DeepEyedentificationLive: Oculomotoric Biometric Identification and Presentation-Attack Detection using Deep Neural Networks

Silvia Makowski, Paul Prasse, David R. Reich, Daniel Krakowczyk, Lena A. Jäger, Tobias Scheffer

Abstract—We study involuntary micro-movements of both eyes, in addition to saccadic macro-movements, as biometric characteristic. We develop a deep convolutional neural network that processes binocular eye-tracking signals and verifies the viewer's identity. In order to detect presentation attacks, we develop a model in which the movements are a response to a controlled stimulus. The model detects replay attacks by processing both the controlled but randomized stimulus and the ocular response to this stimulus. We acquire eye movement data from 150 participants, with 4 sessions per participant and conduct experiments on this new and legacy data sets with varying tracker precision and sampling rate. We observe that the model detects replay attacks reliably. For identification and identity verification, the model attains substantially lower error rates than prior work. We explore the relationships between training population size, training data volume, types of visual stimuli, number of training and enrollment sessions, interval between enrollment and probe sessions on one hand and the model performance on the other hand.

Index Terms—biometrics, eye tracking, deep learning, presentation attack, data quality

1 Introduction

No single biometric characteristic that is known today is by itself sufficiently reliable for all biometric applications, unique, collectible, convenient, and universally available. For instance, while identification based on fingerprint and iris tend to be more accurate than facial recognition, a good-quality fingerprint cannot be obtained for approximately 2-4% of the population due to degradation of the fingerprints from manual labor or hand-related disabilities, while long eyelashes, small eye apertures, cosmetic contact lenses, and conditions including glaucoma and cataract prevent the collection of good-quality images of the iris for an estimated 7% of the population [1]. It is therefore desirable to expand the space of biometric characteristics that can be used by themselves or as part of multimodal biometric identification systems. National population registers can serve as an illustrating example of an application in which multiple modalities are necessary for a biometric system to meet required false-acceptance, false-rejection, and failure-to-enroll rates across a large and diverse population.

At the same time, no universally reliable method for detection of presentation attacks exists, due to both the adversarial nature of the problem and the unbounded space of possible presentation-attack instruments. Especially artifact-detection approaches are vulnerable to the development of new and unforseen presentation attack instruments. Challenge-response approaches can determine whether a presentation exhibits liveness properties. However, if the response requires a voluntary user action, the detection of presentation attacks is in conflict with a convenient user experience. If, in addition, the expected response can be derived easily from the challenge, the presentation attack can incorporate an automated or manually controlled response to an observed challenge. As an example application that calls for high resilience against presentation attacks with unforseen attack instruments, consider physical access control to high-security facilities.

It has long been known that the way we move our eyes in response to a given stimulus is highly individual [2] and more recent psychological research has shown that these individual characteristics are reliable over time [3]. Hence, it has been proposed to use eye movements as a behavioral biometric characteristic [4], [5].

Human eye movements alternate between *fixations* of around 250 ms during which the eye gaze is maintained on a location from which visual input is obtained and *saccades* of around 50 ms which are fast relocation movements that can reach up to $500°/s$ and during which visual intake is supressed. Moreover, three types of involuntary micro-movements always occur during fixations which, among other functions, prevent visual fading of the fixated image. *Drift* movements are very slow movements of around 0.1-0.4$°/s$ away from the center of a fixation which are superimposed by high-frequency, low-amplitude tremor of around 40-100 Hz whose velocity can reach up to $0.3°/s$. *Microsaccades* are occasional small saccades that can reach velocities of up to $120°/s$ and, among other functions, bring back the eye gaze to the intended center of a fixation after a drift movement has occurred [6], [7], [8], [9], [10].

Prior work on biometric identification using eye movements extracts fixations and saccades from the eye tracking signal and measures the values of engineered explicit fea-

- *Silvia Makowski, Paul Prasse, David R. Reich, Daniel Krakowczyk, Lena A. Jäger and Tobias Scheffer are with the Department of Computer Science, University of Potsdam, Germany*
E-Mail: {silvia.makowski, prasse, david.reich, daniel.krakowczyk, lena.jaeger, tobias.scheffer}@uni-potsdam.de
- *Lena A. Jäger is with the Department of Computational Linguistics, University of Zurich, Switzerland*
E-Mail: jaeger@cl.uzh.ch

Manuscript received December 30, 2020; revised May 10, 2021.

tures, such as fixation durations and saccadic amplitudes or velocities. Any information contained in the fixational micro-movements is discarded. Since saccades and fixations occur at a low frequency, a critical limitation of these approaches is that long eye gaze sequences of more than one minute [11] need to be observed before the system can reliably identify a user.

The additional information contained in the high-frequency micro-movements bear the potential of considerably speeding up the identification. Recently, a neural network has been studied that processes a raw monocular eye tracking signal measured during reading [12]. This approach does not rely on any prior detection of specific types of macro- or micro-movements. In order to detect replay attacks, we develop a model in which the eye movements are the ocular response to a challenge in the form of a controlled stimulus. In this setting, however, the identification task becomes more challenging as fixation durations and saccade amplitudes are largely determined by the stimulus, and their distributional properties are less likely to vary across individuals.

This paper reports and extends contributions of the prior conference manuscript of Makowski *et al.* [13]. We develop a deep convolutional neural network (CNN) that (a) processes binocular eye tracking signals. In addition to the eye-movement signals, the network processes the relative positions of the stimuli, enabling it to (b) detect replay attacks. We (c) perform a detailed comparison of the convolutional neural network to prior art on a wide range of data sets that are recorded with eye-tracking devices that differ in their precision and sampling rate. Individual characteristics of eye movements correlate stronger within a session than across multiple sessions [3]. Therefore, we (d) experimentally study a setting in which enrollment and application data are collected on different days.

In addition to the content of the conference presentation [13], we study how (e) the size of training population, (f) the data volume per user for training and enrollment, (g) the type of stimuli and (h) the time intervals between sessions affect verification performance.

The remainder of this paper is structured as follows. Section 2 reviews existing work on biometric identification and presentation-attack detection using oculomotoric measurements. Section 3 states the problem settings and Section 4 presents the *DeepEyedentificationLive* network and system. Section 5 gives an overview of the data sets used for evaluation. The experimental settings and results are presented in Section 6. Section 7 concludes.

2 Related Work

Traditionally, oculomotoric biometric identification and verification rely on eye-tracking data that is preprocessed into saccades and fixations. Spawned by the seminal works of Kasprowski and Ober [4] and Bednarik *et al.* [5], and fueled by competitions in the following decade [14], [15], these methods can be subsumed into three categories: aggregational [16], [17], [18], statistical [19], [20], [21], [22] and generative methods. Suitable generative methods include Markov [23], [24] and graphical models [11], [25], [26].

Recent work uses deep learning, either processing extracted features [27], [28] or learning an embedding end-to-end from the raw eye tracking signal [12], [29], [30]. Out of all prior approaches, *DeepEyedentification* is the only model that is able to utilize micro-movements contained in the raw signal. This work is further extended by Makowski *et al.* [13] to handle binocular data and detect presentation attacks. Prasse *et al.* [31] study the model's susceptibility to decreased tracking resolution.

The spectrum of visual stimuli that have been studied ranges from a static cross [5], images [32], faces [19], [33], [34], text [11], [12], [16], [26], video [35] and various implementations of jumping points [4], [17], [36], [37], [38]. Only a handful of studies evaluate their models on stimuli that have not also been shown to the respective user during enrollment [11], [12], [13], [16], [26], [31], [35]. Repetitions in stimulus sequences, unfortunately, enable an attacker to record the oculomotoric response to the known stimulus, and to perform a replay attack by presenting the recording to the system. Biometric systems that do not challenge the user with a randomized stimulus are left with the challenge of detecting imperfections in the data that are caused by specific presentation-attack instruments [39], [40].

Prior work on presentation-attack detection in the context of gaze-based identification [39], [40] assumes that an attacker generates artificial eye movements, based on a model of a target individual's gaze patterns. The proposed methods use a classifier to discriminate *bona fide* from generated eye movements using the same engineered features that are used for identification. This approach relies on imperfections of the gaze model and cannot detect an attacker who replays actual eye movements that were recorded from the target individual. Approaches to presentation-attack detection that detect artifacts of specific presentation-attack instruments have been studied widely for other biometrics; for instance, for iris recognition. Work of Raja *et al.* [41] exploits phase information which is indicative of presentations on smartphone or tablet screens.

3 Problem Setting

We will study the problems of oculomotoric biometric identification, *identity verification*, and *presentation-attack detection*. The input to each system is given as a sequence of eye gaze yaw and pitch angles of the left and right eye over an observation period.

In a biometric verification scenario, each user first enrolls with one or more enrollment sequences. At application time, these enrollment sequences are compared to a probe sequence by a suitable similarity metric. If a similarity threshold is exceeded for an enrollment sequence, the presumed identity of the user is verified; or otherwise, the user is exposed as an impostor. The algorithm performance can be characterized by a *false-match rate* (FMR, fraction of impostors among all accepted users) and a *false non-match rate* (FNMR, fraction of falsely rejected users among all rejected users). By changing the decision threshold, one can observe a *detection error tradeoff curve (DET curve)*. The *equal error rate* (EER) is the point on this curve for which FMR equals FNMR.

In the *identification* setting, the gaze sequence that is observed at application time is compared to one or more *enrollment sequences* of *multiple* enrolled users. In case of a positive identification, the outcome is the matched identity; otherwise, the user is classified as *impostor*. The DET curve characterizes the trade-offs between *false-positive identification-error rate* (FPIR) and *false-negative identification-error rate* (FNIR) for enrolled users; here, false positive identifications can be impostors or enrolled users who are mistaken for different enrolled users.

For comparison with previously published results, we also conduct experiments in a multi-class classification setting. Here, all users are enrolled and, at application time, a new user has to be identified as one of the enrolled users. Impostors do not exist in this legacy multi-class setting. Here, we measure the identification accuracy.

In some approaches, the similarity metric is defined as a metric on a vector of engineered features which are extracted from the gaze sequence [16], [21]. In our approach, the similarity metric is the the cosine similarity between *neural embeddings* of gaze sequences. This embedding is trained on a separate set of training users which is disjoint from the users that are encountered at application time. The neural network is trained such that the embedding is similar for all gaze sequences of a particular user but different for gaze sequences of distinct users.

The *presentation-attack detection* problem is to detect whether the observed gaze sequence is presented with the goal of interfering with the biometric system. We study the case of a complete artificial replay attack by an adversary who can observe both the size of the display on which the stimulus is presented and the duration for which each stimulus is displayed. The adversary does not, however, have advance information about the randomized positions of the five dots; therefore, they are limited to replaying a gaze sequence for a random stimulus with the same display size and display duration. We measure the DET curve between the *attack-presentation classification-error rate* (APCER)—the proportion of attack presentations incorrectly classified as *bona fide* presentations—and the *bona-fide presentation-classification error rate* (BPCER)—the proportion of *bona fide* presentations that are misclassified as attack.

As an example presentation-attack instrument for this type of attack, an attacker may record eye movements of the target person unnoticed by means of a remote eye tracker. The attacker may then be able to perform a presentation attack by injecting the recorded eye-gaze signal into an eye-tracking device.

Note that presentation attacks by lifeless humans are not possible due to the lack of eye movements, and that humans cannot be altered to exhibit another person's patterns of ocular micromovements. In a nonconformant presentation, the gaze patterns would be absent while a conformant *zero-effort* presentation attack by a human impostor would require a false match to be successful.

4 System and Network Architecture

This section derives the *DeepEyedentificationLive*[1] system and the neural network that performs binocular oculomotoric biometric identification and liveness detection. An eye scanner records the user's eye gaze while a display (see Figure 1) shows a sequence of dots at random locations. The gaze sequence of absolute yaw x and pitch gaze angles y of the left l and right eye r recorded with sampling frequency ρ in Hz is transformed into sequences of yaw δ_i^x and pitch δ_i^y gaze velocities in $°/s$ where $\delta_i^x = \frac{\rho}{2}(x_{i+1} - x_{i-1})$ and $\delta_i^y = \frac{\rho}{2}(y_{i+1} - y_{i-1})$. These four velocity sequences constitute four of the input channels into the network: $\langle \delta_1^{x,l}, \ldots, \delta_n^{x,l} \rangle$ is the sequence of yaw angular velocities of the left eye; $\langle \delta_1^{y,l}, \ldots, \delta_n^{y,l} \rangle$ is the sequence of pitch angular velocities; $\langle \delta_1^{x,r}, \ldots, \delta_n^{x,r} \rangle$ and $\langle \delta_1^{y,r}, \ldots, \delta_n^{y,r} \rangle$ are the corresponding yaw and pitch velocities of the right eye.

Since the velocity of saccadic and fixational eye movements occur at vastly different scales, global normalization would squash the slow fixational drift and tremor to near-zero and as a consequence much of the information in the eye tracking signal would be lost. The solution to this challenge is a model architecture with two separate subnets that process these gaze velocities with different scaling: a fast subnet processes the velocities in a resolution that is suitable for high-velocity movements, and a slow subnet in a resolution suitable for low velocities. The two subnets have the same type of layers except for a transformation layer that transforms the input to resolve the fast and the slow movements, respectively.

For the fast subnet, absolute velocities below a minimal velocity ν_{min} are truncated and z-score normalization is applied (see Equation 1).

$$t_f(\delta_i^x, \delta_i^y) = \begin{cases} z(0) & \text{if } \sqrt{\delta_i^{x\,2} + \delta_i^{y\,2}} < \nu_{min} \\ (z(\delta_i^x), z(\delta_i^y)) & \text{otherwise} \end{cases} \tag{1}$$

For the slow subnet, a sigmoidal function is applied such that the slow fixational movements (drift and tremor) are stretched to within the interval between -0.5 and $+0.5$ whereas the fast microsaccades and saccades are squashed to values between -0.5 and -1 or $+0.5$ and $+1$ (see Equation 2). The values for the threshold ν_{min} of Equation 1 and the scaling factor c of Equation 2 are determined by hyperparameter tuning in a range of psychologically plausible values (see Section 6.2).

$$t_s(\delta_i^x,\ \delta_i^y) = (\tanh(c\delta_i^x),\ \tanh(c\delta_i^y)) \tag{2}$$

The original input velocities are also fed into a subtraction layer that computes the yaw $\langle \delta_1^{x,r} - \delta_1^{x,l}, \ldots, \delta_n^{x,r} - \delta_n^{x,l} \rangle$ and pitch velocity differences between the two eyes $\langle \delta_1^{y,r} - \delta_1^{y,l}, \ldots, \delta_n^{y,r} - \delta_n^{y,l} \rangle$. These two channels are then stacked with each of the outputs of the transformation layers.

The network additionally processes the positions of the visual stimuli to which the gaze sequence is the oculomotoric response. In our experiments on presentation-attack detection, dots are displayed at five random positions in each trial. The stimuli are represented as offsets in x and y direction to the previous stimulus position: $\langle \delta_1^{x,s}, \ldots, \delta_n^{x,s} \rangle$ and $\langle \delta_1^{y,s}, \ldots, \delta_n^{y,s} \rangle$, where each δ_i^s is the offset in degrees between the stimulus displayed at times i and $i-1$; in most time steps, the stimulus position does not change. Note that when a stimulus is displayed from time t to time t' and the user's eye gaze moves from the previous stimulus to exactly

[1] The code is accessible at https://osf.io/8es7z/.

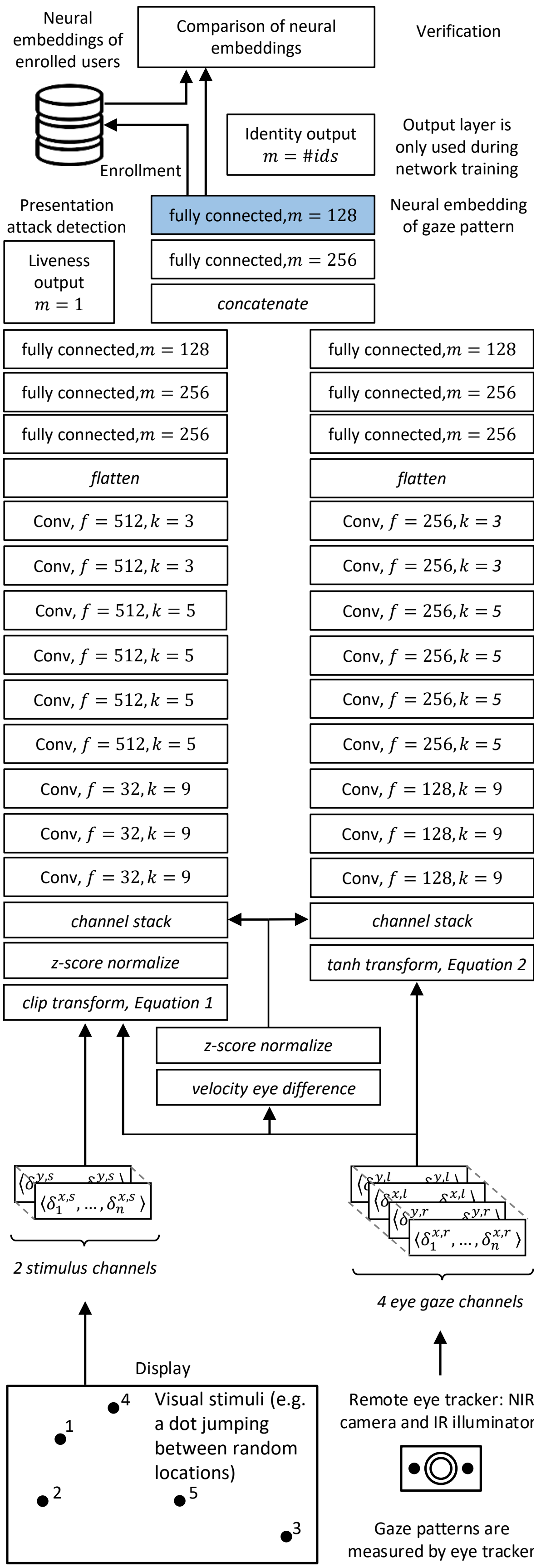

Neural embeddings of enrolled users
Comparison of neural embeddings
Verification
Identity output $m = \#ids$
Output layer is only used during network training
Enrollment
Presentation attack detection
fully connected, $m = 128$
Neural embedding of gaze pattern
Liveness output $m = 1$
fully connected, $m = 256$
concatenate
fully connected, $m = 128$
fully connected, $m = 256$
fully connected, $m = 256$
flatten
Conv, $f = 512, k = 3$
Conv, $f = 512, k = 3$
Conv, $f = 512, k = 5$
Conv, $f = 512, k = 5$
Conv, $f = 512, k = 5$
Conv, $f = 512, k = 5$
Conv, $f = 32, k = 9$
Conv, $f = 32, k = 9$
Conv, $f = 32, k = 9$
channel stack
z-score normalize
clip transform, Equation 1
fully connected, $m = 128$
fully connected, $m = 256$
fully connected, $m = 256$
flatten
Conv, $f = 256, k = 3$
Conv, $f = 256, k = 3$
Conv, $f = 256, k = 5$
Conv, $f = 256, k = 5$
Conv, $f = 256, k = 5$
Conv, $f = 256, k = 5$
Conv, $f = 128, k = 9$
Conv, $f = 128, k = 9$
Conv, $f = 128, k = 9$
channel stack
tanh transform, Equation 2
z-score normalize
velocity eye difference
$\langle \delta_1^{y,s}, \ldots, \delta_n^{y,s} \rangle$
$\langle \delta_1^{x,s}, \ldots, \delta_n^{x,s} \rangle$
2 stimulus channels
$\langle \delta_1^{y,l}, \ldots, \delta_n^{y,l} \rangle$
$\langle \delta_1^{x,l}, \ldots, \delta_n^{x,l} \rangle$
$\langle \delta_1^{y,r}, \ldots, \delta_n^{y,r} \rangle$
$\langle \delta_1^{x,r}, \ldots, \delta_n^{x,r} \rangle$
4 eye gaze channels
Display
Visual stimuli (e.g. a dot jumping between random locations)
1
2
3
4
5
Remote eye tracker: NIR camera and IR illuminator
Gaze patterns are measured by eye tracker

Fig. 1. *DeepEyedentificationLive* architecture. The left strand depicts the fast subnet, the right one the slow subnet.

the new stimulus within this interval, then, both for the left and right eye, it holds that

$$\sum_{i=t}^{t'} \delta_i^x - \sum_{i=t}^{t'} \delta_i^{x,s} = \sum_{i=t}^{t'} \delta_i^y - \sum_{i=t}^{t'} \delta_t^{y,s} = 0. \qquad (3)$$

The network processes the input in one-dimensional convolutions over time. This is in analogy to other CNN architectures that process time-sequential data—for instance, speech recognition—and in contrast to image-processing CNNs that use two-dimensional convolutions. The six input sequences are processed as six channels, in analogy to the red, green and blue channels by image processing CNNs. Convolutional neural networks require an input sequence of fixed width. Therefore, the network processes the gaze sequences in windows of 1,000 time steps, which corresponds to one second of input data. All input sequences are therefore split into subsequences of 1,000 ms, and the results—the similarity metric for identity verification, and the activation of the output unit for liveness detection—are averaged across all subsequences of each gaze sequence.

Parameter f in Figure 1 shows the number of filters, k specifies the kernel size of convolutions. Parameter m characterizes the number of units of fully connected layers. The values of these parameters are determined by hyperparameter tuning (see Section 6.2). Convolutional and fully connected layers are all batch normalized and followed by a ReLu activation function. Each convolutional layer is followed by an average pooling layer with pooling size 2 and stride of 1. The network has a sigmoid layer for the liveness ouput.

For the purpose of training the network, a softmax output layer with one unit for each training user is added. The network is then trained on gaze sequences of training users by minimizing the cross-entropy loss. For half of the training sequences, the correct stimulus is presented to the network as input and the target liveness output is +1. In the remaining cases, a random stimulus with the same display size and display duration is chosen and the target liveness output value is -1. After training, the identification softmax layer is discarded and the embedding layer provides the neural feature embedding of each gaze sequence. Because the network has learned to identify the training users based on the activations of the embedding units, the embedding distills signals that vary across individuals and are indicative of the viewer's identity.

The similarity metric between enrollment and application sequence is given by the cosine similarity, averaged over all input windows of 1,000 ms. The similarity value between an application sequence and a user is the maximum similarity over that user's enrollment sequences. During the enrollment process, the embedding of one or several gaze sequences are determined and stored in a database.

TABLE 1
Data sets with information on the eye tracking device in use, its nominal precision in degrees of visual angle and sampling frequency in Hz, the tracked eyes, use of a chin rest, total number of subjects together with the number of sessions and trials per session, as well as the average trial duration and overall recording time per subject in seconds including their standard deviations

Data set	Device	Prec.	Freq.	Eye(s)	Chin rest	# subj.	# sess.	# trials	Trial dur.	Rec./subj.
JuDo1000 [42]	EyeLink Portable Duo	0.01	1000	both	yes	150	4	108	3 ± 1.6	1152 ± 0
CEM-I (SIM) [43]	Tobii TX300	0.09	300	both	yes	22	1	8	64 ± 47	465 ± 288
CEM-I (COM) [43]	Tobii TX300	0.09	300	both	yes	22	1	2	207 ± 17	386 ± 75
CEM-I (COG) [43]	Tobii TX300	0.09	300	both	yes	22	1	2	99 ± 51	180 ± 95
CEM-I (TEX) [43]	Tobii TX300	0.09	300	both	yes	22	1	4	40 ± 20	148 ± 71
CEM-II (TEX) [43]	EyeLink 1000	0.01	1000	right	yes	32	1	4	51 ± 10	192 ± 38
CEM-III (SIM) [43]	PlayStation Eye	N/A	75	right	yes	173	1	2	89 ± 8	177 ± 18
CEM-III (COM) [43]	PlayStation Eye	N/A	75	right	yes	173	1	2	133 ± 16	264 ± 32
CEM-III (RAN) [43]	PlayStation Eye	N/A	75	right	yes	164	1	2	77 ± 7	154 ± 13
CEM-III (TEX) [43]	PlayStation Eye	N/A	75	right	yes	172	1	2	46 ± 5	91 ± 10
GazeBase (RAN) [44]	EyeLink 1000	0.01	1000	left	yes	322	1-9	2	101 ± 0.03	553 ± 511
GazeBase (TEX) [44]	EyeLink 1000	0.01	1000	left	yes	322	1-9	2	60 ± 0.04	329 ± 304
GazeBase (FXS) [44]	EyeLink 1000	0.01	1000	left	yes	322	1-9	2	15 ± 0.01	82 ± 76
GazeBase (VD1) [44]	EyeLink 1000	0.01	1000	left	yes	322	1-9	2	58 ± 1.50	320 ± 296
GazeBase (VD2) [44]	EyeLink 1000	0.01	1000	left	yes	322	1-9	2	58 ± 1.5	320 ± 296
GazeBase (BLG) [44]	EyeLink 1000	0.01	1000	left	yes	322	1-9	2	37 ± 20	203 ± 178
GazeBase (HSS) [44]	EyeLink 1000	0.01	1000	left	yes	322	1-9	2	101 ± 0.08	554 ± 512

5 Data Sets

We collect the *JuDo1000 data set*[2][3] of binocular eye movement data (horizontal and vertical gaze coordinates) from 150 participants (18 to 46 years old, mean age 24 years), each of whom participate in four experimental sessions with a lag of at least one week between any two sessions. Eye movements are recorded using an Eyelink Portable Duo eye tracker (tripod mounted camera) at a sampling frequency of 1,000 Hz. Participants are seated in front of a 38×30 cm (1280×1024 px) computer monitor with a viewing distance of 68 cm at a height adjustable table with their head stabilized by a chin- and forehead rest.

In each session, participants are presented with a total of 108 experimental trials in which a black dot with a diameter of 0.59 cm (20 px) appears consecutively at 5 random grid positions on a white background. The duration for which each dot is displayed is varied between 250, 500 and 1000 ms with a fixed value within each trial; the size of the screen area in which the dots appear is varied between 7.6×14.0 cm, 11.4×17.0 cm, and 19.0×23.0 cm around the center of the monitor with a fixed area within each trial. The largest screen area corresponds to ±7.9 (and ±6.3) degrees of visual angle horizontally (and vertically). The distance between two consecutive dot positions ranges from 1.6 to 20.3 degrees of visual angle. The combination of display duration and areas results in nine *trial configurations*. Figure 2 shows example eye-movement traces of the left and right eye for different display duration and areas.

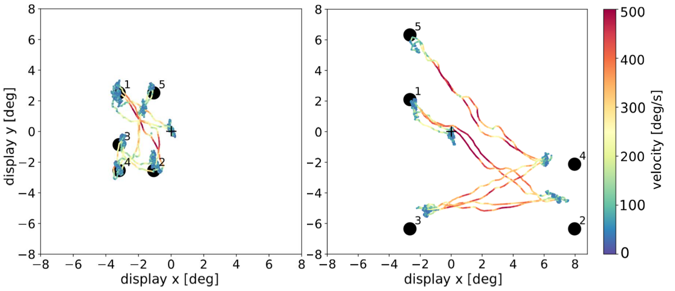


Fig. 2. Exemplary eye traces for trial configurations 500 ms stimulus display duration and small grid (*left*), and 250 ms display duration on big grid (*right*). The cross is displayed before the onset of the trial.

The *GazeBase* data set [44] contains gaze recordings of 322 college-aged participants who are recorded monocularly with an EyeLink 1000 eye tracker at a sampling frequency of 1,000 Hz. Participants are placed at a viewing distance of 55 cm to a monitor with a display size of 47.4×29.7 cm (1680×1050 px), while their heads are stabilized by a chin and forehead rest. Participants are recorded over two identical, consecutive recording sessions with a break of at most 5 minutes and in which they perform seven tasks: A random saccade (RAN), a reading task (TEX), two video viewing tasks (VD1 and VD2), a fixation task (FXS), a horizontal saccade task (HSS) and a gaze-driven game called Balura (BLG). This experiment is repeated over a time period of 37 months, resulting in 9 recording rounds. Particpants of subsequent round are recruited only from the pool of participants of the previous round.

The RAN task resembles our experiment in the *JuDo1000* data set. It requires participants to follow a target point, which appears at random screen positions with a display duration of 1000 ms. The target points appear in ±15 (and ±9) degrees of visual angle horizontally (and vertically) [44].

[2] The *JuDo1000 data set* is accessible at https://osf.io/5zpvk/.

[3] Participants have been informed about the purpose of the research and the procedure of collecting eye-tracking data and have given their informed consent. The study is exempt from approval by the ethic committee of the University of Potsdam because participants are of age and have given their informed consent and, furthermore, being subjected to eye tracking holds no potential of experiencing bodily or mental harm.

The distance between two consecutive dot positions ranges from 0.26 to 33.7 degrees of visual angle.

The *CEM-I* data set [43] contains binocular data of a single session for 22 different subjects and is recorded with the Tobii TX300 eye tracker. The vendor reports a spatial precision of 0.09° at the selected sampling frequency of 300 Hz. In each session, the subjects are presented four different types of distinct stimuli: eight simple patterns (SIM), two complex patterns (COM), two cognitive patterns (COG) and four textual patterns (TEX) [43].

The *CEM-II* data set [43] contains right-eye monocular data of a single session for 32 different subjects and is recorded with the EyeLink 1000 eye tracker. The vendor reports a spatial precision of 0.01° at the selected sampling frequency of 1,000 Hz. In each session, the subjects are presented the same textual patterns as in *CEM-I*.

The *CEM-III* data set [43] contains right-eye monocular data recorded with the PlayStation Eye eye tracker at 75 Hz. The spatial precision remains unspecified. The stimulus types include simple, complex and textual patterns from the *CEM-I* data set, as well as an additional random saccade task stimulus (RAN). Each stimulus is presented for two trials. The number of subjects for each stimulus type vary from 164 to 173.

6 Experiments

This section reports on the experimental results for biometric identification, identity verification and presentation-attack detection.

6.1 Hardware and Framework

For all experiments we use a server with 40-core Intel Xeon CPU E5-2640 processor, 128 GB memory and GeForce GTX TITAN X GPU. For training the network we use Keras, TensorFlow, and the Adam optimizer with a learning rate of 0.001 for the slow and fast subnets, and a learning rate of 0.0001 for the remaining layers. We use a batch size of 64. We use early stopping with a patience of 10 epochs. The loss for early stopping is measured on validation data that is removed from training and evaluation data (see Section 6.2).

6.2 Hyperparameter Tuning

We tune the hyperparameters with a random grid search in the parameter search space shown in Table 2. As validation data we use one hold-out trial from each configuration per session, which is removed from the training and testing data of the final model. We constrain kernel sizes and number of filters of the convolutional layers to be identical within layers 1-3, 4-7 and 8-9 of both subnets. Kernel sizes are furthermore constrained to be smaller or equal and filter sizes greater or equal to the previous layer block. Figure 1 shows the best parameter configuration.

The parameters for the respective input transformations of the fast and slow subnet—i.e., velocity threshold $\nu_{\min}$ and scaling factor c—are tuned in a range of psychologically plausible values on the *JuDo1000* data set. $\nu_{\min}$ is set to 40°/s and c to 0.02.

TABLE 2
Parameter space used for random grid search: kernel size k and number of filters f of all convolutional layers and number of units m of all fully connected layers. Table as in Makowski *et al.* [13].

Parameter	Search space
k	$\{3, 5, 7, 9\}$
f	$\{32, 64, 128, 256, 512\}$
m	$\{64, 128, 256, 512\}$

6.3 Identification and Identity Verification

For the *JuDo1000* and *GazeBase* data sets, we resample 10 times, in each iteration randomly selecting a population of users to train an embedding. Unless we specifically study smaller training populations, we train the embedding on 125 users for the *JuDo1000* data set and 100 users for *GazeBase*. In each resampling round, we select 25 test users who are disjoint from the training users. In the verification setting, each test user is enrolled in turn, and the remaining 24 users act as impostors. In the identification setting, 20 users are selected as enrolled users and 5 users act as impostors.

Enrollment and test data are also split across recording sessions. Unless we specifically control the number of enrollment sessions in an experiment, for the *JuDo1000* data set a user is enrolled using 9 trials with different trial configurations from 3 enrollment sessions; this results in a total of 24 seconds of enrollment data. At application time, embeddings are calculated from probe sequences of the disjoint fourth test session.

In the *GazeBase* data, the number of sessions per user varies, and the gaze patterns are vastly different across stimulus types. In each of 10 resampling rounds, we select 25 test users for whom at least 8 sessions are available; 100 distinct users with at least 2 sessions serve as training users. For each stimulus type, a user is enrolled in turn using 24 seconds of enrollment data for the same stimulus type drawn from sessions 1 through 4, the remaining 24 test users act as impostors. Data from sessions 5 through 8 are used as probe sequences.

Since there is no publicly available implementation of Lohr *et al.* [28] and Friedman *et al.* [22], we compare DeepEyedenticationLive to published results [22], [28] in the exact same experimental setting. We perform 4-fold cross validation over the 269 subjects of the GazeBase TEX data set. In each fold, three-quarters of the subjects are used for training while the remaining are left out for testing. At application time, round one of session one of the test users is used for enrollment and round one of session two serves as probe sequence.

To obtain comparable results on the *CEM* data sets, we use the same multi-class identification evaluation protocol as Holland and Komogortsev [16]: Each stimulus type is evaluated separately. For each stimulus type, we randomly resample 20 times from the data set, in each iteration using half of the subjects as training users and the other half as enrolled users. At application time, each trial from the enrolled users in turn serves as test instance and the remaining trials are used as enrollment data. A correct identification occurs when the similarity between a test trial and the enrollment data of the true user exceeds a threshold and is higher than the similarity to the enrollment data of any other enrolled user.

6.3.1 Reference Methods

As representatives for deep learning based methods we compare against the DeepEyedentification network, which differs from *DeepEyedentificationLive* in that it can only process monocular data and lacks presentation-attack detection; and Abdelwahab and Landwehr [29], who train a distributional sequence embedding on raw gaze sequences and pupil dilations. Additionally, we compare against two recent representatives for deep metric learning based approaches: Whereas Lohr et al. [28] learn an embedding from extracted features of scanpaths by optimizing the triplet loss, Abdelwahab and Landwehr [30] train the same distributional sequence embedding as presented in their prior study [29] end-to-end by optimizing the Wasserstein loss.

Additionally, we compare against several representatives for statistical methods. Statistical methods first preprocess the gaze recording into types of macro-movements and then extract features from these, such as the fixation durations and amplitude, velocity, or acceleration of saccadic movements. Two scanpaths are compared by computing the similarity of their feature distributions by applying statistical tests. We compare against Holland and Komogortsev [20], which is the first method of this kind and Rigas *et al.* [21]. Both were re-implemented by ourselves. As a more recent method, we compare to Friedman *et al.* [22] who preprocess the gaze recording into fixations, saccades and post-saccadic oscillations, then extract over 1000 features from these low-frequency macro-movements and finally perform a feature selection using the intra-class coefficient as a measure of test-retest reliability.

In order to offer a comparison to Lohr *et al.* and Friedman *et al.*, we are limited to report the performance of both presented in Lohr *et al.* [28] and to evaluate our model in the exact same evaluation protocol they use, because of unavailable source code and implementational details.

6.3.2 Comparison to Prior Art

This section compares the performance of *DeepEyedentificationLive* to the performance of known approaches to oculomotoric identification on the *JuDo1000*, *GazeBase* and the *CEM* data sets.

For the *JuDo1000* data set, Figure 4 and Table 4 show the identification performance of *DeepEyedentificationLive* and the baseline models for 20 enrolled users, averaged across all trial configurations. Here, the EER decreases from 0.1197 for one second of input data at test time to 0.0774 after 10 seconds. *DeepEyedentificationLive* obtains a lower FNIR than DeepEyedentification for every FPIR and every trial duration. The difference between the monocular *DeepEyedentification* and binocular *DeepEyedentificationLive* and are explained by the additional information that the binocular model receives as input. The performance gap between *DeepEyedentificationLive* on one hand and Holland and Komogortsev and Rigas *et al.* on the other hand is dramatic. It should be noted that both baselines use distributional properties of fixation durations and saccadic properties which here are largely dominated by the controlled stimuli. The performance gap to Abdelwahab and Landwehr [29] can be attributed to the fact that their architecture is not suitable for short input windows and high sampling rates. For the same model trained with the Wasserstein loss; i.e., Abdelwahab and Landwehr [30], we obtain a performance equal to random guessing. We therefore did not include it into our performance comparisons.

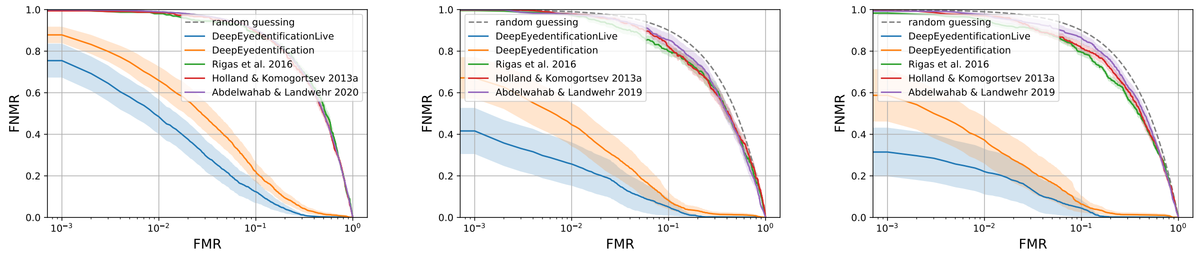
In the verification setting, shown averaged across all trial configurations in Figure 3 and Table 3, there is only one identity each imposter can be confused with. Here, the EER ranges from 0.091 for one second to 0.0301 for 10 seconds of input at test time. Again the performance gap to the baselines is dramatic.

Table 5—adapted from Prasse *et al.* [31]—shows the identification accuracy of the different methods for each of the *CEM* data sets. The rows labeled "ours" report results of our own implementation of the reference methods whereas the row labeled "Holland & Komogortsev" documents results published by Holland and Komogortsev [45]. The difference between these numbers might be due to differing preprocessing algorithms and threshold parameters. While we label micro-saccades as saccades, Holland and Komogortsev treat them as part of a fixation. In addition to the preprocessing and possible implementational details, the fusion metrics used to combine the different fixational and saccadic features differs between our implementation and the one used by Holland and Komogortsev for the evaluation of their model: Whereas they train a linear model to weight the different features, we apply the simple mean metrics as used by Rigas *et al.* [21] in their main evaluation of their model and the method of Holland and Komogortsev [20].

We can see in Table 5 that the *DeepEyedentificationLive* network outperforms the models of Rigas *et al.* [21] and Holland and Komogortsev [20] in most; differences are statistically significant for the COM and TEX stimuli on CEM-I (Tobii TX300 with 0.09° precision and 300 Hz) and all CEM-III tasks (PlayStation Eye with unspecified precision and 75 Hz). Only for the simple-patterns stimulus of CEM-I, our implementation of Holland and Komogortsev [20] significantly outperforms *DeepEyedentificationLive*.

Table 6 compares *DeepEyedentificationLive* to published results of Lohr *et al.* [28] and Friedman *et al.* [22] for stimuli of 60 seconds on the GazeBase TEX data set. We see that DeepEyedentificationLive outperforms Lohr *et al.* [28] and performs as well as Friedman *et al.* [22]. This method extracts features from low-frequency macro-movements and therefore it is not plausible that it could perform well on short sequences. By contrast, we see in Table 7 that with only 10 seconds of input, *DeepEyedentificationLive* reaches a similarly low error rate. We cannot run the method of Friedman *et al.* [22] in a setting with shorter stimuli because no implementation is available and some details of the method are not disclosed.

We conclude that the ongoing micro-scale eye movements comprise many individual characteristic patterns which are lost when only considering macro-scale eye movements that take place less frequently. Perhaps surprisingly, even trackers with low precision and sampling rate contain enough identifying information outside of saccade and fixation features to give *DeepEyedentificationLive* an advantage over statistical methods that use engineered features of fixations and saccades.

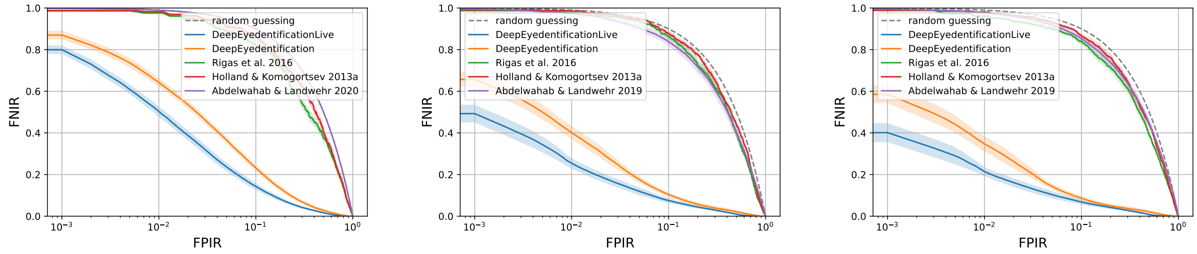


(a) 1 s of test recording, 1000 Hz (b) 5 s of test recording, 1000 Hz (c) 10 s of test recording, 1000 Hz

Fig. 3. Comparison of methods for oculomotoric identity verification on the *JuDo1000* data set. False-Non Matching Rate (FNMR) over False-Match Rate (FMR). Colored bands show the standard error. Figure adapted from Makowski *et al.* [13].

TABLE 3
Performance metrics ± standard errors on the *JuDo1000* data set for the verification setting. Values marked "*" are significantly worse ($p < 0.05$) than results for *DeepEyedentificationLive*.

Metric	DeepEyedentificationLive	DeepEyedentification	Rigas et al. 2016	H & K 2013a	A & L 2020
EER @ 1 s	**0.091 ± 0.0169**	0.1391 ± 0.0177	0.5187 ± 0.0152*	0.4996 ± 0.0126*	0.1515 ± 0.1018*
EER @ 5 s	**0.0397 ± 0.0129**	0.0713 ± 0.0169	0.4527 ± 0.0187*	0.4556 ± 0.0206*	0.2055 ± 0.0666*
EER @ 10 s	**0.0301 ± 0.0124**	0.0548 ± 0.0179	0.4535 ± 0.0092*	0.4642 ± 0.0165*	0.1721 ± 0.0673*
FNMR @ FMR 10^{-2} @ 1 s	**0.4843 ± 0.0838**	0.659 ± 0.0678	0.9809 ± 0.0066*	0.9849 ± 0.0048*	0.99 ± 0.0*
FNMR @ FMR 10^{-2} @ 5 s	**0.2567 ± 0.1043**	0.446 ± 0.1045	0.9731 ± 0.007*	0.9766 ± 0.008*	0.9764 ± 0.0113*
FNMR @ FMR 10^{-2} @ 10 s	**0.2201 ± 0.1096**	0.3721 ± 0.1131	0.9509 ± 0.0113*	0.9712 ± 0.0091*	0.9755 ± 0.0103*



(a) 1 s of test recording, 1000 Hz (b) 5 s of test recording, 1000 Hz (c) 10 s of test recording, 1000 Hz

Fig. 4. Performance of state-of-the-art methods for oculomotoric identification on the *JuDo1000* data set. . False-Negative Identification-Error Rate (FNIR) over False-Positive Identification-Error Rate (FPIR). Colored bands show the standard error. Figure adapted from Makowski *et al.* [13].

TABLE 4
Performance metrics ± standard errors on the *JuDo1000* data set for the identification setting. Values marked "*" are significantly worse ($p < 0.05$) than results for *DeepEyedentificationLive*.

Metric	DeepEyedentificationLive	DeepEyedentification	Rigas et al. 2016	H & K 2013a	A & L 2020
EER @ 1 s	**0.1197 ± 0.0073**	0.1549 ± 0.0069*	0.4314 ± 0.0139*	0.4577 ± 0.0152*	0.1493 ± 0.101*
EER @ 5 s	**0.0838 ± 0.0077**	0.1024 ± 0.0070	0.4522 ± 0.0112*	0.4706 ± 0.0078*	0.2018 ± 0.0642*
EER @ 10 s	**0.0774 ± 0.0079**	0.0917 ± 0.0071	0.4456 ± 0.0095*	0.4695 ± 0.0108*	0.1718 ± 0.0666*
FNIR @ FPIR 10^{-2} @ 1 s	**0.5033 ± 0.0250**	0.6429 ± 0.0216*	0.9757 ± 0.0054*	0.9823 ± 0.0076*	0.9899 ± 0.0001*
FNIR @ FPIR 10^{-2} @ 5 s	**0.2546 ± 0.0256**	0.4013 ± 0.0322*	0.976 ± 0.0057*	0.9851 ± 0.0029*	0.9724 ± 0.0068*
FNIR @ FPIR 10^{-2} @ 10 s	**0.2143 ± 0.0225**	0.3474 ± 0.0374*	0.961 ± 0.0061*	0.9802 ± 0.0048*	0.9763 ± 0.006*
FNIR @ FPIR 10^{-3} @ 1 s	**0.7997 ± 0.0206**	0.8702 ± 0.021*	0.987 ± 0.0047*	0.9878 ± 0.0076*	0.9989 ± 0.0001*
FNIR @ FPIR 10^{-3} @ 5 s	**0.4928 ± 0.0427**	0.6571 ± 0.0352	0.9875 ± 0.0044*	0.9911 ± 0.0027*	0.9943 ± 0.0017*
FNIR @ FPIR 10^{-3} @ 10 s	**0.4014 ± 0.0459**	0.5857 ± 0.0426	0.9884 ± 0.0042*	0.9896 ± 0.0037*	0.9948 ± 0.0018*

6.3.3 *Impact of the Size of the Training Population*

We conduct a further experiment to measure the impact of the training-population size on the performance of *DeepEyedentificationLive*. We train the network on 25, 50, 75, 100, and 125 users using a softmax output layer with as many output units. We conduct experiments in which (a) the total volume of training data increases proportionally to the amount of training users and in which (b) we keep the total volume constant by reducing the duration of gaze data per user proportionally. Given a maximum training time of 28,800 seconds, a training setup with 25 users includes 1152 seconds per user, whereas for a training setup with 100 users 288 seconds per user are included.

Figure 5a shows equal-error rates for both settings. In-

TABLE 5

Identification accuracies ± standard error (in %) on the CEM data sets with different stimulus types. The second and third rows present our re-implementation of Rigas et al. (2016) and Holland and Komogortsev (2013), the bottom row shows the numbers reported by Holland and Komogortsev [45, Table 4]. Values marked "*" are significantly worse ($p < 0.05$) than results for DeepEyedentificationLive. Values marked "†" are significantly better ($p < 0.05$) than results for DeepEyedentificationLive. Table adapted from Prasse *et al.* [31].

	CEM-I				CEM-II	CEM-III			
Method	SIM	COM	COG	TEX	TEX	SIM	COM	RAN	TEX
DeepEyedentificationLive	63 ± 2	**67 ± 3**	**56 ± 2**	**75 ± 2**	**80 ± 2**	**34 ± 1**	**37 ± 1**	**43 ± 1**	**35 ± 1**
Rigas et al. (ours)	67 ± 2	37 ± 2*	55 ± 2	41 ± 1*	78 ± 1	8 ± 2*	8 ± 2*	5 ± 1*	6 ± 1*
Holland & Komogortsev (ours)	**68 ± 2**†	31 ± 2*	48 ± 2*	33 ± 2*	71 ± 2*	8 ± 2*	7 ± 2*	5 ± 1 *	5 ± 1*
Holland & Komogortsev	53	22	19	31	38	7	5	5	4

TABLE 6

Comparison of *DeepEyedentificationLive* to published results [28] for a stimulus of 60 seconds on the *GazeBase* TEX dataset.

	TEX
DeepEyedentificationLive EER @ 60 s	**0.047** ± **0.003**
Lohr *et al.* [28] EER @ 60 s	**0.063** ± **0.010** [28]
Friedman *et al.* [22] EER @ 60 s	**0.047** ± **0.018** [**28**]

creasing the number of different users during the initial training stage appears to improves the model performance even if the total volume of training data stays constant, and even more so if the volume of training data increases. The differences between measurement points are not statistically significant for a constant amount of training data and the p-value of a comparison between 125 and 25 users for an increasing amount of training data reaches 0.058 based on a two-sided t-test. An uptick in EER for 75 users is not statistically significant and likely due to chance. While we can see diminishing returns per additional user for larger training population, the curve does not appear to level off—much larger training populations will likely result in more accurate models. The shape of this curve for substantially larger training populations remains to be explored in large-scale experiments.

6.3.4 Impact of Data Volume per User

In the next experiment, we explore the impact of the duration of gaze data per training user and the data volume during enrollment on the model's accuracy.

First, we keep the number of training users at a constant number of 125, while increasing the data volume per training user from 15 to 300 seconds. Figure 5b demonstrates clearly that increasing the training data volume per user leads to better model performance; at a maximum duration of 300 seconds, the curve shows no sign of leveling off. The difference between 15 and 300 training users is significant ($p < 0.001$) based on a two-sided t-test.

In the next experiment, we train the model on all available data and vary the duration of enrollment gaze sequences between 15 seconds and 300 seconds. Figure 5b indicates no benefit of increasing the enrollment sequence beyond 60 seconds; differences are not statistically significant

6.3.5 Impact of Session Bias

We explore the network's ability to generalize across recording sessions. Addressing this topic is relevant because oculomotor control may be influenced by the user's mental state, the time of day, and other confounding factors.

As there are exactly four sessions available for each user in the *JuDo1000* data set, we evaluate on this available range of number of sessions. We first examine the number of training sessions by randomly drawing a particular number of training sessions and using all the data per session. In a second run, we inversely adjust the duration of training data per session to keep the overall training-data volume constant.

We observe in Figure 5c that increasing the number of sessions per training user improves the model's performance. The effect is even more strongly pronounced if the total data volume increases with the number of sessions.

Finally, we train the network on all four sessions of training users and vary the number of sessions from which enrollment sequences are drawn between 1 and 4. We draw 24 seconds per user from the enrollment sessions for enrollment and test the model on an unseen session. We can observe an almost linear descent in EER when varying the used number of enrollment sessions.

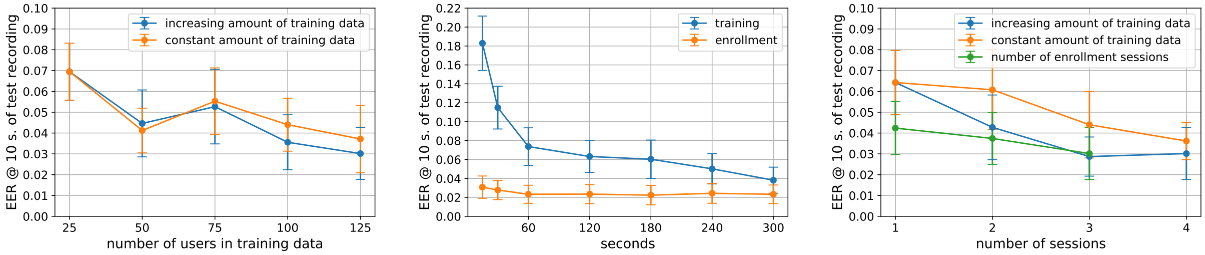
We conclude that in the studied range of 1 through 4 sessions, increasing the number of recordings sessions per training user, and increasing the number of enrollment sessions reduces the equal error rate; the incremental benefit of additional sessions appears to level off after 3 sessions. Differences are not statistically significant.

6.3.6 Impact of Stimulus Type

We conduct a further experiment to evaluate various stimuli types in regard to the performance of *DeepEyedentificationLive*. The broad range of visual stimuli used in the *GazeBase* data set is ideal for such an evaluation. The training populations contain 100 randomly drawn users for whom at least two sessions are available. As in the experiments before, we create a test population of 25 users, where a single user is to be enrolled and verified while the users of the remaining population are treated as impostors. We enroll each test user in turn with data from 3 of the first 4 sessions and verify on a randomly drawn session from sessions 5 to 8.

Table 7 shows that the equal error rate varies widely across stimuli types. The differences get more pronounced by increasing the length of test recordings.

The equal error rate is lowest for the reading task (TEX), followed by the game (BLG), and horizontal saccade task (HSS). Error rates are substantially higher for the random saccade (RAN), and video viewing tasks (VD1 and VD2).



(a) Varying size of training population. (b) Varying recording time per user. (c) Varying number of recording sessions.

Fig. 5. Identity verification performance of the DeepEyedentificationLive model on the *Judo1000* data set. Error bars show the standard error.

The fixation task (FXS) results in the highest error rate. We conclude that unconstrained eye movements such as during reading are best suitable for biometric identification. While watching jumping dots or videos, eye movements are more strongly influenced by the stimulus; forced fixations lead to the least variability across persons and are least suitable for identification.

6.3.7 Impact of the Time Interval Between Sessions

We want to study whether the time interval between enrollment and identity verification has an influence on the model performance. To this end, we evaluate *DeepEyedentificationLive* on the *GazeBase* data set, where time lags between sessions account for up to 32 months.

The dates of each session are specified to within ranges. We use sessions 1 to 4 for enrollment. The resulting time lag intervals available in our evaluation are therefore 5 to 14 months (sessions 1-4 to session 5), 11 to 20 months (sessions 1-4 to session 6), 18 to 26 months (sessions 1-4 to session 7) and 23 to 32 months (sessions 1-4 to session 8).

The resulting Figure 6 does not exhibit any correlation between interval length and equal error rate. However, we observe that the equal error rate varies across sessions; sessions 6 and 8 appear to be "harder" whereas sessions 5 and 7 are "easier". Inspection of the data shows that "harder" sessions have a higher rate of tracker loss, indicating that the data quality is the dominant factor.

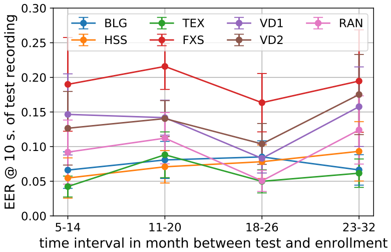


Fig. 6. Performance metrics ± standard errors of the DeepEyedentificationLive model for different stimuli on the *GazeBase* data set. Users are enrolled using 3 of the first 4 sessions and tested on a session in the future. Error bars show the standard error.

6.4 Presentation-Attack Detection

We simulate an attacker who has observed the number of stimuli, display duration, and the size of the display area, and who is able to record and replay, without detectable imperfections, a gaze sequence of the target individual for this exact configuration or is able to control the input to the system. Since we use a randomized stimulus, the attacker does not know the display positions of the jumping dot.

We evaluate presentation-attack detection on the *JuDo1000* and *GazeBase* data set. We randomly resample 10 times, in each iteration selecting 125 users for training and 25 users for testing. At test time, a decision is made based on an input recording of one trial. We generate examples of attacks by pairing a test gaze sequence with a stimulus sequence for which the positions have been randomly drawn with the same display duration and area. For each *bona fide* presentation in the data, we create one attack presentation.

We compare DeepEyedentificationLive against a simple *heuristic* and a *Random Forest* model with engineered features. The heuristic is based on Equation 3; it measures, how well the fixation sequence matches the sequence of stimuli by computing the average, over four stimulus relocations, of the differences between the aggregate gaze movements during presentation of the stimulus and the offset between current and last stimulus. Based on a threshold, a pair of fixation sequence and stimuli is classified as *attack* or *bona fide*. The *Random Forest* baseline uses the four differences between aggregate eye movements and stimulus offsets during presentation of one of the stimuli and the average of these values as features.

Prior work on presentation-attack detection for gaze-based identification assumes that the presentation is generated with imperfections in the distributional features of the user's fixation durations and amplitude and velocity features of saccades [39], [40]. In our setting, the attacker replays a recorded gaze sequence of the target user. Therefore, this known approach cannot distinguish these replay attacks from *bona fide* presentations and we do not include it in the experimental comparison. Reference methods that detect specific artifacts that are indicative of a particular presentation-attack instrument are generally ineffective against different attack instruments. For instance, prior work that exploits phase information which is indicative of smartphone screens [46], or other methods that detect video presentations by detecting mobile devices cannot detect a presentation attack using 3D-printed eyeball replicas. We therefore do not believe that including these methods in our experimental comparison would provide new insights.

To investigate the influence of display durations on

TABLE 7
Performance metrics ± standard errors of DeepEyedentificationLive on the *GazeBase* on different stimuli. Users are enrolled using 3 of the first 4 sessions and tested on a randomly selected session from 5-8.

	RAN	TEX	FXS	VD1	VD2	BLG	HSS
EER @ 1 s	0.218 ± 0.040	**0.162 ± 0.016**	0.333 ± 0.047	0.287 ± 0.047	0.264 ± 0.038	0.193 ± 0.024	0.163 ± 0.032
EER @ 5 s	0.163 ± 0.047	**0.100 ± 0.021**	0.297 ± 0.059	0.219 ± 0.052	0.208 ± 0.050	0.103 ± 0.026	0.114 ± 0.033
EER @ 10 s	0.148 ± 0.049	**0.074 ± 0.021**	0.259 ± 0.065	0.185 ± 0.052	0.188 ± 0.054	0.081 ± 0.025	0.102 ± 0.033

presentation-attack detection performance, we evaluate the models on three subsets of the data (using only trials with 250 ms, 500 ms or 1000 ms display duration in training and test respectively). As Figure 7 and Table 8 show, we attain the lowest EER of 0.011, when only using data from trials with 500 ms stimulus display duration and an EER of 0.041 when using all experimental configurations. The performance gap between DeepEyedentificationLive and both baseline methods is dramatic. The last row of Table 8 and Figure 8 shows the results for presentation-attack detection on the RAN task of *GazeBase*. We observe that presentation-attack detection performance is comparable to the performance on *JuDo1000* in the configuration of a display duration per dot of 1 s.

TABLE 8
Presentation-attack detection with one trial as input at test time. Table shows EER for different display durations (250ms, 500ms, 1000ms) of the five dots. Time in seconds denotes resulting trial length. Values marked "*" are significantly worse ($p < 0.05$) than results for DeepEyedentificationLive.

	DeepEye.Live	**Random Forest**	**Heuristic**
EER	**0.041 ± 0.004**	0.171 ± 0.005*	0.202 ± 0.009*
EER @ $1.25s$	**0.073 ± 0.003**	0.208 ± 0.014*	0.225 ± 0.016*
EER @ $2.5s$	**0.011 ± 0.004**	0.146 ± 0.005*	0.175 ± 0.009*
EER @ $5.0s$	**0.051 ± 0.004**	0.160 ± 0.006*	0.204 ± 0.006*
EER @ $5.0s$ (*GazeBase*)	**0.021 ± 0.001**	0.118 ± 0.001*	0.121 ± 0.001*

7 Conclusion

We have developed *DeepEyedentificationLive*, a convolutional network for oculomotoric biometric identification that processes both the controlled stimulus and the binocular response in the form of sequences of gaze velocities. The model determines an embedding of gaze sequences and simultaneously performs presentation-attack detection.

We conclude that *DeepEyedentificationLive* dramatically outperforms reference methods that extract explicit saccadic and fixational features on the *JuDo1000* data set that we recorded with a high-precision eye tracker as well as on most viewing tasks of the legacy CEM-I and CEM-III data sets that were recorded with low sampling rates and precision. By processing the low-velocity, high-frequency micro-movements in a separate sub-network, DeepEyedentificationLive is able to automatically identify micro-movement features that vary across individuals. DeepEyedentificationLive only receives gaze velocities as input and hence is insensitive to user-specific offsets that otherwise would have to be compensated by calibration.

Deep neural networks for biometric identification require a separate set of training users to train the embedding. From our experiments we can conclude that *more data is better* in almost any aspect: increasing the number of training users, the volume of gaze data per training user, the number of sessions per training user and the number of enrollment sessions continues to improve the model's performance across the studied range. By contrast, increasing the volume enrollment data beyond 60 seconds for a user does not appear to have a beneficial effect. We do not observe any performance deterioration when the interval between enrollment and identity verification grows to as much as 32 months.

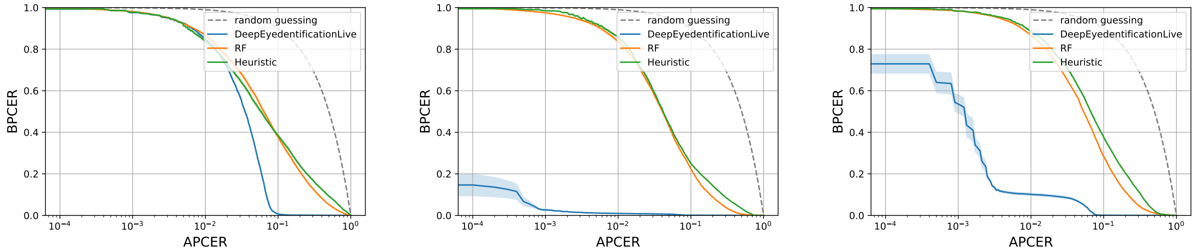
With regard to the stimulus type, we conclude that unconstrained viewing tasks such as reading a text are best suitable for biometric identification whereas watching jumping dots or videos, or performing forced fixations are more challenging because the scanpath is largely determined by the stimulus. By showing randomized jumping dots, *DeepEyedentificationLive* is able to compare a controlled visual stimulus to the induced ocular response at the cost of identity verification becoming more challenging. We have demonstrated that *DeepEyedentificationLive* is able to identify users even in this setting.

Our model of an attacker who is informed about the display duration and area and can replay gaze sequences of the target individual without imperfection is arguably the most challenging attacker model. We conclude that five stimuli displayed for 500 ms each are the best configuration for presentation-attack detection under investigation.

We conclude that using eye movements bears high potential for applications that require a fast and unobtrusive identification. Eye movements are a necessary prerequisite of vision, and are therefore available for a large fraction of the population. Eye movements are orthogonal to established biometric features, but could potentially be measured using the same infrared sensors that can also be used for iris scans or face recognition. Hence, it might complement these technologies in a multimodal biometric system that could be more robust to colored contact lenses and small eye apertures—which may prevent the use of iris scans—and niqabs and masks, which pose problems for facial identification. We made the *JuDo1000* data set available to the research community.

Acknowledgments

This work was partially funded by the German Science Foundation under grant SFB 1294 and the German Federal Ministry of Education and Research under grant 01|S20043.



(a) *JuDo1000*, 250 ms per dot (b) *JuDo1000*, 500 ms per dot (c) *JuDo1000*, 1 s per dot

Fig. 7. Presentation-attack detection performance with one trial as input at test time using data from trials with a dot on-screen duration of 250, 500, and 1000 ms from the *Judo1000* data set. Bona-Fide Presentation-Classification Error Rate (BPCER) over Attack-Presentation Classification-Error Rate (APCER). Colored bands show the standard error.

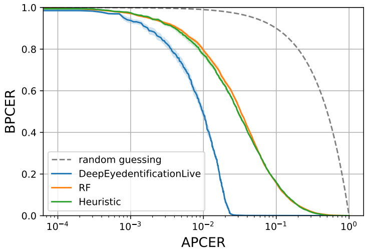


Fig. 8. Presentation-attack detection performance with one trial as input at test time using data from trials with a dot on-screen duration of one second from the *GazeBase* data set. Bona-Fide Presentation-Classification Error Rate (BPCER) over Attack-Presentation Classification-Error Rate (APCER). Colored bands show the standard error.

References

[1] A. K. Jain, "Biometric recognition: overview and recent advances," in *Iberoamerican Congress on Pattern Recognition*. Springer, 2007, pp. 13–19.

[2] D. Noton and L. Stark, "Scanpaths in eye movements during pattern perception," *Science*, vol. 171, no. 3968, pp. 308–311, 1971.

[3] G. Bargary, J. M. Bosten, P. T. Goodbourn, A. J. Lawrance-Owen, R. E. Hogg, and J. Mollon, "Individual differences in human eye movements: An oculomotor signature?" *Vision Research*, vol. 141, pp. 157–169, 2017.

[4] P. Kasprowski and J. Ober, "Eye movements in biometrics," in *International Workshop on Biometric Authentication*, 2004, pp. 248–258.

[5] R. Bednarik, T. Kinnunen, A. Mihaila, and P. Fränti, "Eye-movements as a biometric," in *SCIA 2005*, 2005, pp. 780–789.

[6] S. Martinez-Conde, S. L. Macknik, and D. H. Hubel, "The role of fixational eye movements in visual perception," *Nature Reviews Neuroscience*, vol. 5, p. 229–240, 2004.

[7] S. Martinez-Conde, S. L. Macknik, X. G. Troncoso, and T. A. Dyar, "Microsaccades counteract visual fading during fixation," *Neuron*, vol. 49, pp. 297–305, 2006.

[8] S. Martinez-Conde, S. L. Macknik, X. G. Troncoso, and D. H. Hubel, "Microsaccades: A neurophysiological analysis," *Trends in Neurosciences*, vol. 32, pp. 463–475, 2009.

[9] J. Otero-Millan, X. G. Troncoso, S. L. Macknik, I. Serrano-Pedraza, and S. Martinez-Conde, "Saccades and microsaccades during visual fixation, exploration, and search: Foundations for a common saccadic generator," *Journal of Vision*, vol. 8, no. 14, pp. 21–21, 2008.

[10] K. Holmqvist, M. Nyström, R. Andersson, R. Dewhurst, H. Jarodzka, and J. Van de Weijer, *Eye tracking: A comprehensive guide to methods and measures.* Oxford: Oxford University Press, 2011.

[11] S. Makowski, L. A. Jäger, A. Abdelwahab, N. Landwehr, and T. Scheffer, "A discriminative model for identifying readers and assessing text comprehension from eye movements," in *Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2018. Lecture Notes in Computer Science*. Springer, Cham, 2019, pp. 209–225.

[12] L. A. Jäger, S. Makowski, P. Prasse, S. Liehr, M. Seidler, and T. Scheffer, "Deep Eyedentification: Biometric identification using micro-movements of the eye," in *Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2019. Lecture Notes in Computer Science.* Springer, Cham, 2020, pp. 299–314.

[13] S. Makowski, L. A. Jäger, P. Prasse, and T. Scheffer, "Biometric identification and presentation-attack detection using micro-movements of the eyes," in *Proceedings of the IEEE International Joint Conference on Biometrics (IJCB)*, 2020, pp. 1–10.

[14] P. Kasprowski, "The impact of temporal proximity between samples on eye movement biometric identification," in *Proceedings of the IFIP International Conference on Computer Information Systems and Industrial Management*, 2013, pp. 25–27.

[15] P. Kasprowski and K. Harkeżlak, "The second eye movements verification and identification competition," in *Proceedings of the International Joint Conference on Biometrics (IJCB)*, 2014.

[16] C. Holland and O. V. Komogortsev, "Biometric identification via eye movement scanpaths in reading," in *Proceedings of the International Joint Conference on Biometrics (IJCB)*, 2011, pp. 1–8.

[17] N. Cuong, V. Dinh, and L. S. T. Ho, "Mel-frequency cepstral coefficients for eye movement identification," in *ICTAI 2012*, 2012, pp. 253–260.

[18] D. L. Silver and A. Biggs, "Keystroke and eye-tracking biometrics for user identification," in *ICAI 2006*, vol. 2, 2006, pp. 344–348.

[19] I. Rigas, G. Economou, and S. Fotopoulos, "Biometric identification based on the eye movements and graph matching techniques," *Pattern Recogntion Letters*, vol. 33, pp. 786–792, 2012.

[20] C. Holland and O. Komogortsev, "Complex eye movement pattern biometrics: Analyzing fixations and saccades," in *ICB 2013*, 2013.

[21] I. Rigas, O. Komogortsev, and R. Shadmehr, "Biometric recognition via eye movements: Saccadic vigor and acceleration cues," *ACM Transactions on Applied Perception*, vol. 13, no. 2, p. 6, 2016.

[22] L. Friedman, M. S. Nixon, and O. V. Komogortsev, "Method to assess the temporal persistence of potential biometric features: Application to oculomotor, gait, face and brain structure databases," *PLOS ONE*, vol. 12, no. 6, pp. 1–42, 06 2017. [Online]. Available: https://doi.org/10.1371/journal.pone.0178501

[23] H.-J. Yoon, T. R. Carmichael, and G. Tourassi, "Gaze as a biometric," in *SPIE Medical Imaging Conference: Image Perception, Observer Performance, and Technology Assessment*, 2014.

[24] H. Yoon, T. Carmichael, and G. Tourassi, "Temporal stability of visual search-driven biometrics," in *SPIE Medical Imaging: Image Perception, Obserformance, and Technology Assessment*, 2015.

[25] N. Landwehr, S. Arzt, T. Scheffer, and R. Kliegl, "A model of individual differences in gaze control during reading," in *Proceedings of Empirical Methods in Natural Language Processing*, 2014, pp. 1810–1815.

[26] A. Abdelwahab, R. Kliegl, and N. Landwehr, "A semiparametric model for Bayesian reader identification," in *Proceedings of Empirical Methods in Natural Language Processing*, 2016, pp. 585–594.

[27] A. George and A. Routray, "A score level fusion method for eye movement biometrics," *Pattern Recognition Letters*, vol. 82, pp. 207–215, 2016.

[28] D. Lohr, H. Griffith, S. Aziz, and O. Komogortsev, "A metric

learning approach to eye movement biometrics," in *2020 IEEE International Joint Conference on Biometrics (IJCB)*, 2020, pp. 1–7.

[29] A. Abdelwahab and N. Landwehr, "Quantile layers: Statistical aggregation in deep neural networks for eye movement biometrics," in *Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2019. Lecture Notes in Computer Science.* Cham: Springer International Publishing, 2020, pp. 332–348.

[30] ——, "Deep Distributional Sequence Embeddings Based on a Wasserstein Loss," *arXiv:arXiv:1912.01933*, 2019.

[31] P. Prasse, L. A. Jäger, S. Makowski, M. Feuerpfeil, and T. Scheffer, "On the relationship between eye tracking resolution and performance of oculomotoric biometric identification," *Procedia Comput. Sci.*, vol. 176, pp. 2088–2097, 2020.

[32] A. Darwish and M. Pasquier, "Biometric identification using the dynamic features of the eyes," in *BTAS 2013*, 2013, pp. 1–6.

[33] C. Galdi, M. Nappi, D. Riccio, V. Cantoni, and M. Porta, "A new gaze analysis based softbiometric," in *MCPR 2013*, 2013, pp. 136–144.

[34] V. Cantoni, C. Galdi, M. Nappi, M. Porta, and D. Riccio, "GANT: Gaze analysis technique for human identification," *Pattern Recognition*, vol. 48, pp. 1027–1038, 2015.

[35] T. Kinnunen, F. Sedlak, and R. Bednarik, "Towards task-independent person authentication using eye movement signals," in *ETRA 2010*, 2010, pp. 187–190.

[36] P. Kasprowski, "Human identification using eye movements," Ph.D. dissertation, Silesian Unversity of Technology, 2004.

[37] I. Rigas, G. Economou, and S. Fotopoulos, "Human eye movements as a trait for biometrical identification," in *BTAS 2012*, 2012, pp. 217–222.

[38] N. Srivastava, U. Agrawal, S. Roy, and U. S. Tiwary, "Human identification using linear multiclass SVM and eye movement biometrics," in *IC3 2015*, 2015, pp. 365–369.

[39] O. V. Komogortsev, A. Karpov, and C. Holland, "CUE: Counterfeitresistant usable eye-based authentication via oculomotor plant characteristics and complex eye movement patterns," in *SPIE Defence Security and Sensing Conference on Biometric Technology for Human Identification IX*, 2012, pp. 1–10.

[40] O. V. Komogortsev, A. Karpov, and C. D. Holland, "Attack of mechanical replicas: Liveness detection with eye movements," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 4, pp. 716–725, 2015.

[41] K. B. Raja, R. Raghavendra, and C. Busch, "Video presentation attack detection in visible spectrum iris recognition using magnified phase information," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 10, pp. 2048–2056, 2015.

[42] S. Makowski, L. A. Jäger, and T. Scheffer, "Judo1000 eye tracking data set," https://osf.io/5zpvk/.

[43] C. D. Holland and O. V. Komogortsev, "Biometric verification via complex eye movements: The effects of environment and stimulus," in *BTAS 2012*, 2012, pp. 39–46.

[44] H. Griffith, D. Lohr, E. Abdulin, and O. Komogortsev, "GazeBase: A Large-Scale, Multi-Stimulus, Longitudinal Eye Movement Dataset," pp. 1–9, 2020. [Online]. Available: http://arxiv.org/abs/2009.06171

[45] C. D. Holland and O. V. Komogortsev, "Complex eye movement pattern biometrics: The effects of environment and stimulus," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 12, pp. 2115–2126, 2013.

[46] K. B. Raja, R. Raghavendra, and C. Busch, "Video presentation attack detection in visible spectrum iris recognition using magnified phase information," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 10, pp. 2048–2056, 2015.

Silvia Makowski is a Ph.D. student in computer science at the University of Potsdam. She received a master's degree in computational science from the University of Potsdam in 2017.

Paul Prasse is a Postdoctoral Researcher at the University of Potsdam. He received a master's degree in computer science (Diplominformatiker) in 2010 and a doctoral degree in 2016 from the University of Potsdam.

David R. Reich is a Ph.D. student with Lena A. Jäger in the BMBF-project AEye. He joined the University of Potsdam in 2020 after receiving his bachelor's and master's degree in mathematics from Freie Universität and Technische Universität Berlin, respectively.

Daniel Krakowczyk is a Ph.D. student at the Machine Learning Lab at the University of Potsdam since October 2020. He obtained his master's degree at the same university and his bachelor's degree at Freie Universität Berlin.

Lena A. Jäger is Associate Professor at the Department of Computational Linguistics, University of Zurich and leader of the Junior Research Group *Artificial Intelligence for Eye Tracking Data* (AEye) at the Department of Computer Science, University of Potsdam. She received a doctoral degree in Cognitive Science from the University of Potsdam in 2015.

Tobias Scheffer is a Professor of Computer Science at the University of Potsdam. He received a doctoral degree in 1999 from Technische Universität Berlin. He served as Program Co-Chair of the International Conference on Machine Learning in 2011 and of ECML PKDD in 2006.