EyeTrAES: Fine-grained, Low-Latency Eye Tracking via Adaptive Event Slicing

Citation: Sen, Argha, Bandara, Nuwan, Gokarn, Ila, Kandappu, Thivya and Misra, Archan. 2024. "EyeTrAES: Fine-grained, Low-Latency Eye Tracking via Adaptive Event Slicing." Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 8 (4).

As Published: https://doi.org/10.1145/3699745

Publisher: ACM

Persistent URL: https://hdl.handle.net/1721.1/157898

Version: Final published version: final published article, as it appeared in a journal, conference proceedings, or other formally published context

EyeTrAES: Fine-grained, Low-Latency Eye Tracking via Adaptive Event Slicing

ARGHA SEN*, Indian Institute of Technology Kharagpur, India
NUWAN SRIYANTHA BANDARA, Singapore Management University, Singapore
ILA GOKARN, Singapore-MIT Alliance for Research and Technology (SMART), Singapore
THIVYA KANDAPPU, Singapore Management University, Singapore
ARCHAN MISRA, Singapore Management University, Singapore

Eye-tracking technology has gained significant attention in recent years due to its wide range of applications in human-computer interaction, virtual and augmented reality, and wearable health. Traditional RGB camera-based eye-tracking systems often struggle with poor temporal resolution and computational constraints, limiting their effectiveness in capturing rapid eye movements. To address these limitations, we propose *EyeTrAES*, a novel approach using neuromorphic event cameras for high-fidelity tracking of natural pupillary movement that shows significant kinematic variance. One of *EyeTrAES's* highlights is the use of a novel adaptive windowing/slicing algorithm that ensures just the right amount of descriptive asynchronous event data accumulation within an event frame, across a wide range of eye movement patterns. *EyeTrAES* then applies lightweight image processing functions over accumulated event frames from just a single eye to perform pupil segmentation and tracking (as opposed to gaze-based techniques that require simultaneous tracking of both eyes). We show that these two techniques boost pupil tracking fidelity by 6+%, achieving IoU~=92%, while incurring at least 3x lower latency than competing pure event-based eye tracking alternatives [38]. We additionally demonstrate that the microscopic pupillary motion captured by *EyeTrAES* exhibits distinctive variations across individuals and can thus serve as a biometric fingerprint. For robust user authentication, we train a lightweight per-user Random Forest classifier using a novel feature vector of short-term pupillary kinematics, comprising a sliding window of pupil (location, velocity, acceleration) triples. Experimental studies with two different datasets (capturing eye movement across a range of environmental contexts) demonstrate that the *EyeTrAES*-based authentication technique can simultaneously achieve high authentication accuracy (~=0.82) and low processing latency (~=12ms), and significantly outperform multiple state-of-the-art competitive baselines.

CCS Concepts: • **Human-centered computing** → *Ubiquitous and mobile computing*.

Additional Key Words and Phrases: Eye Tracking, Event Cameras, Adaptive Event Sampling, Authentication

ACM Reference Format:
Argha Sen, Nuwan Sriyantha Bandara, Ila Gokarn, Thivya Kandappu, and Archan Misra. 2024. *EyeTrAES*: Fine-grained, Low-Latency Eye Tracking via Adaptive Event Slicing. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 8, 4, Article 150 (December 2024), 32 pages. https://doi.org/10.1145/3699745

*Corresponding author

Authors' addresses: Argha Sen, Indian Institute of Technology Kharagpur, Kharagpur, India, arghasen10@gmail.com; Nuwan Sriyantha Bandara, Singapore Management University, Singapore, pmnsbandara@smu.edu.sg; Ila Gokarn, Singapore-MIT Alliance for Research and Technology (SMART), Singapore, ila.gokarn@smart.mit.edu; Thivya Kandappu, Singapore Management University, Singapore, thivyak@smu.edu.sg; Archan Misra, Singapore Management University, Singapore, archanm@smu.edu.sg.

2474-9567/2024/12-ART150 $15.00
https://doi.org/10.1145/3699745

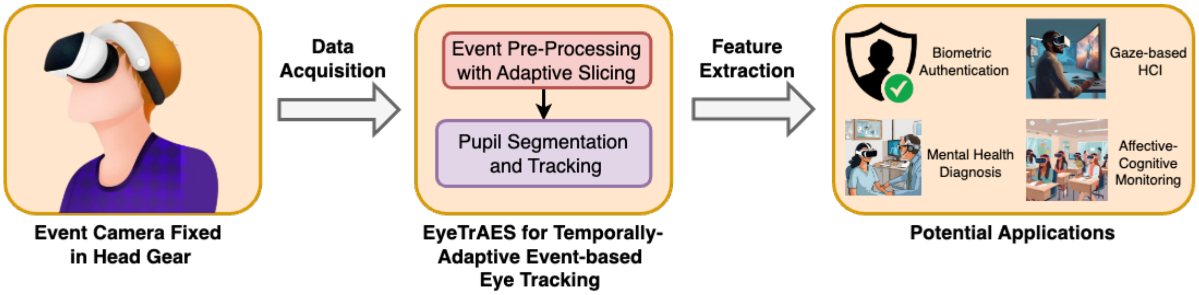

Data Acquisition
Event Pre-Processing with Adaptive Slicing
Pupil Segmentation and Tracking
Feature Extraction
Biometric Authentication
Gaze-based HCI
Mental Health Diagnosis
Affective-Cognitive Monitoring
Event Camera Fixed in Head Gear
EyeTrAES for Temporally-Adaptive Event-based Eye Tracking
Potential Applications

Fig. 1. Broad Overview of *EyeTrAES*.

1 INTRODUCTION

Fine-grained eye-tracking has become an increasingly important enabler of a variety of applications, spanning areas such as human-computer (gaze-based) interaction, consumer visual attention analysis, visuo-motor disease prediction [51], Autism spectrum disorder identification [4], and biometric authentication [10, 30, 41]. Traditional eye-tracking systems typically rely on RGB cameras to capture images of the eye (often both eyes), which are then processed to extract information related to features such as gaze direction, fixation, and saccades. However, these systems often face challenges such as poor temporal resolution, relatively low frame rates, limited dynamic range, and high computational overheads. This is especially important as the eye muscles can generate powerful, short-lived but high-velocity movements. More specifically, the human eye is characterized by angular velocity exceeding speeds of $300°/s$, particularly during saccadic eye motions [3], and ocular acceleration reaching values as high as $24,000°/s^2$ [1].

In recent years, neuromorphic event cameras have emerged as a promising alternative to traditional RGB cameras for eye-tracking applications [3, 8, 38, 48, 58]. Unlike traditional RGB cameras that capture frames at a fixed rate, event cameras produce events asynchronously, on a per-pixel basis, whenever there is a change in the incident intensity and with very low (O(μsec)) latency. As a consequence, such cameras can not only operate at sampling rates of 10,000 Hz and above, but also result in their event generation rate adapting to the underlying event (i.e., eye movement) velocity. These unique characteristics make event cameras well-suited for capturing fast and dynamic phenomena, such as *fine-grained* eye movements, with high temporal resolution and low latency.

A key challenge of such event camera-based eye tracking, besides the well-known lack of underlying color or texture information, is its extremely high event rate (as high as 1.06 Geps (Giga events per second) for dynamic scenes) and the consequent computational burden of event processing. To tackle these challenges, past work on ocular event data processing (e.g., [38]) first aggregates a collection of asynchronous events to create synchronous *event frames*. Most eye tracking-based techniques first attempt to track the location (in the camera frame coordinates) of the eye pupil region within each such *event frame* using standard image processing methods, and then extract pupillary movement-related features from a sequence of such 'pupil location' values. In all past work, frame aggregation is performed over *fixed windows* of either time or events. We shall, however, demonstrate that such a fixed windowing strategy is inadequate to support the wide *variation* in eye movement rates (up to 700°/sec during saccadic eye movements, as found in our studies): a small window captures insufficient kinematic data during periods of slow eye movement, whereas a large window is susceptible to *over-accumulation*, and consequent loss of fine-grained movement information, during periods of rapid eye movement.

In this paper, we explore the challenge of developing an event camera-based *eye movement tracking* technique that has significantly higher spatiotemporal accuracy and is computationally efficient to support low-latency on-device execution. Our vision is that such miniaturized inward-facing event cameras can be mounted on personal wearable devices, such as augmented reality (AR) smartglasses or head-mounted displays (HMDs). By continuously tracking the wearer's natural eye movement, such event cameras can support real-time extraction of pupillary kinematics. Based on results in prior physiological research, we believe that such accurate extraction of temporal pupillary movement features even under high kinematic variance can further increase the fidelity and variety of eye-tracking-based applications – for example, rapid and intricate movements of eyes have been used in (a) diagnosing neurodevelopmental disorders (such as, ASD [4], and ADHD [34]), (b) detecting low vision conditions [21], (c) measuring cognitive load [13], and (d) emotion recognition [53].

We shall show that the lack of adaptation in the accumulation window of current approaches consequently leads to a significant increase in pupil localization error, which in turn can lead to inaccurate computation of fine-grained temporal pupillary movement features. Our approach to on-device pupillary kinematic tracking using event cameras, called *EyeTrAES*[1] first proposes a novel *adaptive slicing technique* to convert asynchronous event streams into event frames that capture just the right amount of pupillary kinematic information. We shall then present a new, classical image processing pipeline (in contrast to in-vogue heavyweight DNN based approaches) that can both extract the pupillary location coordinates from such event frames with high accuracy and is computationally lightweight enough to permit *on-device* execution. We shall experimentally quantify the resulting improvements in real-time, on-device eye movement tracking and pupil localization. To additionally demonstrate the practical benefits of *EyeTrAES's* improved pupillary tracking, we shall finally introduce a new biometric user authentication technique that utilizes a machine learning-based model, trained on natural pupillary *micro-movement* features curated from individual-specific event-based eye movement data. Figure 1 depicts the sequence of novel capabilities that we shall demonstrate.

Key Scientific Contributions:

(a) *Develop an Adaptive Event Windowing and Time Slicing Technique:* We first demonstrate that past approaches for frame-based processing of event data (e.g., [38]), based on fixed windows of either time or events, are ill-suited to support the wide variation in eye movement rates (up to 700°/s during saccadic eye movements, as found in our studies). Instead, *EyeTrAES* employs a novel adaptive slicing technique to convert asynchronous event streams into event frames that capture just the right amount of pupillary kinematic information. Our approach maintains a running average of the (mean, standard deviation) of polarities of incoming events and demarcates a frame boundary only when the standard deviation exceeds a designated threshold. We shall show that this adaptive slicing approach results in a significantly lower error in pupil segmentation (30+% improvement in IoU scores), across multiple segmentation approaches, compared to a conventional fixed window approach.

(b) *Ensure Low-Complexity, Low-Latency Pupil Tracking:* We develop an accurate and low-latency pupil tracking technique, amenable to *on-device* execution, that pipelines a sequence of classical image processing techniques (such as edge detection, Hough transform based circle detection and Kalman filtering) over successive event frames. We show that *EyeTrAES's* pupil segmentation module is superior in its ability to provide both accurate and low-latency pupil segmentation, achieving an IoU of 92% and incurring a per-frame latency of 4.7 ms on a commodity Intel i9 workstation, compared to other alternatives such as Ev-Eye [58] (IoU=89%, latency=700 ms), RGB based segmentation (IoU=84%, latency=8 ms) and E-Gaze [38](IoU=87%, latency=12 ms).

(c) *Establish the Distinctiveness of new Fine-grained (microscopic) Reflexive Eye Movement Features:* Drawing inspiration from studies in vision science and tracking [5, 16, 23, 25, 30, 42], we hypothesize that individual differences in ocular muscle strength generate distinct individual-specific natural spatiotemporal microscopic eye movement patterns. More specifically, we show that novel, short-duration kinematic features, such as *pupillary velocity and acceleration*, together with the pupil's location, effectively serve as a biometric fingerprint.

[1] Eye Tracking via Adaptive Event Slicing, pronounced Eye-Trace

Compared to prior approaches based on gaze features, *EyeTrAES's* classifier requires tracking of only one eye and does not require calibration based on the gaze distance (distance to the viewed object).

(d) *Demonstrate EyeTrAES's use in Biometric User Authentication:* Using data from both a publicly available prior dataset Ev-Eye [58] (10 users, but with fixed eye-screen distances) and an *EyeTrAES* dataset collected via our own user studies (40 participants, no constraint on the eye-screen distance or on head movement), we shall demonstrate the superiority of user authentication using *EyeTrAES*-derived microscopic eye movement features. Our approach achieves significantly higher accuracy (median accuracy ~0.82 on both datasets) than both (i) Ev-Eye (median accuracy of 0.62 and 0.375 for Ev-Eye and *EyeTrAES* datasets, respectively), and (ii) a high frame rate (120 FPS) RGB camera-based authentication technique (median accuracy = 0.71). Moreover, *EyeTrAES* is able to perform such authentication rapidly, with an average authentication response time of ~0.14 sec, depending on the kinematics of pupil movement.

Overall, we believe that *EyeTrAES* dramatically improves the capability for pure event sensor-based pupillary movement tracking and resulting pupil movement-based user authentication, offering a compellingly superior alternative to extant *gaze-based* methods that require significantly greater instrumentation and calibration. Furthermore, we have open-sourced our implementation for the research community, with the code and dataset accessible at https://github.com/arghasen10/EyeTrAES

2 RELATED WORK

Eye-tracking technology has been the subject of extensive research and development in recent years, leading to a wide range of approaches and techniques for analyzing eye movements and extracting features for various applications. In this section, we review some of the key works related to eye-tracking using both RGB and event cameras, image processing methods for pupil tracking, and user authentication based on eye movements.

RGB Frame-based Eye/Gaze Tracking Pupil tracking is a critical step in eye-tracking systems, as it provides information about the user's gaze direction and fixation points. To this end, most existing works for eye or pupil tracking are RGB frame-based approaches that utilize either traditional image processing methods [12, 17–19, 26, 49] or end-to-end deep learning methods [7, 15, 29, 31].

With regard to the traditional image processing approaches for pupil tracking, methods including color filtering, ellipse fitting, and contour analysis are typically implemented in the literature such as [19], in which the authors presented a method for pupil tracking using adaptive thresholding and ellipse fitting, achieving high accuracy in tracking pupil movements. Similarly, [12] utilized canny edge detection and blob detection to segment the frames and ellipse fitting to extract pupil features from the detected blobs while [26] proposed a method for pupil detection using a combination of color segmentation and edge detection, demonstrating robust performance in various lighting conditions. To further build on these approaches and adapt them to real-world non-ideal scenarios, [17] proposed a pupil detection method based on edge filtering and oriented histograms while [18] applied morphological operations and ellipse selection to build an eye-tracking system. However, with the rapid advancement in deep learning, most recent works tend to leverage neural networks for the task such as [7] in which the authors implemented a U-Net-based convolutional neural network architecture to segment the near-eye frame into four regions: background, sclera, iris and pupil. [31] proposed an add-on regression module based on [7] to extract eye features from segmentation in order to fit ellipses for pupil and iris.

Event-based Eye/Gaze Tracking Event cameras [39] have gained attention in the field of eye tracking due to their high temporal resolution, low latency, high dynamic range, sparse data acquisition, and asynchronous operation which eventually lead to capture rapid eye movements, such as saccades and microsaccades, with high precision and minimal motion blur, even in dynamic environments.

Early works on eye or gaze tracking based on event cameras were predominantly proposed to detect faces or eye blinks via recording the subject's face, upper body or whole body [35, 47]. Therefore, these setups were neither near-eye nor captured eye features or gaze features. Even though several works proposed to utilize near-eye setups to track eye and gaze starting from [3], most of these works relied upon RGB frames as well since their proposed pipelines need to take both frame and event data as inputs. In [3], the authors implemented a frame-based eye modelling pipeline and subsequently the captured event data was combined to update the eye model parameters. Eventually, the gaze direction was derived through a polynomial regressor. Following this work, numerous studies attempted to utilize both frame and event data for eye or gaze tracking such as [15, 36, 54, 58]. In [15], collected event data was utilized to predict regions of interest and U-Net-like architecture was then implemented to perform eye segmentation whereas in [36], the authors suggested to utilize stacked event frames along with RGB frames in parallel, to predict the gaze location via a quantification network of state transitions. [58] utilized a U-Net-based eye segmentation pipeline on the collected frames and then the binarized mask corresponding to the pupil area along with the event data were fed into a template-based pupil tracking stage. The prediction for the point of gaze was subsequently derived through a polynomial regressor. [54] was proposed to address the issue of occlusion in eye tracking studies which first interpolated RGB frames with the event data and then utilized a deep multi-scale spatial extraction-fusion network and an anti-blink pupil estimation module to extract semantic information from different scales and to deal with involuntary blinks respectively. However, due to the dependency on RGB frames, these works are unable to fully harness the benefits of the sparse, asynchronous and low-power characteristics of event cameras and thus, the need for an event camera-exclusive eye tracking system is still not properly addressed.

To this end, several recent works [6, 8, 37, 38, 48, 52] attempted to develop fully event-based eye tracking systems. In [48], even though the authors proposed an event-based eye tracking system without being dependant on RGB frames, their system heavily relied upon LEDs to generate glints as markers to execute corneal sphere regression and further the implemented coded differential lighting scheme on LEDs was limited by the sensor bias of the event cameras, especially when operating at high frequency. [37] proposed an convolutional neural network-based pupil tracking system operating on event frames while [8] proposed to replace convolutional neural network architecture with a change-based convolutional long short-term memory network for better performance. [6] further attempted to reduce the model complexity via implementing a spiking neural network and claimed better precision in localizing the pupils than [8] with fewer computational complexity. More recently, [38] proposed to utilize traditional kernel and ellipse fitting methods on event frames, which were accumulated over a fixed number of events, to extract pupil features and subsequently the pupil feature vector was fed into a recurrent neural network to predict the gaze location. However, all these methods still lack the operational capability to run real-time or near-real-time to predict pupil location using event data.

Event Accumulation Most works in the literature follow two approaches when it comes to event accumulation: a fixed time interval [32, 54] or a fixed number of events [3, 38, 58]. In the context of a fixed time interval, the scene dynamics, here the eye motion, may lead to fewer (if the eye moves slightly or does not move at all) or higher (if the eye moves fast) number of events in the accumulation process which eventually result in poor performance in downstream task due to scene instability occurred within a same time interval. On the contrary, the utilization of a fixed number of events seems better since the events are driven by the motion and thus a fixed number of events represents a consistent and stable amount of motion. Further, unlike the fixed time interval approach, the later approach also preserves the asynchronous nature of events by allowing the accumulation to be executed asynchronously. However, optimally determining the fixed number for later approach is challenging: if the selected threshold is too large, the accumulated events will present motion blur where as the threshold is too low, the accumulated events will lack the motion information [57]. In addition, a pure event count based approach does not consider the *informativeness* of the underlying events–e.g., whether they are generating by motion of multiple eye segments vs. motion of a single segment. Therefore, there is a need for an adaptive technique for event accumulation (or slicing) technique which is based on the spatiotemporal dynamics of the scene rather than on an artificially determined threshold.

The use of adaptive downsampling and accumulation for processing event data has been explored in few works. In [44] authors proposed an adaptive event camera that adjusts the event rate based on the scene's motion characteristics, leading to improved tracking performance in dynamic environments. Similarly, [57] introduced a method for adaptively accumulating events based on their relevance to the scene, reducing the data rate while maintaining important information for tracking tasks. These works demonstrate the effectiveness of adaptive downsampling and accumulation in improving the interpretability and efficiency of event data processing.

Eye Movements-based User Authentication User authentication based on eye movements has been explored as a biometric authentication method in numerous studies [16, 23, 25, 28, 30, 41, 42, 55]. Starting from [25] in which the eye movement biometric modality was introduced, earlier works such as [16] required to explicitly classify the eye motion signals into physiologically-grounded events and subsequently the manually-extracted features were fed into statistical models. With the advancement of deep learning, several works were proposed as end-to-end pipelines for eye movement biometrics. [23] presented a task-independent recurrent neural network-based architecture for human identification using gaze points whereas [42] utilized two convolutional subnets to separately focus on saccadic and fixational gaze movements. More recently, [41] claimed to acquire higher performance by utilizing a parameter-efficient DenseNet-based [22] deep architecture. However, all these works are explicitly dependant on the gaze estimations from an off-the-shelf eye tracker and therefore, neglect the potential of utilizing the motion dynamics of the pupil as a promising eye movement biometric.

To this end, only few works exist in the literature, such as [10, 30], in which the movement of pupil is utilized to derive the authentication features. [10] proposed a pupil movement-based system for personal identification number generation in which a set of classifiers were set to identify face, eye, eye-blinks and pupil movement. However this work is highly task-dependent and does not fully harness the motion dynamics of the pupil. In [30] authors investigated the use of eye movement characteristics, such as saccades and fixations, for user authentication and showed that these characteristics can serve as reliable biometric identifiers. They also highlighted the importance of capturing individual differences in eye movements for authentication purposes, which aligns with our motivation for using eye movements as biometric features. However, their system was still dependent on the gaze estimations from an eye tracker to extract the eye movement states in the eye movement classification block.

Katsini et. al. [27] surveyed the role of eye gaze in security and privacy applications. They discuss the evolution of eye tracking technologies and their application in biometric authentication, password entry, and privacy protection. Besides describing how eye tracking algorithms can be integrated into various devices such as smartphones and head-mounted displays, the authors also identify promising research directions and challenges for gaze-based security applications. More recently, Lien and Bhadhuri [40] explored the challenges and opportunities of biometric user authentication in the context of IoT devices. Their survey highlights the potential of biometric authentication to complement traditional knowledge-based methods. They categorize biometric traits into stable and volatile traits, a paradigm that aligns with our proposed approach of using natural eye kinematics as a stable biometric feature for enhanced, continuous and secondary authentication.

In summary, the related work highlights the potential of event cameras for eye-tracking applications, the effectiveness of adaptive event accumulation for processing event data, and the importance of pupil tracking and eye movement analysis for user authentication. Our work builds upon these existing approaches by proposing a novel system for accurate temporal tracking of microscopic eye movement using event cameras, and subsequently using such eye kinematic features to support secure and efficient authentication.

3 BACKGROUND AND MOTIVATION

In this work, our primary objective is to capture reflexive physiological eye movements with high temporal resolution while also ensuring that downstream image processing methods operating on *event frames* are able to track pupil movement with low computational complexity. In this section, we provide a brief background on event cameras and our rationale behind various design choices of our system *EyeTrAES* .

3.1 2D Virtual Event-frame Construction

Contrary to the conventional cameras (where the intensity of light across the visible spectrum incident on the sensor is captured at discrete points in time), event cameras or Dynamic Vision Sensors (DVS) only record changes in brightness (events) at each pixel asynchronously and with high temporal resolution, resulting in sparse data streams that encode motion and brightness changes in real-time. Event cameras are particularly beneficial in scenarios with either (i) low lighting/illuminance or (ii) high-speed motion where conventional cameras may suffer from motion blur and yield lower downstream task accuracy. In Section 8, we compare the impact of illuminance on the accuracy of downstream pupil movement-based user authentication (an exemplar application of our technique) as observed from RGB and event cameras. We show that under low lighting conditions (environment illuminance of 24 lux and near-eye illuminance of 8 lux), RGB-based pupil detection methods suffer a steeper reduction in user authentication of $\sim$ 45% while event cameras suffer a moderate accuracy drop of $\sim$ 33% when compared to the achievable accuracy in standard lighting conditions for both cameras (environment illuminance of 348 lux and near-eye illuminance of 65 lux). While event cameras also suffer from a reduction in user authentication accuracy due to the reduced illumination, they are able to recover $\sim$ 12% of user authentication capability due to the generation of events even under low-lighting conditions, while RGB-based pupil detection and authentication relies on color-based filtering techniques which fail under poor lighting conditions. We also show in Sections 4 and 8 how *EyeTrAES* leverages event cameras to adapt to different pupil kinematics patterns and high-speed pupil motion.

The event camera outputs a series of events on a per-pixel level – an event e_i ($i \in \mathbb{N}$) is denoted by a tuple (x_i, y_i, p_i, t_i), where (x_i, y_i) denotes the corresponding pixel coordinates where the event is generated, p_i represents the change in polarity (positive vs. negative), and t_i is the time of the corresponding event. To efficiently process the sparse stream of spatiotemporal asynchronous events, traditionally, the events are accumulated periodically over a time interval T or a fixed event volume N to generate a virtual frame. Subsequently, the appropriate vision techniques tailored to the specific downstream task (e.g., object detection or tracking) are applied. However, such simple motion-oblivious event accumulation techniques pose several drawbacks especially when tracking dynamic scenes with rapid movements: (i) blurred motion representation (especially when more than an ideal number of events are aggregated), (ii) loss of motion dynamics (occurred when the accumulated events do not accurately capture motion dynamics, i.e., fewer than necessary events are accumulated), and (iii) high computational load (event cameras can generate over 1 Gigaevents per second of data).

To visualize the importance of appropriate event accumulation, in Figure 2, we depict the 2D frame representation of the accumulated events over varying time windows for our targeted task of pupil tracking using a near-eye event camera. Each point in the image represents an event while the color of the point (blue vs. black) denotes the polarity. We illustrate two different pupil event frames, one for normal eye movement and the other including additional movement of the eyelashes during a blinking action. Figure 2(a) denotes the ideal event accumulation, where a single pupil is detected in the 2D frame (duration= 30 ms). As shown in Figure 2(b) and 2(c), accumulating fewer events (i.e., a frame with a lower duration= 10 ms) may not capture adequate motion artifacts, causing a failure to detect any pupil contours, whereas aggregating too many events (i.e., a frame with a larger duration= 100 ms) results in the detection of multiple pupils. Importantly, the ideal frame duration is not constant, but varies with the speed and the spatial dynamics of the pupil movement.

To further validate this point we explicitly take two scenarios where the subject is asked to have faster and slower eye movements in two different sessions. As shown in Figure 2(d) and 2(e), during slower eye movements,

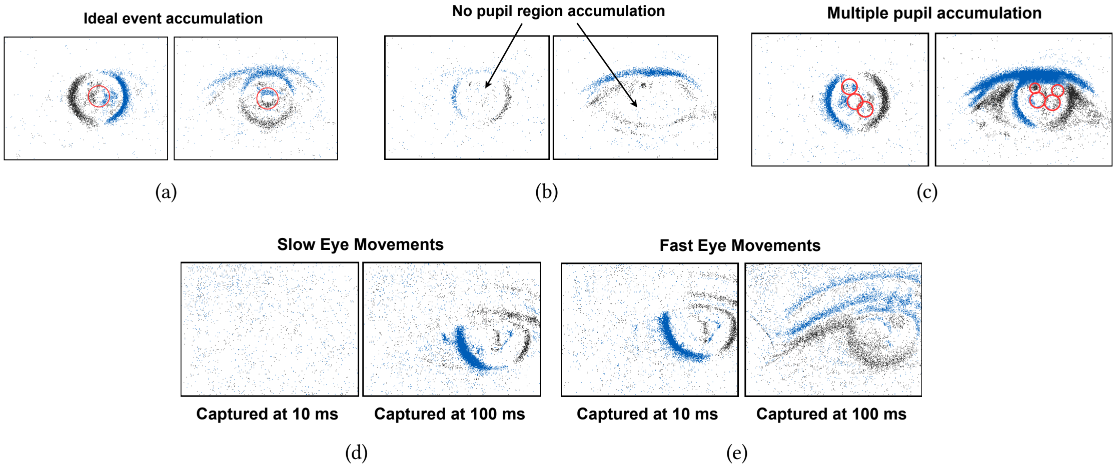


Fig. 2. Temporal Event Accumulation for both Normal and Eyelash-Flickering Eye Movement: (a) ideal event representation (duration = 30 ms), (b) Under-Accumulation (duration = 10 ms), (c) Over-Accumulation (duration =100 ms), (d) Slow eye movements captured at 10 and 100 ms, (e) Fast eye movements captured at 10 and 100 ms.

a smaller accumulation window can't capture the eye movements at all, while for a rapid eye motions having a longer accumulation window can lead to noisy framed representations.

This evidence raises the question: can the virtual 2D-frames be composed in a *motion responsive* manner, such that the spatiotemporal event stream is adaptively "sliced" (or aggregated) based on the underlying rate and spatial dynamics of the event stream? In Section 4.1, we shall introduce the technique for such *adaptive slicing*, as well as empirically demonstrate that natural variations in the speed of human pupil movement translate into significant variations in the "optimal" framing duration. We refer to "event slices" as "event frames" and "framed representations" interchangeably in this work.

3.2 Morphological Segmentation for Eye/Gaze Tracking

Conventional methods of RGB frame-based gaze tracking (captured using two RGB cameras) involve three steps: (i) morphological segmentation of the eye (i.e., segmenting eye parts using vision algorithms, such as canny edge detection), (ii) extraction of 2D eye features for tracking, and (iii) estimation of the direction of human gaze by mapping the 2D eye features extracted from *both left and right eyes* using geometric approaches.

Recent works on event-based gaze tracking involved event cameras that simultaneously capture *both* asynchronous event streams and corresponding RGB frames; for example, the DAVIS346 [2] camera records frames at 30 FPS. These works optimize the accuracy of gaze detection by using sparse RGB frames and initiate the pupil segmentation pipeline using vision algorithms to prime the event-based high-frequency pupil tracking (captured by two event cameras). Subsequently, a polynomial regressor is used to translate the features, representing pupillary information from both eyes, into 3D gaze direction.

We adopt an approach that differs from such prior work in the following ways:

- **Pupil Tracking:** In our *EyeTrAES* approach, we focus on tracking the pupil's spatial coordinates rather than the trajectory of the *gaze direction*: pupil trajectory refers to the movement of the pupil within the eye over time,

[2] https://inivation.com/wp-content/uploads/2023/11/2023-11-iniVation-devices-Specifications.pdf

while the gaze direction represents the direction in which a person is looking relative to their environment. While correlated, pupil trajectory and gaze direction represent different aspects of eye movement. For example, changes in lighting conditions or cognitive load can cause fluctuations in pupil movement without necessarily corresponding to changes in gaze direction. Similarly, reflexive eye movements, such as saccades or smooth pursuit, can cause rapid changes in gaze direction while the pupil trajectory remains relatively stable. Given our end goal of capturing fine-grained, microscopic and reflexive eye movements, we focus on pupil tracking in contrast to the dominant approach of gaze tracking. Additionally, we shall show (Section 8.2) that we can achieve user authentication with higher accuracy and shorter observational period (often less than 120-200 ms) via the use of pupillary kinematic features, instead of gaze-based features.

- **Single Near-Eye Event Camera:** As we rely on tracking the reflexive, physiologically-driven spatiotemporal dynamics of the pupil, we require the use of only a single near-eye event camera. Our approach of tracking a single pupil is based on the assumption that users demonstrate ideal conjugate eye movements, reflecting synchronized ocular motions consistent with typical oculomotor function. Such single pupil tracking also allows us to reduce the computational complexity of the event processing pipeline. This choice differs from the conventional approaches, across both RGB and event-based methods, that employ dual-camera setups.
- **Exclusive Event-Based Pupil Tracking:** While most prior work utilizes both RGB frames and event streams for gaze tracking, a few recent works have focused on event-only data processing. E-gaze [38] proposes an approach for purely event sensing-based gaze estimation based on the accumulation of event data into virtual *event frames*. In E-Gaze, after segmenting the different parts of the eye, the proposed approach leverages kernel density to find the pupil center. We adopt a similar approach, using contour detection together with Kalman filtering to support reliable tracking of pupil location over consecutive frames. However, we shall demonstrate that our method is computationally cheaper, incurring ∼70% lower latency compared to E-Gaze. More recently, Retina [6] integrates a spiking neuron network (SNN) and a specialized hardware accelerator (SynSense Speck [3]) for energy-efficient eye tracking of the pupil from near-eye events. However, such hardware accelerators are not widely available, precluding practical execution of SNN pipelines on current resource-constrained wearable devices.

3.3 Eye Movements for Biometric Authentication

Besides demonstrating an improvement in pupil micro-movement tracking, we shall also use eye-movement based user authentication as an exemplar to illustrate the application-level benefits of *EyeTrAES*. Our approach to biometric authentication is based on the assumption that the reflexive involuntary eye movements of different individuals, in response to simple naturally occurring visual stimuli, are distinctive and driven by natural physiological variations (e.g., in ocular muscle strength). Prior work, such as [20, 24], has used statistical features extracted from the position, velocity and acceleration profiles of the *gaze* sequence for applications such as task classification or user authentication. Individuals are assumed to exhibit distinctive patterns for gaze-related artifacts, such as saccades, fixations, and smooth pursuits. For *EyeTrAES*, we aim to use lower-cost *pupil movement-based* proxies for artifacts such as fixations and saccades; more specifically, instead of attempting to explicitly identify such artifacts, we compute and utilize statistical features, such as the velocity and acceleration profiles for the pupil trajectory, of a single eye.

4 *EYETRAES* OVERVIEW & FUNCTIONAL DETAILS

We now describe the core components (illustrated in Figure 3) in *EyeTrAES*: namely, the (i) event-based data acquisition and adaptive slicing that helps create appropriately informative 2D framed representations, and (ii)

[3] https://www.synsense.ai/products/speck-2/

lightweight, signal processing based pupil segmentation and tracking that operates on a sequence of such 2D event frames.

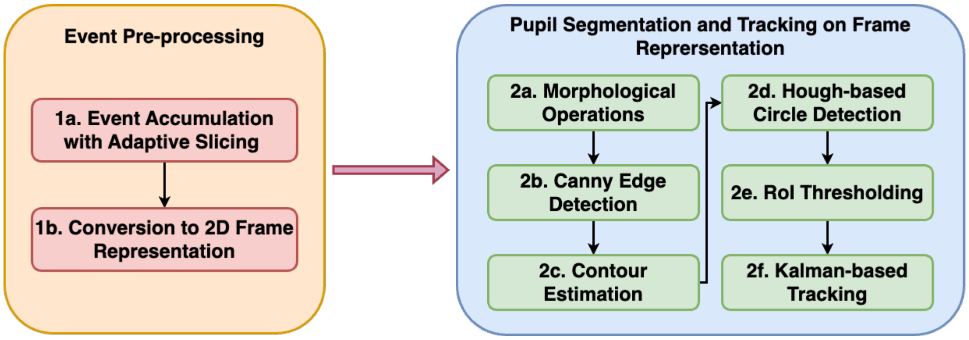

Event Pre-processing
1a. Event Accumulation with Adaptive Slicing
1b. Conversion to 2D Frame Representation
Pupil Segmentation and Tracking on Frame Reprersentation
2a. Morphological Operations
2b. Canny Edge Detection
2c. Contour Estimation
2d. Hough-based Circle Detection
2e. RoI Thresholding
2f. Kalman-based Tracking

Fig. 3. *EyeTrAES*: Block Diagram of Sub-Components.

At a high level, *EyeTrAES* first ingests high-volume event data captured at $O(\mu sec)$ latency and determines the appropriate rate (or boundary) at which events are accumulated into a 2D framed representation. Such processing deviates from standard event accumulation techniques, which aggregate events either over a fixed/periodic time horizon or fixed event volume, and allows *EyeTrAES* to autonomously adapt to the rate of change of information (using statistical measures) being captured by the event camera. *EyeTrAES* then collates the events into a 2-channel frame (one for each polarity) for further processing. The rest of *EyeTrAES's* Pupil Tracking pipeline then evaluates these 2D framed representations for (i) pupil segmentation using traditional OpenCV methods for canny edge detection [45] and circle detection of the pupil using the Hough algorithm [46], and (ii) tracking of the segmented pupil using a Kalman-based Centroid tracker.

4.1 Events Preprocessing with Adaptive Slicing

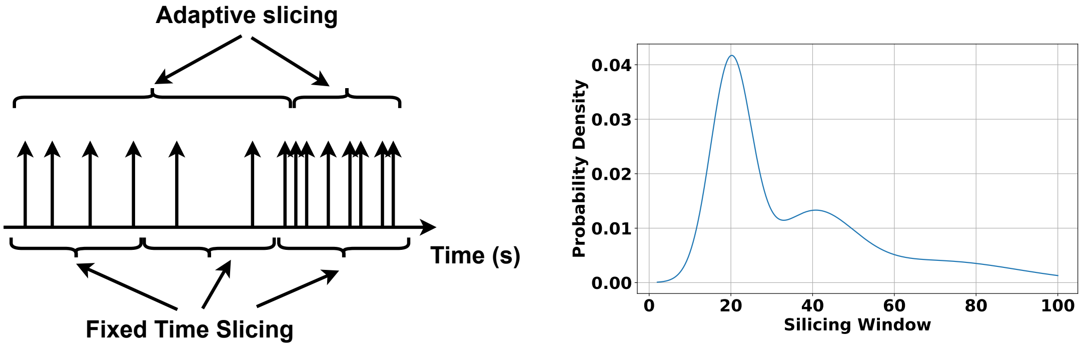
We employ a neuromorphic event camera to capture asynchronous events representing changes in luminance. The event camera provides high temporal resolution and low latency, making it suitable for capturing rapid eye movements, such as saccades and microsaccades. The event data is streamed continuously and processed in real-time. In addition, the event camera also generates RGB frames at the nominal frame rate of 30 FPS.

The Event Pre-processing component of the *EyeTrAES's* pipeline, illustrated in orange in Figure 3, accumulates the arriving asynchronous events to create *event frames*. The adaptive slicing-based accumulation adjusts the event accumulation rate based on the content of the events stream, ensuring that an event frame is *right sized* to capture just an adequate amount of data, as well as to reduce the overall frame rate. As summarised in Algorithm 1, our method involves iterating through the continuous event stream and calculating (i) the sliding mean and (ii) the standard deviation for each pixel's polarity change. (These statistics are computed using the changes in the absolute values of each event's polarity attribute.) If the standard deviation surpasses a predefined threshold *th* (set to 0.001, based on empirical studies), we segment the event stream into a *slice*; otherwise, we continue accumulating events. We also incorporate a downsampling factor of 2 to decrease the statistical mean and standard deviation computations, aiding in more efficient data processing while retaining crucial information. Our proposed slicing strategy leads to a variable slicing duration (i.e., a variable inter-frame gap): as illustrated in Figure 4(a), the slice window is smaller in case of rapid eye movements (larger event volumes with higher polarity variations) and longer in case of slower eye movements (lower event volumes).

For visual reference, in Figure 4(b) we plot the probability density of the length of slices (in ms) in our adaptive slicing method using the publicly available Ev-Eye dataset [58] (described in detail in Section 5.1). As we can see, during involuntary eye movements, both slower (smaller event volume) and rapid (larger slices) eye movements

Algorithm 1 Adaptive Event Slicing

```
Require: Continuous stream $E_{(x,y,p)}$, threshold $thresh$
Initialize running mean and standard deviation $\mu_{running} = 0$, $\sigma_{running} = 0$
Initialize Slice as an empty set
Initialize Frame as a matrix of size $H \times W$ filled with zeros
Set downsample factor $d = 2$
for each $(x, y, p)$ in $E$ do
    Add $(x, y, p)$ to Slice
    if $x\%d == 0$ or $y\%2 == 0$ then
        $p_f = |p|$
        Frame$[x, y] = p_f$
        $\mu_{current} = \text{mean}(F)$
        $\sigma_{current} = \sqrt{\frac{1}{H\times W}(\mu_{current} - p_f)^2}$
        $\sigma_{running} = (1 - \frac{1}{\text{Number of events}})\sigma_{current} + \frac{1}{\text{Number of events}}\sigma_{current}$
        if $\sigma_{running} > thresh$ then
            return Slice
        end if
    end if
end for
```



(a)

(b)

Fig. 4. (a) Event splitting with adaptive slicing vs default fixed time slicing, and (b) Probability density across different slicing window length (in ms).

occur more organically, indicating that adopting a motion-agnostic fixed-time or fixed-event volume-based slicing techniques may fall short in achieving our overall goal of accurately capturing intricate spatiotemporal eye movement dynamics.

4.2 Pupil Segmentation and Tracking

Most prior work (e.g., [3, 58]) has utilized a frame-based pupil segmentation approach, with event data serving merely as a supplementary input to refine the segmented pupil region. However, when the initial template for

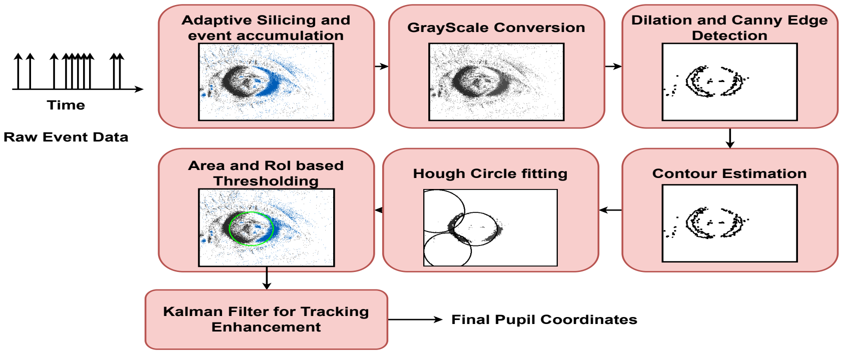

Time
Raw Event Data
Adaptive Silicing and event accumulation
GrayScale Conversion
Dilation and Canny Edge Detection
Contour Estimation
Hough Circle fitting
Area and RoI based Thresholding
Kalman Filter for Tracking Enhancement
Final Pupil Coordinates

Fig. 5. Event-based pupil segmentation.

the segmented pupil region is inaccurate, the addition of event information can introduce more noise, potentially degrading the overall pupil segmentation performance. In this work, we introduce a novel, fully event-based pupil segmentation approach (illustrated in blue in Figure 3) over the accumulated framed representation of events. As shown in Figure 5, the framed representation of the adaptively accumulated events primarily exhibits two distinct colors: blue, symbolizing positive polarities, and black, representing negative polarities. However, due to the inherent sparsity of data in the captured events, we utilize a sequence of preprocessing steps (described next) to enhance the visibility of relevant features. The output of each component in this sequence is also illustrated in Figure 5.

Morphological Operations: Initially, we convert the frame representation to grayscale to simplify subsequent processing steps. A dilation morphological operation is applied to expand the foreground event regions, facilitating better feature extraction.

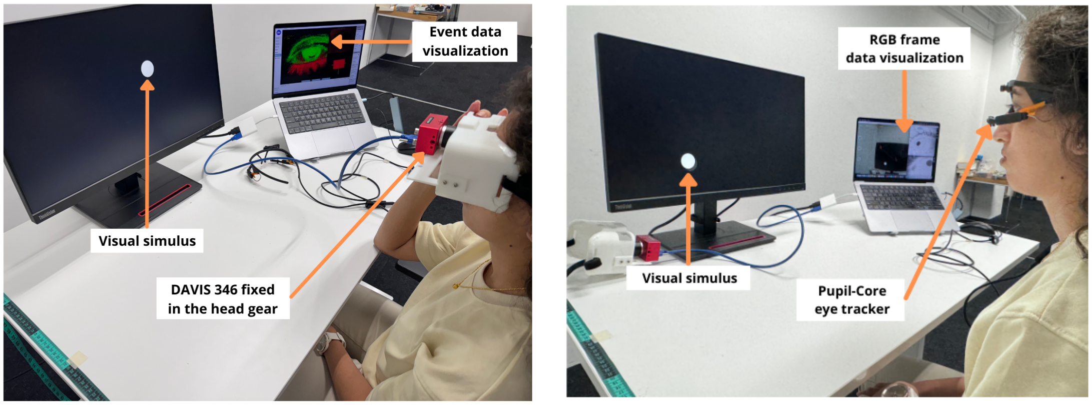
Canny Edge Detection: Subsequently, the dilated frame undergoes canny edge detection [45], a widely used technique for identifying sharp transitions in pixel intensity, thereby delineating the boundaries of significant event contours. This step is crucial for isolating regions of interest corresponding to meaningful event occurrences within the frame.

Contour Estimation: Further refinement is achieved through contour estimation, wherein continuous boundaries connecting adjacent pixels are identified by analyzing variations in pixel intensity. This process enables the extraction of precise contours outlining significant event regions, laying the groundwork for subsequent analysis.

Hough-based Circle Detection: We employ the classical Hough technique [46] used for identifying and characterising complex shapes such as pupils accurately. This technique is particularly robust against gaps in curves and noise, making it well-suited for our application. By detecting circles that approximate the contours of interest, we can effectively estimate unknown boundaries and discern potential pupil regions within the frame.

RoI Thresholding: We next apply area and region-based thresholding techniques over the detected contours to isolate candidate elliptical shapes resembling human eye pupils. By imposing constraints on the size and shape of the detected regions, we enhance the precision of pupil localization (eliminating false positives), thus improving the overall accuracy of our system.

Kalman-based Tracking: We incorporate a Kalman filter to refine the pupil detection process further and mitigate the effects of noise. By recursively updating and refining the estimated pupil centres across successive frames, the Kalman filter helps denoise the detections and improve the stability of the overall tracking process. This adaptive filtering approach ensures robust and reliable pupil localization, even in the presence of varying lighting conditions and occlusions.



(a)

(b)

Fig. 6. Data collection setup using (a) DAVIS346, (b) Pupil-Core eye tracker.

5 DATASETS, USER STUDIES AND PERFORMANCE METRICS

We now describe our evaluation methodology, which consists of both using (i) pre-existing, ground truth-annotated datasets to evaluate the fidelity of *EyeTrAES* in terms of tracking low-level pupil movement feature and (ii) new, in-lab user studies to quantify *EyeTrAES's* performance in terms of continuous user authentication (an illustrative application of *EyeTrAES* that we shall detail in Section 7). Table 1 summarizes the key differences between the two datasets, Ev-Eye, and *EyeTrAES*, that we shall now describe. We emphasize that the pre-existing data is collected under fairly artificial conditions with no head movement, whereas our *EyeTrAES* data is collected under more natural, less-restrictive conditions and thus enables us to study a richer set of user artifacts.

5.1 Ev-Eye Dataset for Eye Kinematics

We first use the published Ev-Eye dataset [58] to evaluate the performance of *EyeTrAES* in terms of its ability to support fine-grained eye movement tracking. Ev-Eye consists of 48 participants' data collected using the DAVIS346 camera that provides both event and RGB (30 FPS) data. For data collection purposes, they used a visual stimulus with a solid red circle (that disappears and reappears at a different randomly chosen spatial location of the screen for 1.5s) displayed on the monitor to guide the gaze movement of the subject. Along with the DAVIS RGB-frames and events, the dataset also includes (i) the reference Point of Gaze (PoG) captured at relatively high frequency (~100 Hz) using Tobii Glasses, (ii) pupil segmentation/localization on events data, and (c) dense gaze references for eye movements, such as fixation, saccades, and smooth pursuits.

The Ev-Eye dataset is especially useful in evaluating the fidelity of *EyeTrAES's* Pupil Segmentation and Tracking component, as it provides ground truth annotations for the user gaze. In addition, its inclusion of Tobii tracker data will allow us to compare *EyeTrAES's* event camera-based user authentication accuracy against that obtained using high frequency, ground truth eye tracking data. However, the Ev-Eye dataset was collected under highly controlled conditions where each user had their head position fixed and resting on a chin-rest.

5.2 *EyeTrAES* User Study & Dataset

While evaluation on Ev-Eye helps establish the benefits of novel *EyeTrAES* components, such as dynamic event slicing, we also need to evaluate *EyeTrAES's* efficacy in terms of low-level eyeball tracking and our eventual exemplar application—user authentication–especially under varying task contexts and more diverse environmental conditions (e.g., low lighting levels, more natural macro-head movements). To achieve this objective, we executed a separate data collection user study, comprising 40 participants, in a laboratory setting. For this study, an individual participant wore only one device (either our event camera-embedded custom headset or the commercial Pupil-Core eye tracker [26]) at any one instant, implying that it was not possible to obtain concurrent ground truth for low-level eye kinematics.

Table 1. Summary of the Datasets

Dataset	**Devices**	**Data Format**	**Experiment Setup**	**End Goal**	**Ground Truth Annot.**
Ev-Eye	DAVIS346	• Events • Grayscale frames (25 FPS)	• **Two** Event cameras • Stationary; Near-eye • Fixed distance b/w visual stimuli and the user	Gaze tracking	• Point of gaze from Tobii Pro Glasses 3 • Pupil segmentation on events data • Annotations for saccades, fixations, and smooth pursuit using gaze references
Eye-TrAES	DAVIS346	• Events • Grayscale frames (25 FPS)	• **Single** Event camera • Head mounted (wearable); Near-eye • Mobile	Eye movement tracking	• Grayscale images (120 FPS) from Pupil Core tracker • Point of gaze from Pupil Core tracker

As illustrated in Figure 6, our data collection had two stages. In the first stage (as depicted in Figure 6(a)), the participants wore a custom-built headgear fixed with a DAVIS346 camera [4] secured around the forehead using a Velcro fastener. The camera was positioned adjacent to the right eye, while the participants were directed to track the visual stimuli using their left eye. In the second stage (as depicted in Figure 6(b)), the participants wore the off-the-shelf Pupil-Core eye tracker [26] that is widely used by the academic research community. The eye tracker uses two near-eye cameras, oriented towards the wearer's eyes, and one world camera, facing outwards, to respectively capture the pupil's view and location, as well as the scene that engages the participant's gaze. By following numerous studies in the literature [9, 11, 33, 43], we design our study protocol to elicit natural eye movements: the visual stimulus appears at the top left corner of the screen and then moves continuously in random directions such that the stimuli exhibits nearly-perfect and smooth collisions when it hits an edge of the screen. The stimulus remains consistent across all participants. To guide the gaze movement of the participants, we displayed the visual stimulus on a 1920 × 1080, 23.8-inch monitor. The distance between the monitor and the participant varied between 45cm and 50cm, resulting in a field of view between $56^\circ \times 34^\circ$ and $62^\circ \times 37^\circ$.

Our sample consists of 40 participants, including 28 males and 12 females, representing diverse ethnic backgrounds (i.e., from 7 different nationalities). Their ages range from 21 to 32 years, with a mean of 26.08 years and a standard deviation of 2.99. The participants had perfect (20/20), contact lens-based corrected or corrective

[4] https://inivation.com/wp-content/uploads/2019/08/DAVIS346.pdf, Accessed: October 28, 2024

spectacles-assisted vision with the percentages being 47.5%, 10% and 42.5% in the participant pool respectively. To ensure that an individual participant was able to exhibit meaningful eye movement, only nearsighted participants were included (i.e., if the participants wore corrective spectacles for any other ocular condition, they were excluded from the study). Nearsighted participants were instructed to remove their corrective spectacles during the experiment (after ensuring that the participants can follow the visual stimuli in their field of vision without any difficulties) since (1) the presence of the spectacles affects the fit of the event camera mounted head-gear and (2) the glint or reflections caused by the spectacles potentially affect the event camera's ability to accurately capture the pixel intensity changes. All participants were recruited through university-wide announcements and local community outreach, ensuring a diverse sample in terms of gender and ethnicity. Inclusion criteria required participants to have a normal or corrected-to-normal vision, verified by a pre-study screening whereas the individuals with significant ocular health issues, those unable to use corrective lenses effectively, or those with conditions other than near-sightedness were excluded.

Ethical Considerations: Our study was approved by the Institutional Review Board (IRB) of our institution, ensuring that it met ethical standards for research involving human subjects. Furthermore, the participants received a monetary incentive for their participation as guided by IRB protocols. All participants were provided with detailed information about the study, including its purpose, procedures, and any potential risks. If the participant felt unable to perform the specified activities, or if the specified activities made the participant feel uncomfortable, the participant was excluded from the study. In addition, participants could choose to withdraw from the study at any time (before or during the data collection) if they felt uncomfortable.

The participants were first seated comfortably on a chair; before the actual data collection, the wearable devices were calibrated mechanically to ensure an optimal capture of each participant's eye region through several steps including (1) adjusting the attached Velcro fastener to ensure the proper fit of the head gear containing the event camera, (2) mechanically adjusting the focal length of the event camera lens such that it is optimal to capture the eye movement dynamics with minimal blur, (3) adjusting the positions and angles of three embedded cameras on the sliding arms in Pupil-Core eye tracker to ensure that the expected views are optimally collected and (4) calibrating and validating the Pupil-Core eye tracker using a 5-point calibration paradigm to ensure that the average calibration error is below 1.5° as measured by the accompanying Pupil-Labs software [26]. The participants had the freedom to move their heads and bodies as they wished, without the need to maintain fixed positions. Throughout both stages, the visual stimulus consisted of a solid white circle against a black background displayed on the monitor, with a diameter of 80 pixels (A video recording of the utilized visual stimuli on the screen is presented in the corresponding repository).

Each session per participant consists of four trials, each lasting four minutes. In the first two trials, the participants wore the DAVIS346 camera; in the last two trials, they wore the Pupil-Core eye tracker. Between every two consecutive trials, there was a resting period of at least 30 seconds to reduce visual fatigue to the participant. The randomized movement pattern of the white circle was identical across a cross-device trial pair (spanning both the DAVIS346 and Pupil-Core device) but varied between the two trials corresponding to the same wearable device. Due to their endeavor to focus their gaze on the white circle, each participant predominantly exhibited smooth pursuit and fixation states when the circle was moving smoothly, while saccadic states were triggered by the occasional discontinuous “jump" in the location of the white circle.

5.3 Key Performance Metrics

We evaluate *EyeTrAES* vs. alternative competitive baselines using multiple metrics that together capture both our primary goal of accurate pupil detection and our secondary, application-level goal of accurate per-user authentication.

For the pupil detection task, we have adopted two key evaluation metrics:

- *Intersection over Union (IoU)*: This metric, widely employed in pupil region segmentation [58], quantifies the overlap between the estimated and ground truth pupil regions.
- *Dice Coefficient*: This metric, commonly used in eye segmentation tasks [58], gauges the similarity between the estimated and ground truth pupil regions.

For eye movement feature-based user authentication (the illustrative application (detailed in Section 7) we choose to demonstrate the efficacy of our proposed *EyeTrAES* technique), assess performance primarily through the per-user authentication *Accuracy* metric, which counts the number of correct user authentications over all the authentication instances. We also compute the False Acceptance Rate (FAR) and the False Rejection Rate (FRR), as well as the *Equal Error Rate* (EER), which is a commonly used metric for biometric authentication, representing the point on a Receiver Operating Characteristic (ROC) curve where FAR equals FRR. The FAR quantifies the probability of an unauthorized person being incorrectly identified as an authorized user; a low FAR is essential to reduce the risk of unauthorized access. The FRR, also referred to as False Non-Match Rate (FNMR), represents the likelihood of an authorized user being incorrectly rejected by the system. A low FRR is critical for ensuring a positive user experience, particularly in scenarios where the authentication process is frequently used, such as unlocking a mobile device. High FRR can lead to user frustration and reduced system usability.

6 *EYETRAES'S* PERFORMANCE OVERVIEW: PUPIL SEGMENTATION

In this section, we first explain the baseline methods we choose to evaluate the efficacy of *EyeTrAES* . Then we provide various quantitative and qualitative analyses to show the superior performance of *EyeTrAES* , specifically on accurate pupil segmentation under various system parameters. Our results are generated on the publicly available Ev-Eye dataset [58].

6.1 Baselines for Pupil Segmentation

For the pupil segmentation task, we consider the following three baselines:

(1) **Ev-Eye [58]:** A hybrid event and frame-based pupil segmentation method, where, the grayscale images from the event cameras are used to segment the pupil region (using conventional CV methods for segmentation and denoising) to generate the pupil boundaries that can be used as pupil template. In our analysis, we use the pupil template centers (this specific annotation is provided in the Ev-Eye dataset) as one of the baselines.

(2) **E-Gaze [38]:** This approach on purely event sensing-based pupil segmentation uses a non-parametric statistical method called two-dimensional kernel density estimation (KDE) to locate the centre of two concentric circles, one representing the iris and the other one representing the pupil region, within an event frame.

(3) **DAVIS346-RGB-30Hz (RGB Frame-based pupil extraction):** For fair comparison, we also introduce a purely RGB frame based pupil segmentation baseline, where we re-implemented *EyeTrAES's* pupil tracking techniques (described in §4.2) on the RGB frames in the Ev-Eye dataset (more precisely, the grayscale representation of RGB images released in Ev-Eye), captured by a DAVIS346 camera. As an initial step, we employ an additional color filtering technique on these grayscale images to isolate regions of interest containing black contours, which typically represent pupils. Following the extraction of black contours, we execute *EyeTrAES's* standard steps of (i) using the Hough technique [46] to identify elliptical shapes within these regions of interest, thereby segmenting the boundaries of candidate pupil objects, and then (ii) applying RoI thresholding (on both the area and aspect ratio of the detected contours) to precisely isolate the pupillary contour.

6.2 Performance of Pupil Segmentation

As summarized in Table 2, *EyeTrAES* achieved higher IoU (≈ 92%) and dice coefficient (≈ 89%) compared to other methods on the Ev-Eye dataset. Please note that *EyeTrAES's* adaptive slicing achieves the nominal frame rate of

Table 2. Performance evaluation of different pupil segmentation methods (Ev-Eye Dataset).

Methods	MAE	IoU (%)	Dice Coeff. (%)	Latency (s)
EyeTrAES	10.13	92	89	0.0047
Ev-Eye	14	89	88	0.71
DAVIS346-RGB-30Hz	16	84	81	0.008
E-Gaze (Fixed 2000 # of events)	26	48	44	0.012
E-Gaze with adaptive slicing	12	87	85	0.012

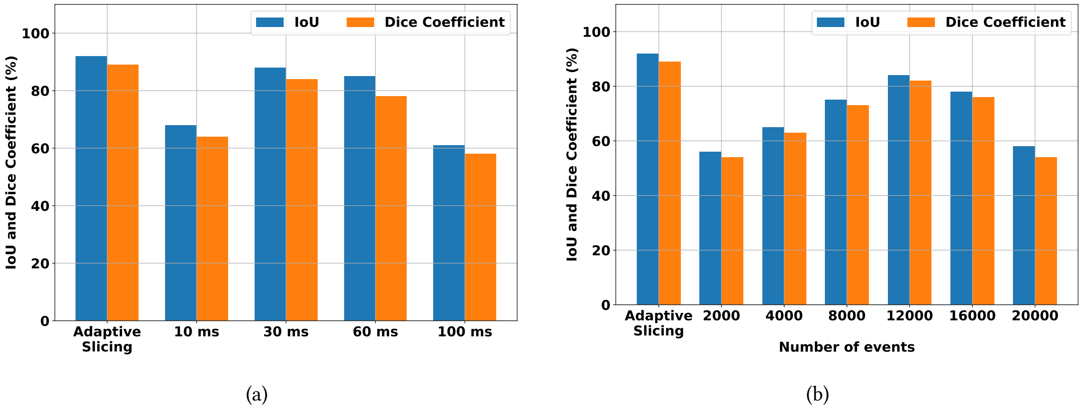
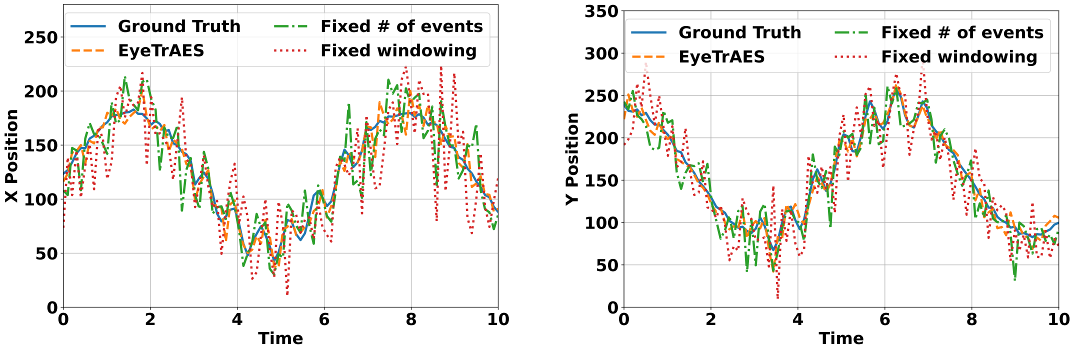


Fig. 7. IoU and Dice Coefficients comparison of *EyeTrAES* against various (a) fixed-time-based framed representations, (b) fixed-event volume-based framed representations.

30 FPS. The Ev-Eye and E-Gaze (with our proposed adaptive slicing for event accumulation) also demonstrated competitive performance with an IoU of 89% and 87%, respectively, but as we shall see shortly, their computation load is much higher. Note, E-Gaze is also a purely event-based approach and it considers a fixed event volume of 2000 events to form the framed representation [38], however, we have demonstrated how varying the slicing approach can impact the pupil segmentation performance later in Section 6.2.4. The purely frame-based approach (evaluated using the DAVIS RGB-frames from the Ev-Eye dataset) has a lower accuracy as the pupil segments are generated using a standard image processing-based approach, which can sometimes lead to false positives in pupil detection. Also, as the Ev-Eye frame data is captured at a lower temporal resolution (≈ 30 FPS) with the DAVIS346 camera, the captured frame can have unstable motion artifacts during periods of rapid eye motion, leading to poorer accuracy compared to the event-based approaches where the intensity of the data stream increases in proportion to the velocity of eye movement.

6.2.1 Pupil Segmentation under Different Slicing Windows. To demonstrate the benefits of adaptive slicing, we next evaluate the performance of *EyeTrAES* vs. alternatives that utilize a fixed window (of either time or event count). Our results, shown in Figure 7(a), indicate that adaptive slicing outperforms fixed slicing windows across different scenarios. For instance, when using a 10 ms slicing window, the IoU is 68% and the Dice Coefficient is 64%. However, with adaptive slicing, the IoU increases to 92% and the Dice Coefficient to 89%. For low values of the slicing window and slow eye movement (fewer events), the eye region may not be fully captured in the framed representation, leading to lower accuracy. Conversely, using a larger slicing window with more events



(a)

(b)

Fig. 8. Comparison of pupil coordinates over time using different slicing strategies. The figure illustrates the performance of adaptive event slicing (AES) versus fixed-number-of-events (8000–16000) and fixed slicing window lengths (30 ms). AES consistently provides a more accurate representation of pupil coordinates, effectively capturing dynamic eye movements.

may result in multiple overlapping pupil regions captured in a single event frame, introducing more noise and reducing the accuracy of the pupil region detection.

6.2.2 Pupil Segmentation under Different Number of Events. We also analyze the impact of varying numbers of events on pupil segmentation accuracy. The number of events is varied from 2000 to 20, 000, and the segmentation accuracy is compared with an adaptive slicing-based approach. Figure 7(b) illustrates that the IoU and Dice Coefficient values demonstrate superior performance when the number of events is greater than 8000 and less than 16000. However, in scenarios with lower eye movement between two rapid eye movements, using a fixed number of events can result in an unstable framed representation. This instability is characterized by the accumulation of two random pupil regions generated at the two extreme timestamps of the slice cut, with intermediate minor events generated due to lower eye movements. In contrast, adaptive slicing waits until it detects a rapid eye motion before slicing out the events, leading to a more stable framed representation. Consequently, adaptive slicing achieves superior IoU and Dice coefficients compared to fixed-number-of-events-based slicing.

To further substantiate our findings, we conducted a qualitative evaluation using 10 seconds of randomly selected data from the dataset, comparing pupil coordinates over time using our adaptive slicing method against fixed-number-of-events (ranging from 8000 to 16000) and a fixed slicing window length of 30 ms. The results, illustrated in Figure 8, reveal that our adaptive slicing method consistently outperforms both fixed slicing strategies. The adaptive approach more accurately captures dynamic pupil movement and adjusts to varying eye motion rates, thereby providing a more precise representation of pupil coordinates over time. Notably, while fixed-number-of-events approaches demonstrate slightly better performance than fixed slicing windows of 30 ms, the improvements are marginal compared to the significant gains achieved with adaptive slicing. This qualitative analysis, based on a randomly selected 10-second segment of the dataset, underscores the advantages of our adaptive method in maintaining high accuracy and stability across diverse eye movement scenarios.

6.2.3 Computation Latency. As demonstrated in Table 2, *EyeTrAES* is also superior in terms of segmentation latency compared to all the baselines, incurring an average computational latency of 4.7 ms. In contrast, the closest baseline, DAVIS346-RGB-30Hz, takes almost twice as long (average=8 ms), as it involves multiple image processing steps including color based filtering, morphological operations and contour detection on a denser

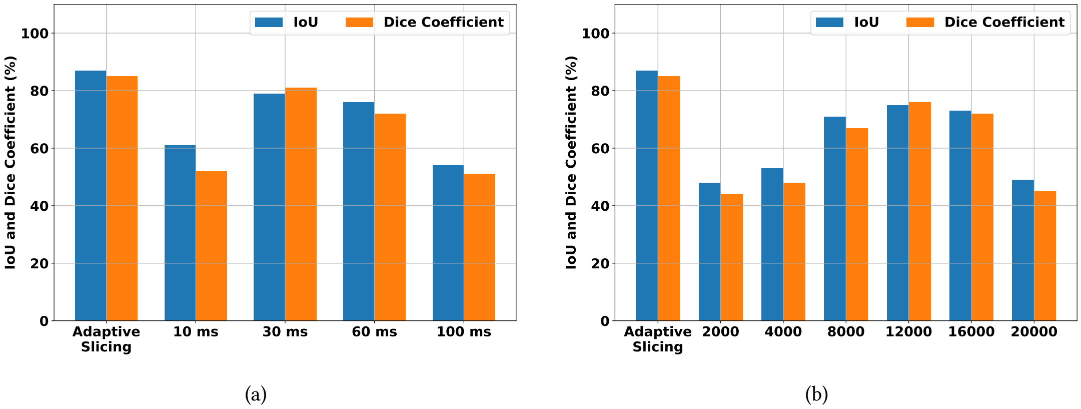


Fig. 9. Performance of E-Gaze with different slicing strategies: (a) with different slicing window length, (b) with different number of events.

pixel representation; note that the IoU of DAVIS346-RGB-30Hz is ~10% lower than that of *EyeTrAES*. For E-Gaze, the latency is almost 3× larger (average~=12 ms), as it utilizes two separate concentric circle fitting steps (for both the pupil and iris regions), while *EyeTrAES* applies a Hough based circle detector only once. Not surprisingly, Ev-Eye has the largest latency (average~=710 ms, almost 150× of that of *EyeTrAES*), as it first uses a computationally complex U-Net model to identify candidate pupil segments, followed by additional candidate point subset estimation to filter out noisy events caused by the movement of eyelashes and eyelids.

6.2.4 Impact of Event Slicing on Other Pupil Segmentation Baselines. To evaluate the effectiveness of our proposed adaptive slicing technique in conjunction with an existing pure event-based pupil segmentation baseline such as E-Gaze [38], we conducted a series of experiments. These experiments involved varying the fixed slicing window lengths or the fixed volumes of events. We then compared the IoU and Dice coefficients for pupil segmentation using these methods against our adaptive slicing strategy. As depicted in Figure 9, our adaptive slicing method significantly improves event aggregation within the framed representation, resulting in superior IoU and Dice coefficients compared to fixed slicing strategies. Notably, E-Gaze captures 2000 events to form the framed representation [38] which performs the poorest on the Ev-Eye dataset.

The results collectively demonstrate not just the superiority of our combined adaptive slicing and lightweight pupil segmentation techniques, but also show that adaptive event slicing helps improve the tracking accuracy of other prior computationally-heavier segmentation baselines. After witnessing how our combined adaptive event slicing and light-weight pupil segmentation technique achieves highly accurate and low-latency pupil localization (segmentation), we shall now explore how *EyeTrAES* allows us to extract high temporal resolution pupillary kinematics to support improved biometric user authentication (as an exemplar application).

7 EXEMPLAR APPLICATION: *EYETRAES*-BASED USER AUTHENTICATION

We now proceed to show how *EyeTrAES* -based pupil segmentation and tracking can be used to provide improved user authentication. Our key hypothesis is that the fine-grained micro-movements of the pupil, that naturally occur during regular viewing activities, vary across individuals (due to variations in ocular muscle strength), and can thus serve as a biometric fingerprint.

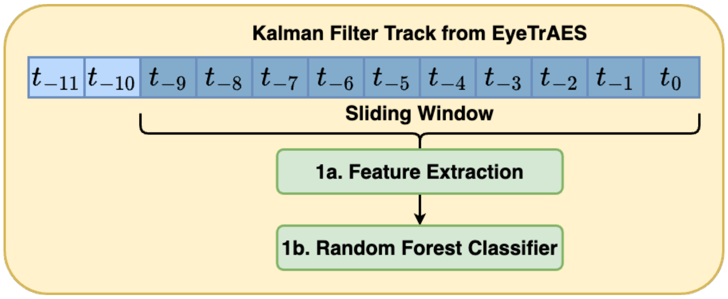

Kalman Filter Track from EyeTrAES
t_{-11} t_{-10} t_{-9} t_{-8} t_{-7} t_{-6} t_{-5} t_{-4} t_{-3} t_{-2} t_{-1} t_0
Sliding Window
1a. Feature Extraction
1b. Random Forest Classifier

Fig. 10. *EyeTrAES*-based User Authentication: Block Diagram of Sub-Components.

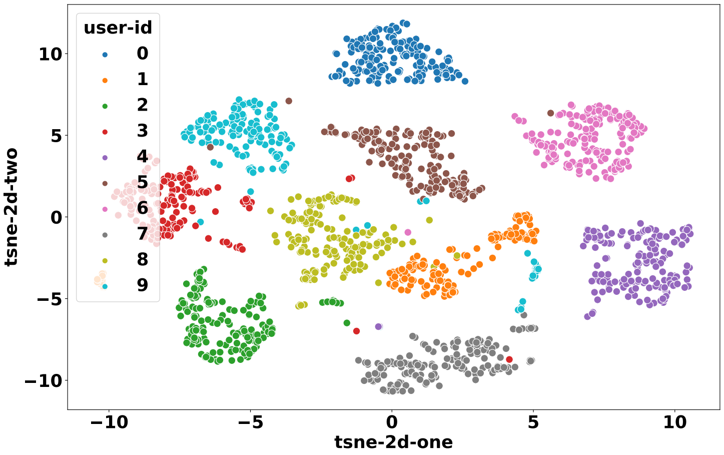


Fig. 11. t-SNE distribution of the computed features across 10 subjects from the Ev-Eye dataset in 2D space.

7.1 Pupil Movement Feature Extraction for User Authentication

As discussed in previous works [2, 56], individuals often exhibit distinctive saccadic and micro-saccadic eye movements which can be used to authenticate users. However, to compute saccades or micro-saccades, one must have the actual gaze location or the screen coordinate on the screen. Our approach of near-field single eye tracking cannot provide such a gaze estimate. Instead, in the final component of *EyeTrAES's* pipeline, illustrated in Figure 10 , we utilize the segmented, Kalman-filtered sequence of pupil locations in a sliding window of size=10 to extract two additional features related to the kinematics of *pupil movement*, using such features as a microscopic proxy for gaze-related artifacts such as saccades, described next.

(1) *Pupil Velocity*: From the tracked pupil coordinates (say x_i, y_i at time t_i), we first compute the first-order derivative of the pupil coordinates $(v_{x,i}, v_{y,i}) = (\frac{x_i - x_{i-1}}{t_i - t_{i-1}}, \frac{y_i - y_{i-1}}{t_i - t_{i-1}})$, signifying the velocity or the relative change in the pupil coordinates. This derivative provides an approximation to saccadic or fixating movements of the eye. For example, while a user has a saccadic eye movement, the relative change in the successive pupil coordinate will be much higher than fixation.

(2) *Pupil Acceleration*: We then compute the second order derivative of the pupil coordinates, deriving acceleration values $(a_{x,i}, a_{y,i}) = (\frac{v_{x,i} - v_{x,i-1}}{t_i - t_{i-1}}, \frac{v_{y,i} - v_{y,i-1}}{t_i - t_{i-1}})$.

7.2 Feature Vector and Random Forest Classifier

The process above creates a tuple of pupil (velocity, acceleration) values for each pair of consecutive event frames–i.e., for the i^{th} frame, we obtain not only the pupil position (x_i, y_i), but also the velocity $(v_{x,i}, v_{y,i})$ and acceleration $(a_{x,i}, a_{y,i})$ values. To clasify an individual user, we then concatenate M consecutive (position, velocity, acceleration) triples, creating an $M \times 3$-dimensional feature vector representing *microscopic pupillary motion* attributes over relatively short time windows. Our *EyeTrAES* implementation uses an empirically derived value of M =10, effectively representing pupil movement-related features predominantly over 100-400 ms time windows. This feature vector is then input to a Random Forest classifier with 100 decision trees, which is trained in a supervised fashion to support binary (one-vs.-rest), per-person classification.

While we defer detailed evaluation of authentication accuracy till later, we now present initial results that validate our hypothesis about the distinctiveness of pupillary kinematic features, such as velocity and acceleration. We compute these features across 10 selected subjects from the Ev-Eye dataset [58] and study the t-distributed stochastic neighbour embedding (t-SNE) [50], a statistical method for visualizing high-dimensional feature distribution in a lower-dimensional space (with 2 dimensions in this case). The 'X' and 'Y' axes indicate the first and second dimensions resulting from the dimensionality reduction process. As observed from the Figure 11, different subjects have distinctive, *non-overlapping* distributions of the features, *strongly suggesting that these features can be used to authenticate individual subjects.*

8 *EYETRAES'S* PERFORMANCE: USER AUTHENTICATION

In this section, we evaluate *EyeTrAES's* performance on user authentication on both Ev-Eye and *EyeTrAES* datasets. While we report the overall aggregate performance results using Ev-Eye dataset, we use *EyeTrAES* dataset to (a) performed more detailed studies on the sensitivity to various framing/slicing techniques, and (b) study the impact of various contextual/ambient conditions on the authentication accuracy.

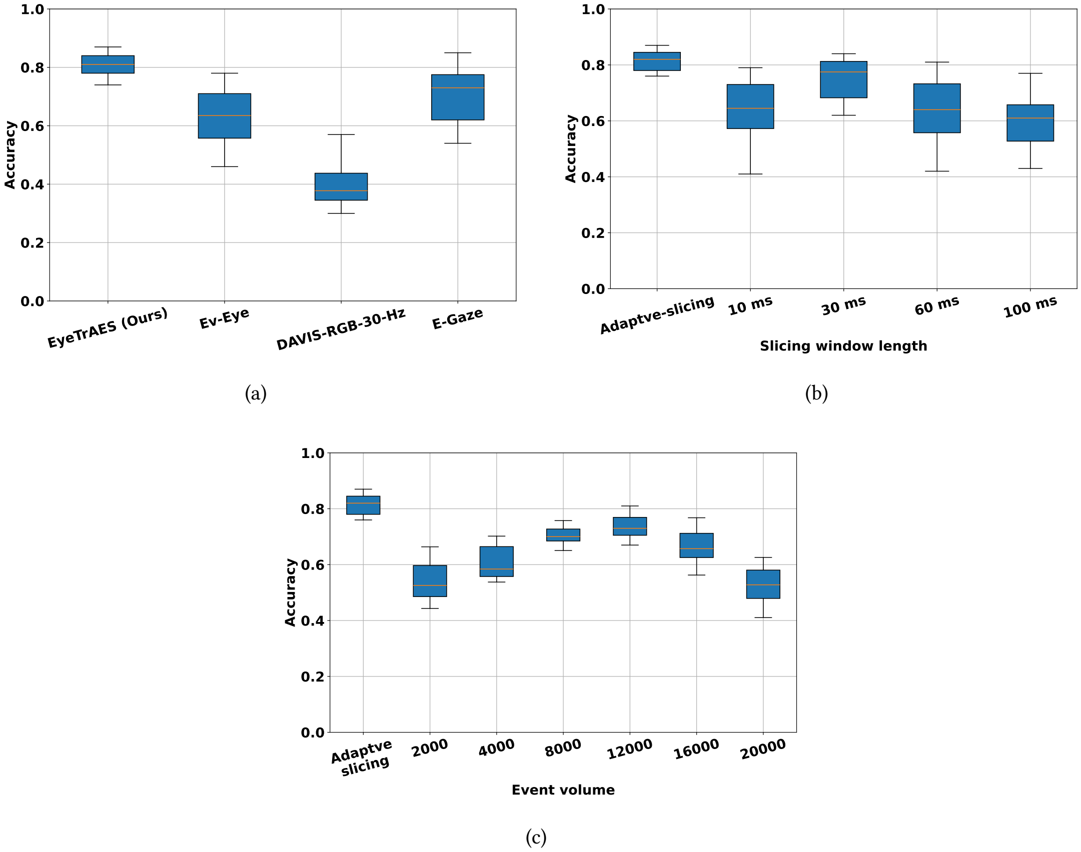
8.1 User Authentication Performance on Ev-Eye Dataset

We conduct a comprehensive comparison of the accuracy achieved by different baselines for eye movement feature-based user authentication, as depicted in Figure 12(a). *EyeTrAES's* performance is superior to all baselines, achieving an accuracy between 0.78 to 0.87, with an impressive median accuracy of 0.82. Both Ev-Eye and frame-based methods rely on the grayscale frame data for pupil segmentation captured at a lower temporal granularity at 30 FPS. They are thus unable to capture the fine-grained saccadic and micro-saccadic eye movement features which we believe to be key components of the biometric fingerprint, and consequently have lower accuracy compared to *EyeTrAES* and E-Gaze, both of which utilize event frames. While *EyeTrAES* relies on adaptive slicing-based framed representation generation, E-Gaze relies on a fixed-event volume based framing. Thus, both these methods are able to broadly generate the eye movement features in sync with the dynamics of the eye movement, and consequently provide superior accuracy over frame-based pupil segmentation approaches. However, for E-Gaze, the accuracy in segmenting out the pupil region is poor as it identifies the pupil segment only when it captures *both* the concentric circles of the iris and pupil. These findings highlight the effectiveness of our proposed method in achieving superior performance in eye movement-based user authentication scenarios.

8.1.1 Performance of Event-based Approach under Different Fixed-time Slicing Window.

For the event-based approach, the slicing technique we use has a significant impact on the ability to estimate the kinematics of the pupil movement, thereby affecting the overall accuracy. This effect is demonstrated in Figure 12(b), where the impact of adaptive slicing on authentication accuracy is compared against different motion agnostic fixed-time windows.

In Figure 12(b), we see that adaptive slicing method achieves significantly higher accuracy, followed by slicing periodically at every 30ms intervals. Slices of shorter (i.e, 10ms) and longer (e.g., 100ms) achieve lower accuracy:



(a)

(b)

(c)

Fig. 12. User authentication accuracy evaluated on Ev-Eye dataset (a) under different approaches, (b) for the event-based approach with different slicing windows, (c) for different numbers of event accumulation.

a smaller slicing window of 10ms can result in very sparsely populated event frames under slower eye movement, whereas a larger slicing window of 30ms or higher can cause fast eye movements to overlap, leading to noise in the computed saccadic and micro-saccadic eye features.

8.1.2 Performance of Event-based Approach under Different Fixed-event Volume Slicing. We now evaluate the impact of adaptive slicing on authentication accuracy and compare the performance against different fixed-event volume slices. As shown in Figure 12(c), the overall accuracy of fixed-event volume slicing is lower compared to adaptive slicing based framed representation. The primary reason behind this can be attributed to the fact that, similar to the fixed-time window, the fixed event volume-based pupil segmentation approach also fails to accurately capture the pupil kinematics, as discussed in Section 6.2.2. The highest median accuracy is achieved when an individual event frame accumulates 12000 events; not surprisingly, this value is where the pupil segmentation accuracy is also much more accurate (Figure 12(c)).

8.1.3 Comparison of Classifier Performance for User Authentication: In this section, we compare the performance of various classifiers for user authentication, focusing on both accuracy and latency. We evaluated the following

Table 3. Accuracy and latency of user authentication across different classifiers.

Classifier	Median Accuracy (%)	Latency (ms)
Random Forest	82	12
SVM	76.4	20
RBF Network	78.1	25
Gradient Boosting	79.3	15

Table 4. Feature abalation study

Feature Combination	Median Accuracy (%)
Position only	44.5
Velocity only	53.4
Acceleration only	55.7
Position and Velocity	59.2
Position and Acceleration	58.3
Velocity and Acceleration	71.4
Position, Velocity, and Acceleration	82

classifiers: Random Forest, Support Vector Machines (SVM), Radial Basis Function (RBF) Networks, and Gradient Boosting Trees. Our goal was to identify the classifier that provides the best balance of accuracy and latency for real-time applications.

The classifiers were trained and tested on the Ev-Eye dataset comprising position, velocity, and acceleration features of the selected pupil region. The performance metrics used for comparison were accuracy and latency.

From the results as shown in Table 3, we can observe that the Random Forest classifier outperforms the other approaches in terms of both accuracy and latency. The higher accuracy and lower latency make Random Forest the most suitable choice for real-time user authentication applications. The Random Forest classifier achieved an median accuracy of 82.0% and the lowest latency of 12 ms. Its ability to handle high-dimensional data and provide robust predictions makes it an ideal choice for this application. The SVM classifier achieved an accuracy of 82.5% with a higher latency of 20 ms. While SVMs are effective for classification tasks, their higher computational cost makes them less suitable for real-time applications compared to Random Forest. The RBF Network showed an accuracy of 80.3% and a latency of 25 ms. Despite its ability to model complex relationships, its performance was not competitive with the other classifiers. The Gradient Boosting classifier achieved an accuracy of 83.1% with a latency of 15 ms. While its performance was close to that of Random Forest, the slightly higher latency made it less favorable for real-time applications.

8.1.4 Feature Ablation Study. To understand the contribution of different features to the performance of our user authentication system, we conducted a feature ablation study. We evaluated the model's accuracy by considering various combinations of position, velocity, and acceleration vectors of the eye pupil region. This study helps to highlight the importance of each feature and their combined effect on the model's performance. We tested the following exhaustive feature combinations: (i) Position vectors only; (ii) Velocity vectors only; (iii) Acceleration vectors only; (iv) Position and velocity vectors; (v) Position and acceleration vectors; (vi) Velocity and acceleration vectors; and (vii) Position, velocity, and acceleration vectors.

As shown in Table 4 using only position vectors yielded a median accuracy of 44.5%. While position data provides basic information about eye movements, it lacks the dynamic aspects captured by velocity and acceleration.

Considering only velocity vectors improved the accuracy to 53.4%. Velocity captures the rate of change in position, providing more insight into the movement dynamics. Using acceleration vectors alone resulted in an accuracy of 55.7%. Acceleration captures changes in velocity, adding another layer of dynamic information.

Combining position and velocity vectors resulted in an accuracy of 59.2%. The addition of velocity data to position vectors significantly improved the model's performance. The combination of position and acceleration vectors yielded an accuracy of 58.3%. While better than using position alone, it was slightly less effective than combining position and velocity. Using both velocity and acceleration vectors improved the accuracy to 71.4%. This combination captures both the rate of change and the changes in the rate of change, providing a richer representation of pupil movements.

The best performance was achieved by combining all three features, with a median accuracy of 82.0%. This indicates that higher-order pupil movement features, such as velocity and acceleration, significantly enhance user authentication performance.

In the following section, we further investigate and provide more in-depth analyses on user authentication accuracy using our own *EyeTrAES* dataset.

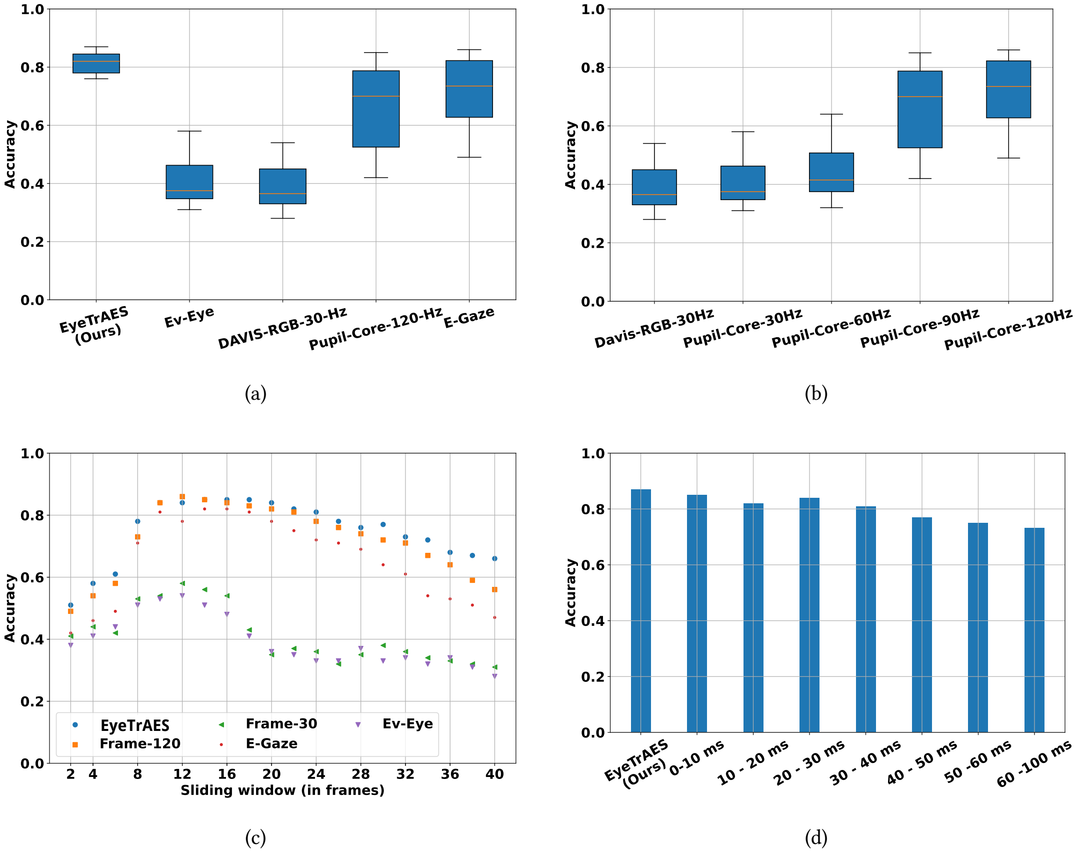
8.2 User Authentication Performance on *EyeTrAES* Dataset

In this section, we discuss the overall user authentication accuracy evaluated on our *EyeTrAES* dataset. We use the same set of baselines described in §6.1, with the slight modification for **Ev-Eye** where we use the published Ev-Eye U-Net-based pupil segmentation model to re-train on *EyeTrAES* dataset for pupil segmentation and localization. In addition to the previous baselines, we include we add a few variations of **DAVIS346-RGB-30Hz** baseline. In particular, we leverage the grayscale images captured by the Pupil Core eye tracker at the nominal frame rate of 120 FPS, and create additional baselines at the down-sampled frame rate of 30, 60, and 90 FPS. These additional baselines will help us understand the efficacy of RGB-based methods in user authentication at higher frame rate, as opposed to the proposed event-based *EyeTrAES* approach which has a lower nominal frame rate.

As depicted in Figure 13(a), *EyeTrAES* demonstrates higher accuracy than other alternatives, achieving a median accuracy of 0.82. Ev-Eye trained on our dataset demonstrated a significantly lower median accuracy of 0.43. Additionally, the frame-based method with 30 FPS exhibited an accuracy range of 0.34 to 0.58, with a median accuracy of 0.41. However, as expected, the authentication accuracy for the RGB frame-based approach increases with increasing frame rate, with 120 FPS RGB streams resulting in a median accuracy of 0.7. Higher frame rates can capture the rapid saccadic and micro-sacaddic eye movements with greater precision, compared to a lower frame rate of 30 FPS. E-Gaze also demonstrated moderately good performance, with a median accuracy of 0.74, attributed to its precise detection of the pupil region compared to other baseline approaches. Both Ev-Eye and DAVIS-RGB-30Hz utilize frame data captured at a lower rate=30 Hz, and thus have lower accuracy compared to the other baselines.

In addition to visualizing the performance differences between our proposed method and the baseline methods, we conducted a statistical analysis (using a t-test) to compare the accuracies achieved by *EyeTrAES* against those of the baseline methods. The t-test results as shown in Table 5 demonstrate that the pupil tracking accuracy of *EyeTrAES* is statistically significantly different than the accuracies achieved by the other alternatives, such as Ev-Eye, DAVIS-RGB-30Hz, Pupil-Core, and E-Gaze, with p-values less than 0.001. These findings confirm that the accuracy gains of our approach are indeed statistically significant.

8.2.1 Performance of Frame-based Approach under Different FPS. To understand the impact of different frame rates on user authentication accuracy, we use the Pupil-Core grayscale frames captured at 120 FPS and downsample them to {90, 60, 30} FPS by dropping the relevant intermediate frames. As demonstrated in Figure 13(b) increasing the frame rate has a direct impact on the ability to estimate the kinematics of the eye movements, leading to higher accuracy.



(a)

(b)

(c)

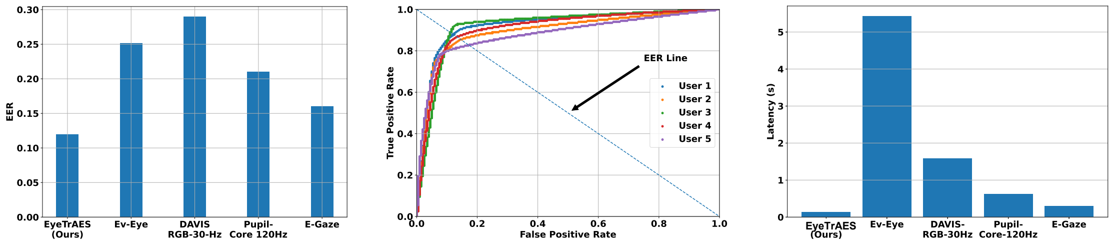
(d)

Fig. 13. User authentication accuracy evaluated on *EyeTrAES* dataset: (a) different baselines, (b) RGB frame-based approach at different FPS, (c) Average accuracy of *EyeTrAES* for varying sliding window length, and (d) different slicing window regions from the adaptive slices.

Table 5. T-test Results Comparing Proposed Method and Baseline Methods

Comparison	p-value
***EyeTrAES* vs. Ev-Eye**	2.51×10^{-15}
***EyeTrAES* vs. DAVIS-RGB-30Hz**	6.09×10^{-17}
***EyeTrAES* vs. Pupil-Core**	6.41×10^{-11}
***EyeTrAES* vs. EGaze**	0.0000156

8.2.2 Impact of Sliding Window Length. Having established the superiority of *EyeTrAES* in extracting the pupil kinematics features even with the lower nominal frame rate, we next evaluate the impact of the length of the sliding window (over which the kinematic features are computed) on authentication accuracy. Figure 13(c) depicts the average user authentication accuracy across varying sliding window length (measured in number of frames) of *EyeTrAES* and baselines. The results indicate that the authentication accuracy generally improves with larger



(a)

(b)

(c)

Fig. 14. (a) Equal Error Rate (EER) for different approaches, (b) ROC curves across different users, (c) Response time in a valid authentication under each method.

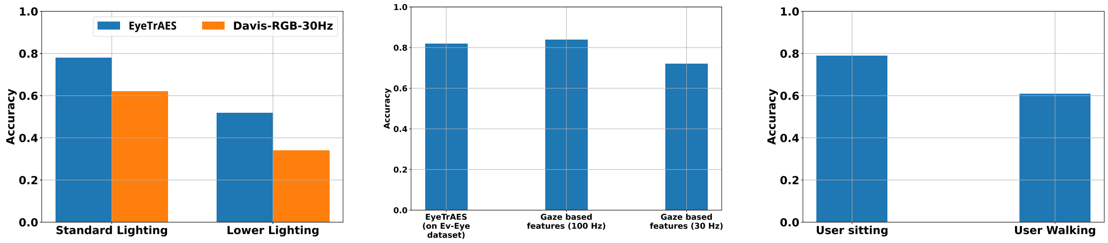
sliding window length, reaching a peak median accuracy of 0.87 for *EyeTrAES* at a window length of 16 frames. However, beyond a certain point, increasing the window length leads to a decrease in accuracy as a larger window size leads to over-accumulation of features, potentially representing different states of eye movement. A higher sliding window will also lead to a longer duration (i.e., lower responsiveness) for successful user authentication. Overall, a sliding window length of 10 seems to be a suitable choice, achieving classification accuracy of 82% and providing a relatively low authentication response time.

These findings suggest that the choice of sliding window length has a significant impact on the system's performance, and selecting an appropriate window length is crucial for achieving optimal accuracy.

8.2.3 Impact of Individual Slicing Factors. We next study how our *EyeTrAES's* adaptive slicing technique affects the authentication accuracy associated with different slice durations. We consider slice windows in 7 distinct ranges: 0−10, 10−20, 20−30, 30−40, 40−50, 50−60, and 60−100 ms. Because each authentication vector comprises 10 slices, possibly of varying duration, we first use *dominant class labeling* to assign each authentication vector a specfic *slice label*–i.e., the label corresponding to the modal slice duration. Figure 13(d) plots the average accuracy achieved by *EyeTrAES* across different window ranges. We observe that the average accuracy is higher for samples corresponding to dominant slices < 40 ms, with the accuracy progressively dropping slightly for samples with dominant slices > 60 ms. More importantly, we see that the accuracy is relatively constant (≈0.8±0.03) across all ranges, indicating that *EyeTrAES* is reasonably successful in preserving salient eye movement features across different ranges.

8.2.4 User Re-authentication Performance. To verify a user's identity, we have also used the Equal Error Rate (EER), which is a commonly used metric used for biometric authentication systems. The EER represents the point on a Receiver Operating Characteristic (ROC) curve where the False Acceptance Rate (FAR) is equal to the False Rejection Rate (FRR). As shown in Figure 14(a), compared to other baselines, *EyeTrAES* performs better with a lower EER rate. A high EER indicates that the system is unable to effectively balance between false acceptances and false rejections. This can lead to authentication issues if impostors are frequently accepted or inconvenience if legitimate users are frequently rejected. Overall, *EyeTrAES* shows an average EER of 0.12, as also shown by the ROC curves in Figure 14(b) across 5 selected users from our dataset.

8.2.5 Authentication Responsiveness: We have also calculated the authentication response time, which refers to the time required for the classifier to successfully authenticate a legitimate user for the first time, across all methods. This computation excludes the initial bootstrapping of the sliding window for generating features, assuming that the process has already been completed. Instead, we focus on the time taken from the computation



(a) (b) (c)

Fig. 15. Mean Accuracy of (a) *EyeTrAES* on Ev-Eye dataset, and with gaze-based features collected at 100 Hz and downsampled to 30 Hz, (b) under standard lighting conditions and at poor lighting conditions, (c) *EyeTrAES* while the subject is sitting in the default setting vs when the subject is walking.

of the latest pupil segmentation to the prediction time until a valid authentication is achieved. The response time for *EyeTrAES* -based successful authentication is a maximum of 0.14 sec. As illustrated in Figure 14(c), *EyeTrAES* outperforms the baselines in terms of such response time. Ev-Eye exhibits the longest response time at ≈ 5.4 seconds due to the inherent latency of its DNN-based pupil segmentation, while E-Gaze shows the shortest response time among the baselines at around 0.3 seconds. The frame-based methods benefit from a higher frame rate (FPS) of 120, leading to faster accumulation of eye movement features and a response time of 0.62 seconds. In contrast, the default 30 FPS captured by DAVIS346 results in slower feature accumulation and a response time of 1.6 seconds.

8.2.6 Performance under Different Lighting Conditions. To evaluate the performance of *EyeTrAES* under different lighting conditions, we explicitly collected additional data for a typical subject (id: 1) under poor lighting conditions. The light level is measured using a digital luminance light meter (LX1330B[5]). The measured light level is 24 Lux in the environment under poor lighting conditions, dropping to 8 Lux near the eye after wearing the custom headgear. We compare this subject authentication accuracy with the data collected in the default lighting condition, where the measured environmental and near-eye illuminance was 348 Lux and 65 Lux, respectively. Since Pupil-Core uses IR LEDs for eye tracking, its performance remains unaffected by lighting changes. We compared *EyeTrAES* vs. the RGB-based baseline (DAVIS346-RGB-30Hz) under these different ambient lighting conditions. As shown in Figure 15(a), under poor lighting conditions, subject 1's authentication accuracy drops by 33% to achieve accuracy ∼=62%, as the rate of event generation also gets impacted under poor lighting conditions. However for DAVIS346-RGB-30Hz, the performance degradation is even more pronounced, with accuracy dropping by ≈ 45% to ∼36%. This drop occurs as the RGB method relies on segmenting out the pupil coordinates with color-based filtering techniques, which fails under poor lighting conditions.

8.2.7 Performance from the Perspective of Usable Security. In the context of biometric authentication, the evaluation of performance metrics such as False Acceptance Rate (FAR) and False Rejection Rate (FRR) is crucial for assessing both the security and usability aspects of the authentication system. Our proposed method demonstrates superior FAR performance compared to existing approaches, ensuring robust security. Specifically, our method achieved a FAR of 12%, which is significantly lower than the typical FAR values observed in similar systems, often ranging from 20% to 34%. This reduction highlights the enhanced security capability of our method, making it well-suited, especially as a secondary, continuous authentication mechanism, for environments where security

[5] https://drmeter.com/products/lx1330b-digital-illuminance-light-meter

is a primary concern. Our method also achieved an FRR of 9%, which is lower compared to the FRR values of alternative systems ranging from 11% to 32%. This balance between FAR and FRR is a critical factor in designing an effective biometric authentication system. More specifically, if used in a wearable device as a form of continuous, secondary authentication, having a lower FRR than FAR (as exhibited by *EyeTrAES*) is more critical. In addition, the annoying sporadic failure of legitimate authentication attempts can be further reduced by declaring an authentication failure only after multiple incorrect attempts: with a simple threshold of 3 consecutive failures, our effective FRR reduces to $\sim 7x10^{-4}$–i.e., roughly a failure rate of less than one in a thousand. Our method strikes an effective balance between these metrics, demonstrating superior performance in terms of both security and usability.

Table 6. Comparison of FAR and FRR Across Different Baselines

Method	**FAR (%)**	**FRR (%)**
EyeTrAES (Ours)	12	9
Ev-Eye	25	21
DAVIS-RGB-30-Hz	34	32
PupilCore-120-Hz	20	11
E-Gaze	24	19

8.3 Impact of Gaze vs Eye Movement-related Features in Authentication

For biometric eye gaze-based authentication, previous works [20, 24] have proposed the Point of Gaze (PoG) coordinates, velocity, and acceleration of gaze changes as primary features. To understand how these extracted features perform on user authentication, we use the Ev-Eye dataset, where PoG coordinates are logged along with the event data using a Tobii Pro Glass 3[6]. From these gaze coordinates, we generate the features and pass them on to the classifier in two different settings: one where the gaze coordinates are passed at the default rate of collection, i.e., 100 Hz, and the other where the coordinates are downsampled to 30 Hz, emulating the sampling rate of an RGB frame or event accumulation window of 33 ms. From Figure 15(b), we observe that gaze-based classification, under the default sampling rate of 100 Hz, offers the highest accuracy (~84%), as such high frequency data can better capture the accadic and micro-saccadic movements. Note, however, that gaze-based methods require concurrent sensing of both eyes, as well as additional knowledge of the viewing distance. However, if the gaze data is downsampled to 30 Hz (to mimic the sampling rate of a standard DAVIS346 RGB frame or event accumulation window of 33 ms), *EyeTrAES* provides superior accuracy. *EyeTrAES's* event-based pupil tracking is smoother and less noisy due to its use of a Kalman filter; in contrast, downsampled gaze coordinates have more discontinuity in the extracted gaze features, leading to slightly lower accuracy.

9 DISCUSSION

While our results demonstrate that *EyeTrAES* provides significant enhancements to current capabilities pupil tracking and eye motion-based user authention, there are several open areas that require further investigation. *Single vs. Dual Eye Tracking*: *EyeTrAES* currently uses wearable based near-eye tracking of only one eye, and thus cannot directly take advantage of gaze-related features such as saccades and fixation. We believe that our approach of tracking a single pupil's movement suffices, as users typically tend to exhibit conjugate eye movement, moving both eyes in tandem. That said, it is possible that microscopic distinctions may exist between the pupillary movements of the right and left eyes, perhaps because of the differences in ocular muscle strength (most people

[6] https://www.tobii.com/products/eye-trackers/wearables/tobii-pro-glasses-3

have one dominant eye). There are two potential consequences of such possible distinctions. First, we may need to train separate *EyeTrAES* models for each eye, as the pupillary movement of the dominant vs. non-dominant eye may exhibit salient differences. Second, it may be possible to further improve the user authentication accuracy by combining pupillary motion features from both eyes, as they may collectively encode subtler person-specific variations than available from tracking a single eye. Studying the implications of eye-specific variations, especially for users who may suffer from certain eye impairments such as myopia or amblyopia (aka lazy eye), is needed to demonstrate the applicability over a broad population.

Reliable Classification under Different Motion Conditions: The *EyeTrAES* dataset was collected in a controlled lab setting, with the user seated comfortably in a chair and gazing at the stimulus displayed on a stationary screen. While users were not restricted, unlike in EV-Eye, to keep their head stationary, we should note that our studies did not capture eye movement behavior under various real-world motion conditions, such as different ambulatory states (e.g., running, climbing) or different vehicular usage (e.g., buses, trains). There are two distinct reasons by which the captured pupillary motion during such real-world conditions may differ from those observed in the *EyeTrAES* dataset. First, user movement can lead to continuous displacement of the wearable sensor relative to the human eye, leading in turn to noise in the captured event stream. This limitation is essentially due an imperfection in the sensing mechanism and can be overcome by simply ensuring a snug fit of the wearable device on the face. The second reason, however, is more fundamental: an individual's *pupil movement itself can be fundamentally altered due to such external context*—e.g., a user viewing a screen while walking may continuously glance, perhaps even without focusing their gaze, in multiple directions to maintain situational awareness. To perform a preliminary testing of this possibility, we collected additional pupil movement data, using our snugly-fitted wearable prototype, from a single subject while they were engaged in multiple activities such as walking on a treadmill or climbing stairs. As reported in Figure 15(c), we observed that *EyeTrAES's* authentication accuracy for this user drops from 79% (for test data collected while they are seated) to 61% (for test data collected during such activities). These preliminary results suggest that *EyeTrAES's* accuracy can be possibly enhanced by incorporating additional macro-motion features (e.g., captured by smartglass-mounted inertial sensors) into the authentication classifier.

Native SNN-based Processing of Events: As explained earlier, we adopt a strategy of frame-based event accumulation instead of processing the events natively using an SNN model, simply because of the current lack of embedded neuromorphic processors that can support efficient SNN execution. We adopted this decision because preliminary studies indicating that a software-based emulation of an SNN, using the SpikingJelly framework [14], is simply too slow and can process at most 9-10 "frames" per second, even on a powerful Jetson ORIN platform. Should neuromorphic processors become available, we anticipate that SNN-based approaches will prove to be more efficient, at least until the pupil segmentation and tracking stage. The resulting change in the frequency and accuracy of the stream of inferred pupil location data is likely to require a modification of the set of pupillary kinematic features and the corresponding Random Forest classifier model. This remains future work.

10 CONCLUSION

In this paper, we have presented a novel approach for fine-grained low-latency pupillary movement tracking using event cameras. Our approach, *EyeTrAES* , uses a novel adaptive event accumulation technique coupled with a light-weight pupil segmentation algorithm to track the eye pupil region with significantly higher accuracy and lower latency – pupil segmentation IoU score ∼=92% while incurring frame processing latency of only ∼4.7 ms. Further, as an illustrative application, we showcase the extracted microscopic eye kinematic features (such as pupil velocity and acceleration) from the high-fidelity pupil tracking data exhibit distinctive trends across users and can in turn be used as a means for robust user authentication. Our exemplar application, *EyeTrAES*-based user authentication, has several key advantages. Firstly, it offers high accuracy and reliability, as eye movements are unique to each individual and can serve as reliable biometric identifiers. Secondly, it provides a seamless and non-intrusive authentication experience, as users can be authenticated simply by looking at a screen or a camera. Thirdly, it is more robust to variations in lighting conditions and facial expressions, making it suitable for real-world applications. We have demonstrated the effectiveness of our approach through experiments and evaluations, showing that it outperforms traditional RGB camera-based authentication systems by achieving median user authentication accuracy ∼=0.82 and lower latency of ∼=12ms, showing its ability to achieve real-time on-device execution.

ACKNOWLEDGMENTS

This work was supported by both the National Research Foundation, Prime Minister's Office, Singapore under its NRF Investigatorship grant (NRF-NRFI05-2019-0007) and by the Ministry of Education (MOE) Academic Research Fund (AcRF) Tier 1 grant (Grant ID: 22-SIS-SMU-044). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore.

REFERENCES

[1] Richard A Abrams, David E Meyer, and Sylvan Kornblum. 1989. Speed and accuracy of saccadic eye movements: characteristics of impulse variability in the oculomotor system. *Journal of Experimental Psychology: Human Perception and Performance* 15, 3 (1989), 529.

[2] Sahar Mahdie Klim Al Zaidawi, Martin HU Prinzler, Jonas Lührs, and Sebastian Maneth. 2022. An extensive study of user identification via eye movements across multiple datasets. *Signal Processing: Image Communication* 108 (2022), 116804.

[3] Anastasios N Angelopoulos, Julien NP Martel, Amit P Kohli, Jörg Conradt, and Gordon Wetzstein. 2021. Event-Based Near-Eye Gaze Tracking Beyond 10,000 Hz. *IEEE Transactions on Visualization and Computer Graphics* 27, 5 (2021), 2577–2586.

[4] R. Asmetha Jeyarani and Radha Senthilkumar. 2023. Eye Tracking Biomarkers for Autism Spectrum Disorder Detection using Machine Learning and Deep Learning Techniques: Review. *Research in Autism Spectrum Disorders* 108 (2023), 102228. https://doi.org/10.1016/j.rasd.2023.102228

[5] Gary Bargary, Jenny M Bosten, Patrick T Goodbourn, Adam J Lawrance-Owen, Ruth E Hogg, and John D Mollon. 2017. Individual differences in human eye movements: An oculomotor signature? *Vision research* 141 (2017), 157–169.

[6] Pietro Bonazzi, Sizhen Bian, Giovanni Lippolis, Yawei Li, Sadique Sheik, and Michele Magno. 2024. Retina : Low-Power Eye Tracking with Event Camera and Spiking Hardware. arXiv:2312.00425 [cs.CV]

[7] Aayush K Chaudhary, Rakshit Kothari, Manoj Acharya, Shusil Dangi, Nitinraj Nair, Reynold Bailey, Christopher Kanan, Gabriel Diaz, and Jeff B Pelz. 2019. Ritnet: Real-time semantic segmentation of the eye for gaze tracking. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. IEEE, 3698–3702.

[8] Qinyu Chen, Zuowen Wang, Shih-Chii Liu, and Chang Gao. 2023. 3ET: Efficient event-based eye tracking using a change-based convlstm network. In *2023 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, 1–5.

[9] CJS Collins and Graham R Barnes. 2009. Predicting the unpredictable: weighted averaging of past stimulus timing facilitates ocular pursuit of randomly timed stimuli. *Journal of Neuroscience* 29, 42 (2009), 13302–13314.

[10] Indrajit Das, Ria Das, Shalini Singh, Amogh Banerjee, Md. Golam Mohiuddin, and Avirup Chowdhury. 2020. Design and Implementation of Eye Pupil Movement Based PIN Authentication System. In *2020 IEEE VLSI DEVICE CIRCUIT AND SYSTEM (VLSI DCS)*. 1–6. https://doi.org/10.1109/VLSIDCS47293.2020.9179933

[11] Gabriel Diaz, Joseph Cooper, Dmitry Kit, and Mary Hayhoe. 2013. Real-time recording and classification of eye movements in an immersive virtual environment. *Journal of vision* 13, 12 (2013), 5–5.

[12] Kai Dierkes, Moritz Kassner, and Andreas Bulling. 2019. A fast approach to refraction-aware eye-model fitting and gaze prediction. In *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications*. 1–9.

[13] Andrew T Duchowski, Krzysztof Krejtz, Izabela Krejtz, Cezary Biele, Anna Niedzielska, Peter Kiefer, Martin Raubal, and Ioannis Giannopoulos. 2018. The index of pupillary activity: Measuring cognitive load vis-à-vis task difficulty with pupil oscillation. In *Proceedings of the 2018 CHI conference on human factors in computing systems*. 1–13.

[14] Wei Fang, Yanqi Chen, Jianhao Ding, Zhaofei Yu, Timothée Masquelier, Ding Chen, Liwei Huang, Huihui Zhou, Guoqi Li, and Yonghong Tian. 2023. SpikingJelly: An open-source machine learning infrastructure platform for spike-based intelligence. *Science Advances* 9, 40 (2023), eadi1480.

[15] Yu Feng, Nathan Goulding-Hotta, Asif Khan, Hans Reyserhove, and Yuhao Zhu. 2022. Real-time gaze tracking with event-driven eye segmentation. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 399–408.

[16] Lee Friedman, Mark S Nixon, and Oleg V Komogortsev. 2017. Method to assess the temporal persistence of potential biometric features: Application to oculomotor, gait, face and brain structure databases. *PloS one* 12, 6 (2017), e0178501.

[17] Wolfgang Fuhl, Thomas Kübler, Katrin Sippel, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2015. Excuse: Robust pupil detection in real-world scenarios. In *Computer Analysis of Images and Patterns: 16th International Conference, CAIP 2015, Valletta, Malta, September 2-4, 2015 Proceedings, Part I 16*. Springer, 39–51.

[18] Wolfgang Fuhl, Thiago C Santini, Thomas Kübler, and Enkelejda Kasneci. 2016. Else: Ellipse selection for robust pupil detection in real-world environments. In *Proceedings of the ninth biennial ACM symposium on eye tracking research & applications*. 123–130.

[19] Wolfgang Fuhl, Marc Tonsen, Andreas Bulling, and Enkelejda Kasneci. 2016. Pupil detection for head-mounted eye tracking in the wild: an evaluation of the state of the art. *Machine Vision and Applications* 27 (2016), 1275–1288.

[20] Anjith George and Aurobinda Routray. 2016. A score level fusion method for eye movement biometrics. *Pattern Recognition Letters* 82 (2016), 207–215.

[21] Jesse Grootjen, Alexandra Sipatchin, Siegfried Wahl, Tonja-Katrin Machulla, Lewis Chuang, and Thomas Kosch. 2023. Assessing Eye Tracking for Continuous Central Field Loss Monitoring. In *Proceedings of the 22nd International Conference on Mobile and Ubiquitous Multimedia*. 54–64.

[22] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. 2018. Densely Connected Convolutional Networks. arXiv:1608.06993 [cs.CV]

[23] Shaohua Jia, Amanda Seccia, Pasha Antonenko, Richard Lamb, Andreas Keil, Matthew Schneps, Marc Pomplun, et al. 2018. Biometric recognition through eye movements using a recurrent neural network. In *2018 IEEE International Conference on Big Knowledge (ICBK)*. IEEE, 57–64.

[24] Paweł Kasprowski, Oleg V Komogortsev, and Alex Karpov. 2012. First eye movement verification and identification competition at BTAS 2012. In *2012 IEEE fifth international conference on biometrics: theory, applications and systems (BTAS)*. IEEE, 195–202.

[25] Pawel Kasprowski and Józef Ober. 2004. Eye movements in biometrics. In *International Workshop on Biometric Authentication*. Springer, 248–258.

[26] Moritz Kassner, William Patera, and Andreas Bulling. 2014. Pupil: an open source platform for pervasive eye tracking and mobile gaze-based interaction. In *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing: Adjunct publication*. 1151–1160.

[27] Christina Katsini, Yasmeen Abdrabou, George E Raptis, Mohamed Khamis, and Florian Alt. 2020. The role of eye gaze in security and privacy applications: Survey and future HCI research directions. In *Proceedings of the 2020 CHI conference on human factors in computing systems*. 1–21.

[28] Mohamed Khamis, Ludwig Trotter, Ville Mäkelä, Emanuel von Zezschwitz, Jens Le, Andreas Bulling, and Florian Alt. 2018. Cueauth: Comparing touch, mid-air gestures, and gaze for cue-based authentication on situated displays. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 4 (2018), 1–22.

[29] Joohwan Kim, Michael Stengel, Alexander Majercik, Shalini De Mello, David Dunn, Samuli Laine, Morgan McGuire, and David Luebke. 2019. Nvgaze: An anatomically-informed dataset for low-latency, near-eye gaze estimation. In *Proceedings of the 2019 CHI conference on human factors in computing systems*. 1–12.

[30] Oleg V Komogortsev, Sampath Jayarathna, Cecilia R Aragon, and Mechehoul Mahmoud. 2010. Biometric identification via an oculomotor plant mathematical model. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*. 57–60.

[31] Rakshit S Kothari, Aayush K Chaudhary, Reynold J Bailey, Jeff B Pelz, and Gabriel J Diaz. 2021. Ellseg: An ellipse segmentation framework for robust gaze tracking. *IEEE Transactions on Visualization and Computer Graphics* 27, 5 (2021), 2757–2767.

[32] Xavier Lagorce, Garrick Orchard, Francesco Galluppi, Bertram E Shi, and Ryad B Benosman. 2016. Hots: a hierarchy of event-based time-surfaces for pattern recognition. *IEEE transactions on pattern analysis and machine intelligence* 39, 7 (2016), 1346–1359.

[33] Michael F Land and Peter McLeod. 2000. From eye movements to actions: how batsmen hit the ball. *Nature neuroscience* 3, 12 (2000), 1340–1345.

[34] Dong Yun Lee, Yunmi Shin, Rae Woong Park, Sun-Mi Cho, Sora Han, Changsoon Yoon, Jaheui Choo, Joo Min Shim, Kahee Kim, Sang-Won Jeon, et al. 2023. Use of eye tracking to improve the identification of attention-deficit/hyperactivity disorder in children. *Scientific Reports* 13, 1 (2023), 14469.

[35] Gregor Lenz, Sio-Hoi Ieng, and Ryad Benosman. 2020. Event-based face detection and tracking using the dynamics of eye blinks. *Frontiers in Neuroscience* 14 (2020), 495936.

[36] Jiading Li, Zhiyu Zhu, Jinhui Hou, Junhui Hou, and Jinjian Wu. 2024. Denoising Distillation Makes Event-Frame Transformers as Accurate Gaze Trackers. *arXiv preprint arXiv:2404.00548* (2024).

[37] Nealson Li, Ashwin Bhat, and Arijit Raychowdhury. 2023. E-track: Eye tracking with event camera for extended reality (xr) applications. In *2023 IEEE 5th International Conference on Artificial Intelligence Circuits and Systems (AICAS)*. IEEE, 1–5.

[38] Nealson Li, Muya Chang, and Arijit Raychowdhury. 2024. E-Gaze: Gaze Estimation with Event Camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2024).

[39] Patrick Lichtsteiner, Christoph Posch, and Tobi Delbruck. 2008. A 128 × 128 120 dB 15μs latency asynchronous temporal contrast vision sensor. *IEEE journal of solid-state circuits* 43, 2 (2008), 566–576.

[40] Chi-Wei Lien and Sudip Vhaduri. 2023. Challenges and opportunities of biometric user authentication in the age of iot: A survey. *Comput. Surveys* 56, 1 (2023), 1–37.

[41] Dillon Lohr and Oleg V Komogortsev. 2022. Eye know you too: Toward viable end-to-end eye movement biometrics for user authentication. *IEEE Transactions on Information Forensics and Security* 17 (2022), 3151–3164.

[42] Silvia Makowski, Paul Prasse, David R Reich, Daniel Krakowczyk, Lena A Jäger, and Tobias Scheffer. 2021. DeepEyedentificationLive: Oculomotoric biometric identification and presentation-attack detection using deep neural networks. *IEEE Transactions on Biometrics, Behavior, and Identity Science* 3, 4 (2021), 506–518.

[43] David L Mann, Hiroki Nakamoto, Nadine Logt, Lieke Sikkink, and Eli Brenner. 2019. Predictive eye movements when hitting a bouncing ball. *Journal of vision* 19, 14 (2019), 28–28.

[44] Urbano Miguel Nunes, Ryad Benosman, and Sio-Hoi Ieng. 2023. Adaptive Global Decay Process for Event Cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9771–9780.

[45] OpenCV. 2024. OpenCV Canny Edge Detection. *OpenCV* (2024). https://docs.opencv.org/4.x/dd/d1a/group__imgproc__feature.html#ga04723e007ed888ddf11d9ba04e2232de

[46] OpenCV. 2024. OpenCV Hough Circle Detection. *OpenCV* (2024). https://docs.opencv.org/4.x/dd/d1a/group__imgproc__feature.html#ga47849c3be0d0406ad3ca45db65a25d2d

[47] Cian Ryan, Brian O'Sullivan, Amr Elrasad, Aisling Cahill, Joe Lemley, Paul Kielty, Christoph Posch, and Etienne Perot. 2021. Real-time face & eye tracking and blink detection using event cameras. *Neural Networks* 141 (2021), 87–97.

[48] Timo Stoffregen, Hossein Daraei, Clare Robinson, and Alexander Fix. 2022. Event-based kilohertz eye tracking using coded differential lighting. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2515–2523.

[49] Marc Tonsen, Julian Steil, Yusuke Sugano, and Andreas Bulling. 2017. Invisibleeye: Mobile eye tracking using multiple low-resolution cameras and learning-based gaze estimation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 1–21.

[50] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008).

[51] Kailas Vodrahalli, Maciej Filipkowski, Tiffany Chen, James Zou, and Yaping Joyce Liao. 2022. Predicting Visuo-Motor Diseases From Eye Tracking Data. *Biocomputing 2022* (2022), 242–253. https://doi.org/10.1142/9789811250477_0023

[52] Zuowen Wang, Chang Gao, Zongwei Wu, Marcos V Conde, Radu Timofte, Shih-Chii Liu, Qinyu Chen, Zheng-jun Zha, Wei Zhai, Han Han, et al. 2024. Event-Based Eye Tracking. AIS 2024 Challenge Survey. *arXiv preprint arXiv:2404.11770* (2024).

[53] Haiwei Zhang, Jiqing Zhang, Bo Dong, Pieter Peers, Wenwei Wu, Xiaopeng Wei, Felix Heide, and Xin Yang. 2023. In the blink of an eye: Event-based emotion recognition. In *ACM SIGGRAPH 2023 Conference Proceedings*. 1–11.

[54] Tongyu Zhang, Yiran Shen, Guangrong Zhao, Lin Wang, Xiaoming Chen, Lu Bai, and Yuanfeng Zhou. 2024. Swift-Eye: Towards Anti-blink Pupil Tracking for Precise and Robust High-Frequency Near-Eye Movement Analysis with Event Cameras. *IEEE Transactions on Visualization and Computer Graphics* (2024).

[55] Yongtuo Zhang, Wen Hu, Weitao Xu, Chun Tung Chou, and Jiankun Hu. 2018. Continuous authentication using eye movement response of implicit visual stimuli. *proceedings of the acm on interactive, mobile, wearable and ubiquitous technologies* 1, 4 (2018), 1–22.

[56] Youming Zhang, Jyrki Rasku, and Martti Juhola. 2012. Biometric verification of subjects using saccade eye movements. *International Journal of Biometrics* 4, 4 (2012), 317–337.

[57] Yisa Zhang, Yuchen Zhao, Hengyi Lv, Yang Feng, Hailong Liu, and Chengshan Han. 2022. Adaptive Slicing Method of the Spatiotemporal Event Stream Obtained from a Dynamic Vision Sensor. *Sensors* 22, 7 (2022), 2614.

[58] Guangrong Zhao, Yurun Yang, Jingwei Liu, Ning Chen, Yiran Shen, Hongkai Wen, and Guohao Lan. 2024. Ev-eye: Rethinking high-frequency eye tracking through the lenses of event cameras. *Advances in Neural Information Processing Systems* 36 (2024).