and Dumais, S. 2011. No clicks, no problem: using cursor movements to understand and improve search. Proceedings of the 2011 annual conference on Human factors in computing systems - CHI ’11 (New York, New York, USA, 2011), 1225.
12. Huang, M.X., Kwok, T.C.K., Ngai, G., Leong, H.V. and Chan, C.F.S. 2014. Building a Self-Learning Eye Gaze Model from User Interaction Data. ACM Multimedia (2014).
13. Jacob, R.J.K. 1990. What you look at is what you get: eye movement-based interaction techniques. Proceedings of the SIGCHI conference on Human factors in computing systems Empowering people - CHI ’90 (New York, New York, USA, 1990), 11–18.
14. Judd, T., Ehinger, K., Durand, F. and Torralba, A. 2009. Learning to predict where humans look. 2009 IEEE 12th International Conference on Computer Vision (Sep. 2009), 2106–2113.
15. Liebling, D.J. and Dumais, S.T. 2014. Gaze and mouse coordination in everyday work. Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct Publication - UbiComp ’14 Adjunct (2014), 1141–1150.

16. Lu, F., Okabe, T., Sugano, Y. and Sato, Y. 2014. Learning gaze biases with head motion for head pose-free gaze estimation. Image and Vision Computing. 32, 3 (Mar. 2014), 169–179.
17. Lu, F., Sugano, Y., Okabe, T. and Sato, Y. 2012. Head pose-free appearance-based gaze sensing via eye image synthesis. International Conference on Pattern Recognition (ICPR), (2012), 1008 – 1011.
18. Lu, F., Sugano, Y., Okabe, T. and Sato, Y. 2014. Inferring Human Gaze from Appearance via Adaptive Linear Regression. IEEE Transactions on Pattern Analysis and Machine Intelligence. (2014), 1–1.
19. Rodden, K., Fu, X., Aula, A. and Spiro, I. 2008. Eye-mouse coordination patterns on web search results pages. Proceeding of the twenty-sixth annual CHI conference extended abstracts on Human factors in computing systems - CHI '08 (New York, New York, USA, 2008), 2997.
20. Saragih, J.M., Lucey, S. and Cohn, J.F. 2009. Face alignment through subspace constrained mean-shifts. 2009 IEEE 12th International Conference on Computer Vision (Sep. 2009), 1034–1041.
21. Sugano, Y., Matsushita, Y. and Sato, Y. 2013. Appearance-based gaze estimation using visual saliency. IEEE transactions on pattern analysis and machine intelligence. 35, 2 (Feb. 2013), 329–41.
22. Sugano, Y., Matsushita, Y., Sato, Y. and Koike, H. 2008. An Incremental Learning Method for Unconstrained Gaze Estimation. 10th European Conference on Computer Vision, ECCV' 2008 (2008), 656–667.
23. Williams, O., Blake, A. and Cipolla, R. Sparse and Semi-supervised Visual Mapping with the S^3GP. 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1 (CVPR'06) 230–237.
24. Wood, E. and Bulling, A. 2014. EyeTab. Proceedings of the Symposium on Eye Tracking Research and Applications - ETRA '14 (New York, New York, USA, 2014), 207–210.
25. Xiong, X. and De la Torre, F. 2013. Supervised Descent Method and Its Applications to Face Alignment. 2013 IEEE Conference on Computer Vision and Pattern Recognition (Jun. 2013), 532–539.
26. Zhang, Y. and Hornof, A.J. 2014. Easy post-hoc spatial recalibration of eye tracking data. Proceedings of the Symposium on Eye Tracking Research and Applications - ETRA '14 (2014), 95–98.
27. Zhang, Y. and Hornof, A.J. 2011. Mode-of-disparities error correction of eye-tracking data. Behavior research methods. 43, (2011), 834–842.
28. Zhu, Z., Ji, Q. and Bennett, K.P. 2006. Nonlinear Eye Gaze Mapping Function Estimation via Support Vector Regression. 18th International Conference on Pattern Recognition (ICPR'06) (2006), 1132–1135.