Augmented Reality and Eye-Tracking in iOS Development

Prune Ollier
Supervisor: Lucas Burget

MOTIVATION

This report explores the development of an iOS eye-tracking app using Apple's ARKit technology.

Executed within the CHILI research lab at EPFL in the context of my Bachelor's semester project, the focus was not merely on app performance but on the research and development that lead to the accuracy analysis of this technology.

The project's core objective was to harness the eye-tracking capabilities within Apple's ARKit framework to build an iOS app. While the current iteration is optimized for iPhone, strategic considerations anticipate future adaptations for iPads, broadening the potential applications of this technology.

METHODS

The project used Swift, Apple's programming language, to harness the different frameworks. The development process included:

- Learning Swift and understanding basic iOS app structures.
- Exploring ARKit and SpriteKit through hands-on experimentation and Apple's documentation and tutorials.
- Implementing AR and eye-tracking features in an iOS app, with a focus on integrating ARKit's real-world tracking and SpriteKit's graphical capabilities.
- Iterative testing and refining for optimal performance.
- Analysis of the results.

RESULTS

The performance evaluation essentially involved experiments where the user followed their finger with their gaze across the screen.

The experiments showed that gaze estimation accuracy was influenced by head rotation and position.

- The gaze's x-coordinate was primarily affected by the head's y-coordinate and rotation around the x-axis, highlighting the differences between ARKit and SpriteKit referentials.
- Conversely, the gaze's y-coordinate was influenced by the head's x and y coordinates and rotation about the y-axis, indicating better performance in estimating gaze along the x-axis and less stability for the y-coordinate.
- The head's z-coordinate and rotation about the z-axis had minimal impact on gaze estimation.