Optimization-Based Eye Tracking using Deflectometric Information

Tianfu Wang*, Jiazhang Wang**, Oliver Cossairt**, Florian Willomitzer***

*Department of Computer Science, ETH Zürich, Zürich, Switzerland, 8092
**Department of Electrical and Computer Engineering, Northwestern University, Evanston, IL, 60208
***Wyant College of Optical Sciences, University of Arizona, Tuscon, AZ, 85721

mailto:fwillomitzer@arizona.edu

We attempt to improve reflection-based eye tracking methods by using deflectometric measurements in combination with optimization-based inverse rendering methods. By exploiting image-screen correspondence data, we determine eye *rotation*, *translation*, and *shape* using gradient descent. We demonstrate the working principle of our method in simulation and with real experiments.

1 Introduction

In the last decades, accurately tracking the position and gaze direction of the human eye became a crucial technology for both commercial and scientific domains. Current eye tracking methods fall into two categories: image-based methods, which use 2D eye features to determine the gaze direction, and reflection-based methods like "glint tracking," which use sparse light reflections on the eye surface. Recently, we introduced a simple but effective method to *drastically increase* the number of reflection points and thus to obtain a *pixel-dense sampling* of the eye surface [1]: by replacing the array of sparse point light sources with an extended self-illuminated screen. The screen displays a specific (known) pattern, and the reflection of this pattern over the eye surface is observed with the camera. The deformation of the pattern in the camera image reveals information about the rotation and translation of the measured eye as well as its shape (if desired). With this idea, the acquired information about the eye surface is drastically increased. See [1, 2] for more details.

In this contribution, we present an inverse-rendering based eye tracking method utilizing the pixel-dense eye surface information obtained from the idea above. We develop a differentiable rendering pipeline similar to [3] that simulates a virtual eye under screen illumination. Eventually, we exploit the image-screen-correspondence from the captured measurements to find the eye's *rotation*, *translation*, and *shape* parameters with our renderer via gradient descent. We demonstrate the robustness of our method through real-world experiments. Moreover, we show in simulation an improvement of 6X over a representative state-of-the-art glint tracking method [4].

2 Method

Our method (Fig. 1) takes a set of real deflectometric measurements of the eye, and aims to find the eye's *rotation*, *translation*, and *shape* through inverse rendering via our developed differentiable deflectometry renderer. To ensure accurate estimation and updating of eye parameters, we utilize PyTorch3D, a differentiable renderer based on rasterization. However, PyTorch3D does not provide native support for screen illumination. To address this limitation, we develop a specialized shader similar to [3] that simulates specular reflection from area lighting and serves as a single bounce ray-tracer. Eventually, we use our renderer to "build" a virtual deflectometry setup in simulation. The simulated setup uses a realistic computer-generated (CG) eye model and the known geometry of our calibrated deflectometry setup [2], meaning that it can be seen as a "simulated digital twin" of our real setup. We start our evaluation procedure by placing our CG eye at an arbitrary start position in the virtual setup. For each pixel, we calculate the specular reflection off the CG eye surface by reflecting the camera view direction vector off the eye's surface normal. By intersecting this reflection ray with the screen, we obtain the screen position, and can establish the screen-image correspondence.

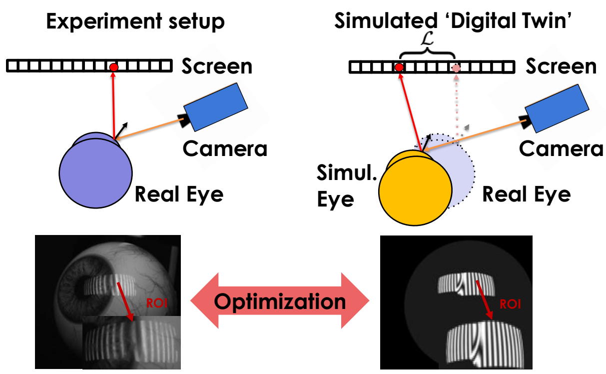

Fig. 1 *Schematic of our proposed method.*

To estimate the gaze direction of the real (not simulated) eye, our objective function for optimization uses the screen-image correspondence information obtained from the real measurements and compares

DGaO Proceedings 2023 – http://www.dgao-proceedings.de – ISSN: 1614-8436 – urn:nbn:de:0287-2023-B007-4
submitted: 01.09.2023 – published: 25.09.2023

them with the screen-image correspondence information obtained from the digital twin setup. Our objective is then to minimize the mean squared distance between the real and virtual correspondence locations on the screen. During the gradient descent optimization, the CG eye model is moved/rotated in the virtual space until the virtual correspondences match closely with the real correspondences. For real-world experiments, it is possible that the shape of the measured eye varies for different human subjects. It is therefore necessary to develop additional methods to accommodate for varying eye shapes for improved robustness. For easier optimization, we add an additional constraint on the eye that it is rotational symmetric around its optical axis. Under this assumption, we model the shape of the eye as a set of connected circular edge loops centered around the optical axis, and optimize the radii of the edge loops for the frontal part of the eye model. Additionally, we regularize the gradient and curvature of the radii, as well as the mesh Laplacian of the whole eye shape, to ensure the eye shape is smooth and without bumps. We refer to [5] for more details.

3 Results

Real World Experiments: To validate our joint shape and gaze optimization model in a quantitative fashion, we conduct real-world experiments on a realistic physical 3D eye model on a rotation stage with fixed rotation positions. Since the *absolute* ground-truth gaze direction of the eye model cannot be evaluated, we propose to use *relative gazing angles* for error evaluation. For each rotation position a, we take 20 different measurements and calculate the gaze angle θ_a for each measurement. Eventually, we evaluate the mean relative error ϵ_{0° at each rotation position a w.r.t. the 0° position: $\epsilon_{0^\circ} = ||\bar{\theta_a} - \bar{\theta_{0^\circ}}| - |a - 0^\circ||$. Since the exact shape of the eye model is not known, we also optimize the eye shape while finding the gaze. The results are shown in Tab. 1. We achieve precision values between 0.11° and only 0.02° and a mean relative error between 0.45° and only 0.27°. This demonstrates the robustness of our joint shape and gaze optimization model for real-world experiments (results without shape optimization are shown for comparison in Tab. 2).

Rotation	**0°**	**2°**	**4°**	**-2°**	**-4°**
Precision σ_a	.02°	.10°	.08°	.11°	.11°
Mean error ϵ_0	0°	.33°	.28°	.27°	.45°

Tab. 1 *Gaze angle estimation for real-world experiments*

Rotation	**0°**	**2°**	**4°**	**-2°**	**-4°**
Precision σ_a	.04°	.02°	.05°	.09°	.06°
Mean error ϵ_0	0°	.01°	.49°	.44°	1.17°

Tab. 2 *Gaze angle estimation for real-world experiments* ***without shape regularizer***

Comparison with glint-based eye tracking: We use our deflectometry renderer to compare our optimization approach with a representative glint tracking method (in simulation), particularly a technique using 12 glints [4]. In our renderer we simulate images of our virtual eye model with corneal glint reflections. We compare glint tracking against our framework in simulation using our correspondence loss method, with sinusoid patterns. Additionally, we display an image of an arbitrary movie scene to simulate VR display content. We use the top 50 closest SIFT feature matches between the image and the rendered eye image as image-screen correspondences. We tested all methods under the same set of 50 random gaze angles, and calculated the precision and the mean angle error. Our method showed lower average gaze direction errors and better precision (Tab. 3). We attribute this advantage to utilizing dense light source information and 3D scene modeling. Our experiment also revealed that using SIFT features is effective, achieving similar gaze error to glint tracking.

Method	**Glint Track.**	**Sin Corr.**	**SIFT Corr.**
Precision	0.11°	0.02°	0.02°
Mean Error	0.20°	0.03°	0.16°

Tab. 3 *Simulation: Comparison of two different flavors of our method with glint tracking*

Besides the discussed optimization-based approach, we have also introduced another approach based on "classical" deflectometry at this year's DGaO [2, 6].

References

[1] J. Wang, B. Xu, T. Wang, W. J. Lee, M. Walton, N. Matsuda, O. Cossairt, and F. Willomitzer, "VR eye-tracking using deflectometry," in *Computational Optical Sensing and Imaging*, pp. CF2E–3 (2021).

[2] J. Wang, T. Wang, B. Xu, O. Cossairt, and F. Willomitzer, "Accurate Eye Tracking from Dense 3D Surface Reconstructions using Single-Shot Deflectometry," arXiv preprint arXiv:2308.07298 (2023).

[3] O. Cossairt, F. Willomitzer, C.-K. Yeh, and M. Walton, "Low-budget 3D scanning and material estimation using PyTorch3D," in *54th Asilomar Conference on Signals, Systems, and Computers* (2020).

[4] C. Mestre, J. Gautier, and J. Pujol, "Robust eye tracking based on multiple corneal reflections for clinical applications," Journal of Biom. Opt. **23**(3) (2018).

[5] T. Wang, J. Wang, O. Cossairt, and F. Willomitzer, "Optimization-Based Eye Tracking using Deflectometric Information," arXiv preprint arXiv:2303.04997 (2023).

[6] J. Wang, T. Wang, B. Xu, O. Cossairt, and F. Willomitzer, "VR Eye Tracking using Single-Shot Deflectometry„" DGaO-Proceedings (2023).