

A Real-Time Gaze Estimation Framework for Mobile Devices

Yu Feng¹, Nathan Goulding-Hotta², Asif Khan², Hans Reyserhove², Yuhao Zhu¹

¹University of Rochester, ²Facebook Reality Lab

Keywords—gaze tracking, eye tracking, eye segmentation, ROI prediction, augmented and virtual reality.

I. INTRODUCTION

Tracking eyes becomes an important component to unleash new ways of human-machine interactions in augmented and virtual reality (AR/VR). To make the eye tracking system responsible, eye tracking systems need to operate at a real-time rate ($> 30\text{Hz}$). However, from our experiments, modern gaze tracking algorithms operate at most 5 Hz on mobile processors. In this talk, we present a real-time eye tracking algorithm that operates at 30 Hz on a mobile processor. Our algorithm achieves sub- 0.5° gaze accuracy, while requiring only 30K parameters, which is one to two orders of magnitude smaller than state-of-the-art algorithms.

Temporal Correlation While most of the prior eye tracking algorithms work on a frame-by-frame basis, eye tracking in most AR/VR use cases processes continuous, sequential frames, which exposes temporal information that is often ignored. Our system leverages this temporal correlation to achieve real-time eye tracking.

Specifically, we use the past temporal correlation across frames to predict the Region of Interest (ROI) in the current frame and propose a novel ROI prediction algorithm that is tailored for eye tracking. The ROI prediction allows us to process eye frames within the ROIs and significantly improves the execution speed.

Lightweight Segmentation Coupled with the ROI prediction, we propose a lightweight neural network-based algorithm to learn the gaze from a ROI. The way to estimate the gaze is to first parameterize an eye image through eye segmentation and then estimate the gaze through fitting a geometric eye model. To accelerate the heavyweight eye segmentation, we propose a new segmentation network inspired by the classic MobileNet [2], we show that depth-wise separable convolution provides a segmentation backbone that balances accuracy and compute complexity. Our network requires 13–93 \times fewer parameters (30K in total) and 20–31 \times fewer operations than existing DNN algorithms.

Reducing Data Transmission Apart from delivering real-time tracking on off-the-shelf mobile platforms today, our system has a potential to reduce data transmission overheads, both between the image sensor and the back-end processor and between the

processor and memory. We show that in-sensor Auto-ROI provides 2–4 \times reduction in the transmission energy.

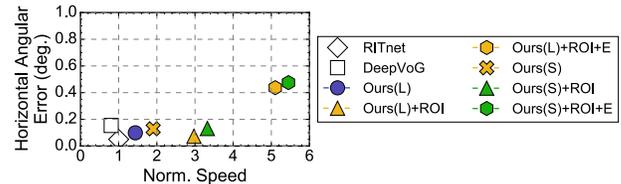


Fig. 1. The gaze accuracy and speed comparison of different methods. The speedup values are normalized to the speed of RITnet. Ours (S) and Ours (L) are two eye segmentation networks. +ROI and +E denote Auto-ROI is enabled with two different execution modes.

II. EVALUATION

We compare against two eye segmentation methods, RITnet [1] and DeepVoG [3]. Both the baselines are trained using the same procedure as our algorithm. We use two accuracy metrics, the angular error for gaze estimation and the mean Intersection-over-Unit (mIoU) metric for eye segmentation. We also compare the speed of our algorithms with the baselines. We measure the speed on a state-of-the-art mobile computing platform, Nvidia’s Jetson Xavier board.

Variants We use two DNN variants of our eye segmentation network, Ours (S) and Ours (L). Both are designed with the same architecture but differing in the channel width. Table 1 compares the FLOPs and number of parameters of the two networks against the baseline segmentation networks.

Table 1: FLOPs and number of parameters in different eye segmentation networks under an input size of 640×400 .

| Network | DeepVOG | RITnet | Ours (L) | Ours (S) |
|----------------------------|---------|--------|----------|----------|
| FLOPs (Billion) | 36.5 | 23.1 | 2.6 | 1.2 |
| Norm. FLOPs | 31.1 | 19.7 | 2.3 | 1.0 |
| # of Parameters (Thousand) | 2835.7 | 391.0 | 73.0 | 30.6 |
| Norm. # of Parameters | 92.6 | 12.8 | 2.4 | 1.0 |

Result Figure 1 compares the gaze error and the speed of our algorithms with different baselines. The speedups are normalized to the speed of RITnet. Ours (L), and Ours (S) can keep the absolute error rate below 0.1° as RITnet with better speedup. Overall, our algorithm achieves an about 5.5 \times speedup over the baselines with a sub- 0.5° gaze error.

III. REFERENCES

- [1] A. K. Chaudhary, R. Kothari, M. Acharya, S. Dangi, N. Nair, R. Bailey, C. Kanan, G. Diaz, and J. B. Pelz. Ritnet: Real-time semantic segmentation of the eye for gaze tracking. In 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), pp. 3698–3702. IEEE, 2019.
- [2] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861, 2017.
- [3] Y.-H. Yiu, M. Aboulatta, T. Raiser, L. Ophey, V. L. Flanagan, P. Zu Eulenburg, and S.-A. Ahmadi. Deepvog: Open-source pupil segmentation and gaze estimation in neuroscience using deep learning. *Journal of neuroscience methods*, 324:108307, 2019.