Wearable Eye Tracker Calibration
Lab Project

October 3, 2017

Description

Eye tracking measures the point of gaze (where one is looking) or the rotation of the eye relative to the head. Many different application domains can benefit from knowing where the users direct their visual attention. Market researchers can better understand and improve the design or advertising of a product. Eye gaze and other biometric data can help doctors diagnose certain diseases. A website’s usability can be evaluated by taking the point of gaze into account. The above mentioned applications are only a few which use eye tracking in a passive way. Our eyes can also be used interactively to control user interfaces, help people with disabilities to communicate, or create more immersive video games.

Nowadays, most eye-trackers are video-based. They rely on infra-red cameras to track the movement of the eye and can be divided into two categories: remote and wearable eye trackers. Remote eye trackers (e.g., the Tobii EyeX\(^1\)) have to be attached to or next to a display, thus limiting the scope of the application. Wearable eye trackers like the Pupil\(^2\) allow the users to walk freely in the surrounding environment. In this project, the focus will be on wearable devices: Head-worn with at least two cameras, one facing the eye and a scene camera. By calibrating the eye camera with respect to the scene camera, it is possible to know, with a certain error, where the user is looking.

The calibration procedure is an important step in any eye tracking application. Remote eye trackers are usually calibrated by showing the user certain patterns on a screen. This enables collecting a few sample points which are used to find a mapping between the eye characteristics and a position on the display. While this method works well for remote eye trackers, wearable eye trackers cannot always use a screen for the calibration process. Additionally, head worn devices could slightly move which would require a continuous calibration technique. User experiments [1] further report that traditional calibration techniques are cumbersome and difficult for inexperienced users.

This project aims to develop a novel calibration technique for wearable eye trackers which will simplify this process for its users.

Work packages

WP 1

Setting-up the environment and exploring eye tracking calibration

\textit{Expected time effort: 1 month}

\footnote{https://tobiiigaming.com/product/tobii-eyex/}

\footnote{https://pupil-labs.com/pupil/}
In this work package, the student is expected to set-up the working environment, get familiar with the tools, and read existing literature about eye tracking.

**Deliverable:** a report explaining the state of the art for eye tracking calibration.

**WP 2**

**Baseline implementation**

*Expected time effort: 1 month*

The student is expected to identify at least two closely related works, understand them, and implement them. This will help understand existing approaches and, at the same time, create the baseline for the system evaluation.

**Deliverable:** Implementation of at least two closely related projects and their evaluation.

**WP 3**

**Propose a novel calibration method**

*Expected time effort: 2 months*

In this work package, the student is expected to develop a novel, simpler calibration technique.

**Deliverable:** Design and implementation.

**WP 4**

**Testing and evaluation**

*Expected time effort: 1 month*

The calibration method developed in WP3 must be evaluated and compared to the baseline from WP2.

**Deliverable:** a report containing the performance evaluation.

**(Optional) WP 5**

**User evaluation**

*Expected time effort: 1 month*

The student has to develop a user experiment to validate whether the proposed calibration technique is perceived as simpler than existing state of the art methods.

**Deliverable:** a report containing the user evaluation.

**Report and Presentation**

The students must present the project in a talk (15 minutes + 5 minutes if the talk includes a demonstration). Additionally, a written report has to be submitted and it should cover the work done, theoretical aspects of the thesis, encountered difficulties, and other relevant topics (e.g., code documentation).
Required Skills

- Good programming skills (Android programming is a plus)
- Ability to work independently
- Good problem solving skills
- Computer vision skills are a plus

Grading

Grading will be based on the following criteria:

<table>
<thead>
<tr>
<th>Criterion</th>
<th>Weight (± 0.5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Implementation (functionality, extensibility, documentation)</td>
<td>3</td>
</tr>
<tr>
<td>Report (content, illustration, writing)</td>
<td>2</td>
</tr>
<tr>
<td>Presentation (content, illustration, quality of talk)</td>
<td>1</td>
</tr>
<tr>
<td>Approach (organization, approaching problems, independence, involvement)</td>
<td>1</td>
</tr>
<tr>
<td>Contribution to the state of the art research</td>
<td>1</td>
</tr>
<tr>
<td>Completion of all tasks</td>
<td>3</td>
</tr>
</tbody>
</table>

Supervision

Advisor:
Mihai Băce, ETH Zürich, Distributed Systems Group, mihai.bace@inf.ethz.ch

Supervisors:
Prof. Friedemann Mattern, ETH Zürich, mattern@inf.ethz.ch

References