IMT School for Advanced Studies, Lucca
Lucca, Italy

From Subpixel Accuracy to Scanpaths Analysis: Smart Strategies for Implementing Deep Learning Algorithms in Eye Movement Research and Applications

PhD Program in Cognitive, Computational and Social Neurosciences

XXXV Cycle

By

Sean Anthony Byrne

2023

The dissertation of Sean Anthony Byrne is approved.

PhD Program Coordinator: Emiliano Ricciardi, IMT School for Advanced Studies Lucca

Advisor: Prof. Massimo Riccaboni, IMT School for Advanced Studies Lucca

Co-Advisor: Prof. Luca Polonio, Università degli Studi di Milano Bicocca

Co-Advisor: Prof. Enkelejda Kasneci, Technical University of Munich

Co-Advisor: Prof. Diederick Niehorster, Lund University

The dissertation of Sean Anthony Byrne has been reviewed by:

Adam Zylbersztejn, Université Lumiere Lyon 2

Prof. Arantxa Villanueva, Public University of Navarre

IMT School for Advanced Studies Lucca
2023

For Sandra, David, and Niamh, simply, thank you.

Contents

List of Figures

List of Tables

Acknowledgements

First and foremost, I'd like to express my gratitude to my parents, Sandra and David, as well as my sister Niamh. Please believe me when I say that I couldn't have completed my PhD without their unwavering support. To be honest, I'm just as surprised as all of you that I've managed to finish.

Secondly, I would like to express my gratitude to my supervisors, of whom there are many: Massimo Riccaboni, Luca Polonio, Enkelejda Kasneci, and Diederick Niehorster. I am also grateful to my many co-authors, including Adam Peter Frederick Reynolds, Carolina Biliotti, Efe Bozkir, Falco J Bargagli-Stoffi, Virmarie Maquiling, Luca Polonio, Nora Castner, Marcus Nyström, Ard Kastrati, Martyna Beata Płomecka, William Schaefer, and Zoya Bylinskii for their contributions to the work that led to the creation of this thesis.

During my time as a PhD student, I was fortunate enough to visit many great institutions, such as the Technical University of Munich, Lund University, and the University of Tübingen. I would like to thank everyone who helped make that happen, with a special mention to Enkelejda Kasneci for always being a willing host, Diederick Niehorster and Marcus Nyström for working closely with me on the gaze estimation projects and making them a lot of fun. Finally, I want to express my gratitude to Nora Jane Castner, Kai Otto and Mori for making my time in Tübingen so fun and productive, and for the best cappuccino in Germany.

Vita

November, 1992	Born, Dublin, Ireland
2017	Degree in Economics & Finance Final mark: 2.1 Technical University of Dublin, Ireland
2018	MSc. Behavioural Economics Final mark: 2.1 University College Dublin, Ireland

Publications [1]

1. *Sean Anthony Byrne, Marcus Nyström, et al. (Dec. 2023). "Precise Localization of Corneal Reflections in Eye Images Using Deep Learning Trained on Synthetic Data". In: *Behavior Research Methods*. ISSN: 1554-3528. DOI: 10.3758/s13428-023-02297-w. URL: https://doi.org/10.3758/s13428-023-02297-w *

2. *Sean Anthony Byrne, Virmarie Maquiling, Marcus Nyström, et al. (2023). *LEyes: A Lightweight Framework for Deep Learning-Based Eye Tracking using Synthetic Eye Images*. arXiv: 2309.06129 [cs.CV] *

3. *Sean Anthony Byrne, Adam Peter Frederick Reynolds, et al. (2023). "Predicting choice behaviour in economic games using gaze data encoded as scanpath images". In: *Scientific Reports* 13.1, p. 4722 *

4. *Sean Anthony Byrne, Virmarie Maquiling, Adam Peter Frederick Reynolds, et al. (May 2023). "Exploring the Effects of Scanpath Feature Engineering for Supervised Image Classification Models". In: *Proc. ACM Hum.-Comput. Interact.* 7.ETRA. DOI: 10.1145/3591130. URL: https://doi.org/10.1145/3591130 *

5. Sean Anthony Byrne, Nora Castner, Ard Kastrati, et al. (2023). "Leveraging Eye Tracking in Digital Classrooms: A Step Towards Multimodal Model for Learning Assistance". In: *Proceedings of the 2023 Symposium on Eye Tracking Research and Applications*. ETRA '23. Tubingen, Germany: Association for Computing Machinery. ISBN: 9798400701504. DOI: 10.1145/3588015.3589197. URL: https://doi.org/10.1145/3588015.3589197

6. Virmarie Maquiling, Sean Anthony Byrne, Marcus Nyström, et al. (Sept. 2023). "V-ir-Net: A Novel Neural Network for Pupil and Corneal Reflection Detection trained on Simulated Light Distributions". In: *25th International Conference on Mobile Human-Computer Interaction (MobileHCI '23 Companion)*. Athens, Greece: ACM. ISBN: 978-1-4503-9924-1/23/09. DOI: 10.1145/3565066.3608690

7. Virmarie Maquiling, Sean Anthony Byrne, Diederick C. Niehorster, et al. (2023). *Zero-Shot Segmentation of Eye Features Using the Segment Anything Model (SAM)*. arXiv: 2311.08077 [cs.CV]

[1] Asterisks $*$ denote the papers used as the basis of chapters in this dissertation

8. Sean Anthony Byrne, Nora Castner, Efe Bozkir, et al. (2024). “From Lenses to Living Rooms: A Policy Brief on Eye Tracking in XR Before the Impending Boom”. In: *2024 IEEE International Conference on Artificial Intelligence and eXtended and Virtual Reality (AIxVR)*. IEEE, pp. 90–96

Presentations

1. "Deep Learning in Eye Tracking" at *Yachay Scientific Computing Summer School*, Yachay Tech University, Ecuador *(online)* 2021.
2. "Anticipating Choice Behaviour in Strategic Settings via Machine Learning Modeling of Scanpath Subsequences " ECEM 2022
3. "Exploring the Effects of Scanpath Feature Engineering for Supervised Image Classification Model " ETRA 2023
4. "Leveraging Eye Tracking in Digital Classrooms: A Step Towards Multimodal Model for Learning Assistance " ETRA 2023

Abstract

Eye-tracking research has been influential across various sectors, encompassing both the creation of eye-tracking devices and the analysis of the data they produce. These facets are known as gaze estimation and gaze analysis. The former identifies where an individual is gazing based on images captured by cameras aimed at the eyes, while the latter discerns the duration and sites of gaze, typically using characteristics like saccades and fixations to deduce an individual's cognitive activities. Recently, a significant transformation has taken place with both fields now heavily leaning on deep learning. This integration of deep learning methods has significantly improved precision, efficiency, and adaptability in both realms. It also ushers in advanced implementations, such as real-time gaze forecasting in areas like virtual reality and gaming. Yet, the infusion of deep learning comes with its set of challenges, notably when faced with the limited and often expensive eye-tracking datasets. This dissertation delves into these issues, focusing on the role of deep learning in both gaze estimation and analysis. Amongst the myriad of deep learning techniques for eye tracking, this work highlights two: first, the efficacy of using synthetic data in gaze estimation models and its performance in synthetic and real-world pipelines. Second, within the context of an economic experiment, we investigate the impact of feature engineering for scanpath formulation and the potential to foresee a user's choice before they decide, a concept that holds significance in numerous sectors, especially as eye tracking devices such as virtual headsets gain traction.

Chapter 1

Introduction

[1]

Eye-tracking research spans various fields such as psychology, economics, and health sciences, focusing on Gaze Estimation and Gaze Analysis. Researchers in psychology, economics, and medicine use eye-tracking to study cognitive processes (Polonio, Di Guida, and Coricelli, 2015) and attention allocation (Lee and Ahn, 2012), with applications from Alzheimer's classification (Sriram, Conati, and Field, 2023) to advertisement analysis (Lee and Ahn, 2012). In contrast, gaze estimation researchers in engineering and computer science aim to improve eye-tracking technology efficiency and accuracy, using devices ranging from laboratory head-mounted trackers (De Kloe et al., 2022) to virtual reality systems (Byrne, Maquiling, Nyström, et al., 2023) and "Gaze-in-the-wild" settings (Fuhl,

[1] This introduction is based on the following co-authored work: 1.Sean Anthony Byrne, Marcus Nyström, et al. (Dec. 2023). "Precise Localization of Corneal Reflections in Eye Images Using Deep Learning Trained on Synthetic Data". In: *Behavior Research Methods*. ISSN: 1554-3528. DOI: 10.3758/s13428-023-02297-w. URL: https://doi.org/10.3758/s13428-023-02297-w, 2.Sean Anthony Byrne, Virmarie Maquiling, Marcus Nyström, et al. (2023). *LEyes: A Lightweight Framework for Deep Learning-Based Eye Tracking using Synthetic Eye Images*. arXiv: 2309.06129 [cs.CV], 3.Sean Anthony Byrne, Adam Peter Frederick Reynolds, et al. (2023). "Predicting choice behaviour in economic games using gaze data encoded as scanpath images". In: *Scientific Reports* 13.1, p. 4722, 4.Sean Anthony Byrne, Virmarie Maquiling, Adam Peter Frederick Reynolds, et al. (May 2023). "Exploring the Effects of Scanpath Feature Engineering for Supervised Image Classification Models". In: *Proc. ACM Hum.-Comput. Interact.* 7.ETRA. DOI: 10.1145/3591130. URL: https://doi.org/10.1145/3591130

Santini, Kasneci, Rosenstiel, et al., 2017).

Machine learning has emerged as a unifying principle in gaze analysis and estimation, enhancing data interpretation and improving estimation accuracy beyond traditional methods. This thesis, based on four distinct papers, integrates deep learning in both gaze estimation and analysis. It covers concepts in both fields, illustrating applications from synthetic data-trained CNNs for corneal reflection identification to deep learning-enhanced scanpath analysis in economic games.

This thesis, grounded in four academic papers, is poised to showcase a series of practical use cases for machine learning within the realms of gaze estimation and analysis. Central to this exploration is the innovative use of synthetic data in training deep neural networks for gaze estimation tasks. Additionally, the thesis will delve into the methodologies for optimally displaying gaze information in images, tailored for supervised learning algorithms. This investigation specifically focuses on a singular experimental paradigm: a normal-form economic game. Through this targeted approach, the thesis aims to illuminate the intricate interplay between advanced machine learning techniques and eye-tracking technology, highlighting their potential in both refining gaze estimation methods and enhancing the analysis of gaze data. The thesis begins by introducing the fundamental concepts of gaze estimation and analysis to familiarize the reader with the subject. Subsequent chapters then detail the key findings and methodologies employed in the studies, systematically unfolding the research and its implications. Finally, this dissertation concludes by highlighting how these findings contribute to the field and points to future avenues of research.

1.1 A Brief Introduction to Gaze Estimation

Gaze estimation involves determining where an individual is looking by analyzing eye images to produce gaze direction or point coordinates (Akinyelu and Blignaut, 2020). Its applications are diverse, including VR, healthcare, economics, neuroscience, and education (Garbin et al., 2020; Palmero et al., 2021; Pierce et al., 2016; Byrne, Reynolds, et al., 2023; Byrne, Cast-

ner, Kastrati, et al., 2023). Video-based eye-tracking methodologies, particularly P-CR eye tracking, are a focus of this thesis, along with deep learning techniques in gaze estimation.

Despite traditional methods like thresholding or ellipse fitting being common (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2022; Shortis, Clarke, and Short, 1994), they face limitations in resolution and accuracy (Holmqvist and Blignaut, 2020). Recently, deep learning models have been developed for CR localization, such as multi-task CNNs (Wu et al., 2019), U-NET models (Chugh et al., 2021), and compact models with attention mechanisms (Niu et al., 2021). To address the need for large datasets, synthetic data has been utilized, as seen in works like "learning-by-synthesis" strategies and generative adversarial networks (Sugano, Matsushita, and Sato, 2014; Wood et al., 2015b; Shrivastava et al., 2016). For a complete overview of deep learning methods in gaze estimation, see (Cheng et al., 2021)

1.2 A Brief Introduction to Gaze Analysis

Gaze analysis involves transforming eye movements recorded using gaze estimation techniques to create cognitively relevant features such as fixations and saccades. Fixations occur when the eye remains focused on a single point or object, while saccades are rapid movements shifting focus from one point to another (Holmqvist, Nyström, Andersson, et al., 2011; Hessels et al., 2018; Rayner, 1998). Analyzing these features using metrics such as through fixation duration or saccade direction and amplitude, provides insights into the attention and visual search efficiency (Einhäuser, Rutishauser, Koch, et al., 2008) of the recorded subject. Building on these features additional tools such as heatmaps and Areas of Interest (AOIs) offer deeper understanding of visual attention distribution and object-focused gaze behavior (Blignaut, 2010; Pfeiffer and Memili, 2016; Jarodzka, Holmqvist, and Nyström, 2010; Clay, König, and Koenig, 2019; Naspetti et al., 2016; Niehorster et al., in press).

A scanpath is created through a series of continuous fixations and saccades, forming a trace that evolves over time and space and may

intersect with its own path (Goldberg and Helfman, 2010). Scanpath analysis, comparing gaze behavior across different contexts or tasks, has proven useful in various domains, from art viewing to medicine, highlighting diverse human visual strategies (Buswell, 1935; Yarbus, 1967; Byrne, Maquiling, Reynolds, et al., 2023; Byrne, Reynolds, et al., 2023; Stein, Jossberger, and Gruber, 2022; Strukelj and Niehorster, 2018; Liszio and Masuch, 2016; Popelka and Beitlova, 2022; Castner, Kuebler, Scheiter, Richter, Eder, et al., 2020; Brunyé et al., 2019). Focusing on image classification scanpaths, various deep learning models have been employed, including CNNs and LSTM networks, for conditions like ASD and schizophrenia classification (Elbattah et al., 2019; Praveena and Mahalakshmi, 2022; Kacur et al., 2020; Vortmann et al., 2021; Tao and Shyu, 2019; Chen and Zhao, 2019). Techniques such as Gaussian blurring are used for data normalization. Additionally, alternative representations of scanpaths, like Markov Transition Fields and graphs, have been explored (Geisler et al., 2020; Coutrot, Hsiao, and Chan, 2018; Venuprasad et al., 2020; Kacur et al., 2020; Kumar et al., 2020). Researchers have experimented with creating images from raw scanpath data to preserve information relevant for models (Kacur et al., 2020). Examples include scanpath images for CNN and RNN inputs (Sims and Conati, 2020; Ahmed and Jadhav, 2020), generative models using emojis (Fuhl, Bozkir, Hosp, Castner, Geisler, Santini, et al., 2019a), and novel methods employing temporal coloring for ASD classification (Atyabi et al., 2022). A study by (Bhattacharya et al., 2020) achieved up to 80% accuracy in predicting text relevancy through eye movement behavior, illustrating the potential of these approaches. For a complete overview of gaze analysis particularly referring to decision making (Wedel, Pieters, and Lans, 2023).

Chapter 2

Enhancing Accuracy in P-CR Eye Tracking: A Deep Dive into Synthetic Data Application using a Single Corneal Reflection-Based Study

[1]

In many video-based eye-tracking systems, a crucial step within the image processing sequence is to pinpoint the center of specific ocular features, predominantly the pupil (P) (Fuhl, Santini, Kasneci, Rosenstiel, et al., 2017; Li, Winfield, and Parkhurst, 2005) and one or several corneal reflections (CRs) (Peréz et al., 2003; Nyström, Niehorster, Andersson, Hes-

[1] This discussion is grounded in a co-authored manuscript Sean Anthony Byrne, Marcus Nyström, et al. (Dec. 2023). "Precise Localization of Corneal Reflections in Eye Images Using Deep Learning Trained on Synthetic Data". In: *Behavior Research Methods*. ISSN: 1554-3528. DOI: 10.3758/s13428-023-02297-w. URL: https://doi.org/10.3758/s13428-023-02297-w which is published in the Journal "Behavioural Research Methods", where the author of this dissertation is the first author.

sels, and Hooge, 2022; Chugh et al., 2021; Byrne, Maquiling, Nyström, et al., 2023). The precision in locating these features directly influences the accuracy of the gaze data generated by the eye-tracking device.

In this chapter, our primary focus is on pinpointing the center of a solitary CR, which, in conjunction with the pupil center, has formed the foundation for video-based eye tracking for many decades (Merchant, Morrissette, and Porterfield, 1974). This methodology, termed P-CR eye tracking, has been widely adopted in premier commercial systems, exemplified by the EyeLink from SR Research (Ontario, Canada), where the individual's head is typically stabilized using chin and forehead supports.

This chapter aims to investigate the usefulness of the application of synthetic data to train deep learning models for CR center identification answering the following questions: 1) In the context of synthetic data, does our methodology outperform traditional algorithmic techniques in terms of precision? 2) Is a CNN, when trained on synthetic images, capable of localizing the CR center in real eye images? 3) When our method is applied to real eye images, does it offer a CR position reading with superior fidelity compared to conventional algorithmic methods? Our investigation specifically homes in on high resolutions eye images. This is pivotal because, in top-tier eye trackers, precise CR identification is essential to ensure that even the minutest and slowest ocular activities, such as microsaccades and slow pursuits, can be reliably differentiated from random fluctuations (c.f., Holmqvist and Blignaut, 2020; Nyström, Niehorster, Andersson, Hessels, and Hooge, 2022; Niehorster, Zemblys, and Holmqvist, 2021a). Additionally, in an effort to gauge the broader applicability of our technique to images of lesser resolution, we also test our methodology on spatially reduced eye images in a subsequent experiment detailed herein.

Historically, researchers employed algorithmic techniques, consisting of predefined sequences of steps, to determine feature centers in input images. Nonetheless, such methods have shown limitations, especially when discerning the pupil or CR amidst image noise (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2022). See the 1.1 literature

review for a detailed review of gaze estimation techniques.

In response, we present a deep learning approach, trained on synthetic eye images, proficient in accurately pinpointing CR centers in real-world ocular images. Using synthetic data circumvents the longstanding challenge of sourcing or curating extensive, annotated datasets typically needed for deep learning. Our method not only surpasses conventional algorithmic techniques but also pioneers a novel paradigm for those keen on employing deep learning for gaze estimation. This method is straightforward to train and can be customized according to user-specific synthetic image needs. Furthermore, it obviates the requirement for vast datasets and the labor-intensive task of manual image annotation, often burdensome for research labs with limited resources.

Our strategy encompassed a two-pronged approach: initiating with a conventional thresholding technique for preliminary localization and subsequently employing a CNN-based technique for heightened precision. Given the satisfactory CR localization performance of traditional methods like thresholding, we adopted it for an initial CR center estimation, followed by centroid computations. The CNN then further refined this estimate by focusing on image patches around these preliminary sites. We tested the efficacy of our deep learning approach against established techniques such as radial symmetry and thresholding alone. We hypothesized that an optimally trained CNN would outshine these traditional methodologies in terms of localization prowess as this technique has achieved remarkable results in microscopy (*cf.*, Helgadottir, Argun, and Volpe, 2019). To generate the CRs we varied 2D Gaussian distributions placed against noise-infused backgrounds, as elaborated in 3. We use a deep learning framework known as DeepTrack (Helgadottir, Argun, and Volpe, 2019), originally developed for particle tracking in a microscopy setting. This framework utilizes a CNN trained on synthetic data to track single particles and also includes a U-Net model for tracking multiple particles. Subsequent work using this framework has also incorporated single-shot self-supervised object detection and geometric deep learning models (Midtvedt, Pineda, et al., 2022; Pineda et al., 2022). The DeepTrack 2.1 Python library (Midtvedt, Helgadottir, et al., 2021)

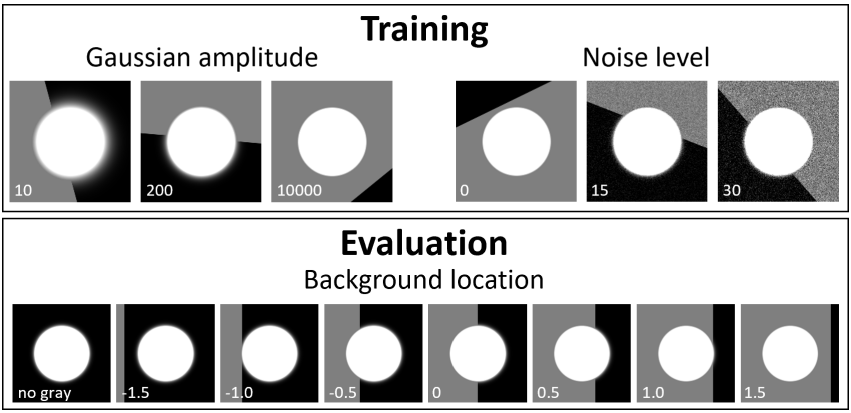


Figure 1: Example simulated CRs. Top row: example images used during model training and for the validation set. Left column: different values of Gaussian amplitude A. Right column: different pixel noise values σ_n^2 (image levels). For both columns, random positions (within $[-1.5r, 1.5r]$) and orientations of the dividing line between the dark and light sections of the background are shown. Bottom row: example images used for evaluation, showing different background locations E as well as a CR image without a gray background. The value for the varied parameter is denoted on the panels. A was set to 10000 for all panels except the top-left. For illustration purposes, the CR radius (r) in these panels is 50 pixels. During both training and evaluation the pixel intensity of the lighter section of the background was also varied (not shown).

makes it easy to generate a synthetic dataset and train a deep learning model in the same pipeline.

2.1 Evaluation Criteria

2.1.1 Evaluation Criteria for Synthetic Images

To assess the precision with which various methods can pinpoint the CR's center, we moved an input light distribution with a horizontal center x_c incrementally ($\delta_{x_c} = 0.01$) over a span of one pixel, totaling 100 steps. We subsequently contrasted the input position with the outputs from three distinct methodologies:

1. the conventional thresholding technique (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2022);
2. the radial symmetry algorithm presented by (Parthasarathy, 2012); and
3. the CNN trained on synthetic light distributions introduced in this study.

A preliminary analysis led to the exclusion of a fourth technique, which calculates the centroid of all pixels in the input image (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2022, termed the intensity-based approach in). Given the partially grey backdrop of our evaluation images, this method yielded significant errors and was deemed impractical for applications beyond the theoretical context proposed by (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2022).

The evaluation considered multiple CR radii r, Gaussian amplitudes A, gray background locations E, pixel intensity values I of the brighter background section, and noise levels σ_n. Specifically, the test set encompassed the Cartesian product of

$$r = \{2, 4, 6, \ldots, 18\},$$

$$A = \{10, 50, 200, 1000, 10000\},$$

$$\sigma_n = \{0, 2, 4, \ldots, 18\},$$

$$E = \{\text{no gray}, -1.5, -1, -0.5, 0, 0.5, 1, 1.5\},$$

and

$$I = \{38, 51, 64, 77, 89, 102, 115, 128, 140, 153\},$$

representing all possible combinations of these parameters. For each parameter combination, the horizontal center of the CR x_c traversed 100 steps with $\delta_x = 0.01$ pixels as detailed earlier. The demarcation between the two background sections remained vertical, positioned in relation to the synthetic CR. Here, $E = 0$ denotes the boundary between sections aligning with the CR center. For $E = -1$, it was situated 1 CR radius r left of the CR center, and for $E = 1$, it lay 1 CR radius to its right (refer to the bottom row of Figure 1).

2.1.2 Evaluation Criteria for Real Eye Images

How well does our approach perform on real eye images? To answer this question we tested our method against the thresholding and the radial symmetry methods when localizing the center of the CR in high resolution, high framerate videos of real eyes performing a collection of fixation tasks. Two different datasets were collected.

Dataset One *Participants*. Eye videos were recorded from three participants. Two are authors of the current paper and the third is an experienced participant in fixation tasks. None of the participants wore glasses or contact lenses. Videos were recorded from the left eye. The study was approved by the Ethical Review Board in Sweden (Dnr: 2019-01081).

Apparatus. The visual stimuli were presented on an ASUS VG248QE screen (531 x 299 mm; 1920 x 1080 pixels; 60 Hz refresh rate) at a viewing distance of 79 cm.

Videos of the subject's left eye were acquired using our FLEX setup (Hooge, Niehorster, Hessels, Cleveland, et al., 2021; Nyström, Niehorster, Andersson, Hessels, and Hooge, 2022), a self-built eye tracker. The setup included a Basler camera (Basler Ace acA2500-60um) equipped with a

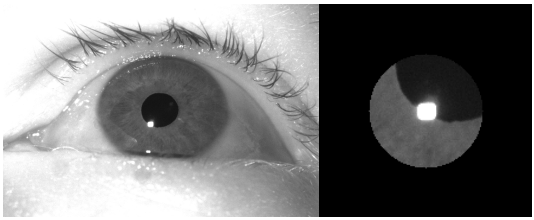


Figure 2: Full eye image (left) and masked cutout as processed by the radial symmetry and CNN methods (right).

50 mm lens (AZURE-5022ML12M) and a Near-IR Long pass Filter (MIDOPT LP715-37.5). Eye videos were recorded at 500 Hz with a resolution of 896 × 600 pixels (exposure time: 1876 µs, Gain: 10 dB) and converted into mp4 files using custom software with libavcodec (ffmpeg) 5.1.2 and the libx264 h.264 encoder (preset: veryfast, crf: 0 (lossless), pixel format: gray). Videos maintained an 8-bit luminance resolution. The EyeLink 890 nm illuminator was employed (at 75% power) to provide illumination to the eye, creating a reflection on the cornea observable in the eye image. An exemplar eye image can be found in the left panel of Figure 2.

Procedure. Participants engaged in tasks where they fixated on a blue disk (1.2° diameter), centered with a red dot (0.2° diameter):

Dataset Two A shortened protocol was employed to gather more eye footage from the left eye of 17 participants (age 30-61 yrs (average 45.4 yrs), five women, eleven men, one non-binary) who didn't wear glasses or contacts. The research received approval from the Swedish Ethical Review Board (Dnr: 2019-01081). Two participants are co-authors of this paper. One individual was omitted due to a corneal impurity resulting in an extra corneal reflection that none of the tested techniques could manage.

To better assess the resilience of our technique to changes in the brightness profile of the input eye photos, we conducted a second recording using the FLEX setup, set to capture images at a rate of 1000 Hz. At this faster rate, the obtained eye images appeared dimmer because of the

reduced possible exposure duration. The videos were captured at a resolution of 672×340 pixels with an exposure time of $882\mu s$ and a gain of 12 dB. We used the EyeLink 890 nm illuminator at full power (100%) to light the eye.

This data collection used the same displays as the previous, and the reduced protocol consisted of:

1. Nine 1-second fixations on a 3×3 grid of fixation points positioned at $h = \{-7, 0, 7\}$ deg and $v = \{-5, 0, 5\}$ deg in random order.

2. One 30-second fixation on a dot that was presented at $(0, 0)$ deg on a middle gray background.

3. Two blocks of fifteen 1.5-second fixations on a 5×3 grid of fixation points positioned at $h = \{-7, -3.5, 0, 3.5, 7\}$ deg and $v = \{-5, 0, 5\}$ deg. Fixation locations were randomly ordered within each block.

Image Analysis Image analysis was performed frame-wise. A first stage was performed using the steps described in (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2022). Briefly, an analysis ROI and fixed pupil and CR thresholds were set manually for each participant's videos to identify the pupil and CR in the images, as is commonly performed (Peréz et al., 2003; San Agustin et al., 2010; Barsingerhorn, Boonstra, and Goossens, 2018; Zimmermann et al., 2016; Ivanchenko et al., 2021; Hosp, Eivazi, et al., 2020). We ran the analyses at different CR and pupil thresholds and selected the thresholds that maximized the precision of the signals. These thresholds were used to binarize the images and after morphological operations to fill holes, the pupil and CR were selected based on shape and size criteria. The center of the pupil and CR were then computed as the center of mass of the binary blobs. The CR center provided by this method will be referred to as the CR center localized using the thresholding method.

In a second stage, a 180×180 pixel cutout centered on the center location identified by the thresholding method was made. A black circular mask with a radius of 48 pixels (about three times the horizontal size

of the CR blob) was furthermore applied to the input image (see right panel in Figure 2). These masked images were then fed into the radial symmetry and CNN methods and their indicated CR centers stored.

To assess whether our method also works on lower resolution eye images as may be delivered by other eye tracking setups, we reran the image analysis described above with all input images downsampled by a factor of 2. The processing method and parameters were identical to those for the full resolution eye videos, except that the radius of the black circular mask applied to the CNN's input images was also halved.

Data Analysis To investigate the data quality of the resulting signals, the following metrics were calculated:

First, RMS-S2S precision (Holmqvist, Nyström, and Mulvey, 2012; Niehorster, Zemblys, Beelders, et al., 2020; Niehorster, Santini, et al., 2020) of the CR center signals estimated using the three methods was computed in camera pixels for all the collected gaze data using a moving 200-ms window, after which for each trial the median RMS from all these windows was taken (Niehorster, Hessels, and Benjamins, 2020; Hooge, Niehorster, Nyström, et al., 2018; Hooge, Niehorster, Hessels, Benjamins, et al., 2022). The same calculation was performed for the pupil center signal. RMS-S2S precision of the calibrated gaze signal computed based on the three CR center signals were estimated. The RMS-S2S precision, used in gaze position analysis, is defined as:

$$\text{RMS-S2S} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n-1}((x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2)} \tag{2.1}$$

where n denotes the number of gaze position samples, x and y represent the horizontal and vertical components of the gaze position, respectively, and the summation is over the squared distances between successive gaze positions. This metric is a measure of the average velocity of eye movements during fixations, assuming a constant sampling frequency(Niehorster, Zemblys, Beelders, et al., 2020). In this case, gaze location was determined using standard P–CR methods: after subtracting

the CR center location from the pupil center location, the resulting P–CR gaze data were calibrated using the gaze data collected on the 3×3 grid of the first task. Calibration was performed with second-order polynomials in x and y including first-order interactions (Stampe, 1993; Cerrolaza et al., 2012):

$$p_{gaze} = a + bx + cy + dx^2 + ey^2 + fxy, \tag{2.2}$$

where p_{gaze} is the gaze position in degrees. The same formula was applied to compute the horizontal and vertical gaze positions. The accuracy of the gaze signal was then computed for each trial as the offset between the median estimated gaze location and the fixation point location for the data of task 4 in dataset one and task 3 in dataset two. The accuracy values for the repeated fixations on the 15 fixation targets were averaged. Similarly to the CR and pupil center signals, RMS-S2S and also STD precision of the gaze signal was computed in moving 200-ms windows for all the collected gaze data, after which for each trial the median RMS or STD value from all these windows was taken.

2.2 Results

2.2.1 Optimal CR center Localization Performance in Synthetic Images

Figure 3 shows the best obtainable CR center localization performance based on the information in the synthetic CR images for the different Gaussian amplitudes (different panels) and CR sizes (lines within each panel). As can be seen, appreciable errors in CR center location occur at all examined Gaussian amplitudes for the smallest CR size (2), and also for CR size 4 for higher Gaussian amplitudes (i.e. images containing narrower tails). Furthermore, error increases as a function of Gaussian amplitude (narrower tails). To illustrate how close the different CR center localization methods are to their optimal performance, the results of this examination will be used as reference lines when presenting the evaluation on synthetic images in the "Evaluation on Synthetic CRs" section below.

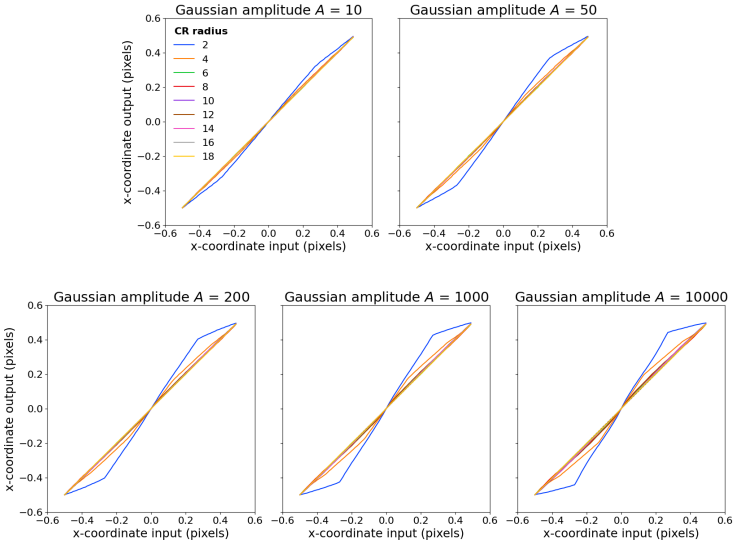


Figure 3: Best achievable CR center localization errors for different Gaussian amplitudes A (different panels) and CR radii r (different lines in each panel).

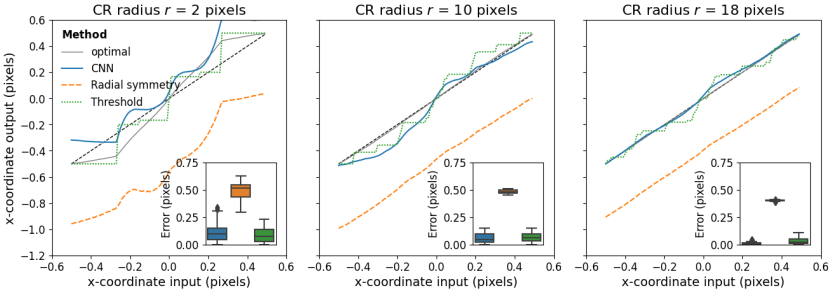


Figure 4: Errors in CR center localization for different CR sizes r for three methods. The panel insets show boxplots of the CR center localization error for each estimated input position. For all these simulations, $A = 10000$, $E = 0$, $I = 128$

2.3 Evaluation on Synthetic CRs

First, we sought to investigate whether our method could determine the CR center more accurately than two commonly utilized algorithmic strategies when applied to synthetic data. Figure 4 displays the error in CR center localization achieved by the three methods for three distinct CR sizes. Negative errors are to the left, and positive to the right. For illustration, results are presented with a Gaussian amplitude $A = 10000$ and a half-grey background ($E = 0$, $I = 128$). It can be observed that, for the smallest CR size, the CNN and thresholding methods perform similarly, while the radial symmetry method exhibits a greater bias towards the image's left, which is the gray side. As the CR size enlarges, this bias towards the image's grey side only marginally diminishes for the radial symmetry method. For these larger CR sizes, the center localization output from the threshold and CNN methods becomes more stable, and the CNN generally exhibits a lower error compared to the threshold method.

Localization performance of the three methods as a function of CR size for three different pixel noise levels is depicted in Figure 5 (top-left panel). Noticeably, the error in localization is nearly independent of CR size for the threshold and CNN methods, while the error decreases as a

function of CR size for the radial symmetry method. The thresholding and CNN methods remain unaffected by noise in the input image within the examined range and attain a comparable CR center localization error. Conversely, the radial symmetry method was profoundly influenced by pixel noise level (errors were predominantly over 0.5 pixels at noise level 8 and varied between 3 and 8 pixels at noise level 18, not shown). Therefore, subsequent plots showcase results at noise level 0 to underscore the optimum possible performance of the radial symmetry method.

The impact of the pixel intensity level of the lighter background section is illustrated in Figure 5 (top-right panel). Analogous to the effect of pixel noise level, the localization error of the threshold and CNN methods, but not the radial symmetry method, is nearly independent of CR size and background pixel intensity level. Moreover, the performance of the CNN method closely mirrors that of the threshold method, both attaining errors of around to well below 0.1 pixels across CR sizes.

The influence of the location of the grey background on localization performance is presented in Figure 5 (bottom-left panel). While all methods performed nearly flawlessly when the background was entirely black, only the threshold and CNN methods exhibit stability over different locations of the gray background. The radial symmetry method's performance is significantly impacted by the position of the gray background, displaying considerably larger errors than when no gray background existed.

The impact of the width of the tail of the CR is demonstrated in Figure 5 (bottom-right panel). Recall that more saturated Gaussians (those with larger amplitude A) possess narrower tails (c.f. Figure 1). Observably, the effect of tail width on CR localization performance is minimal for all three methods. Collectively, it is crucial to note that the localization performance of the CNN is equivalent to the best-performing algorithmic approach to CR localization, achieving average errors of around or well below 0.1 pixels in all instances.

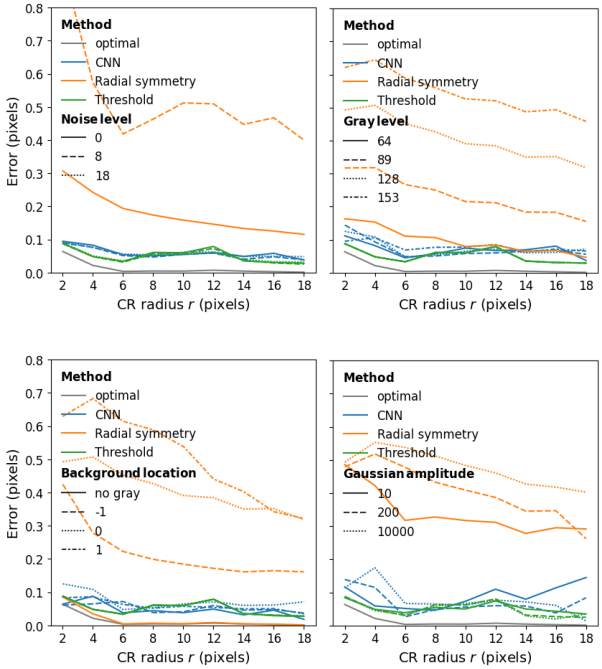


Figure 5: Errors in CR center localization for the three methods as a function of CR radius, for different noise levels (top-left), pixel intensity levels of the lighter background section (top-right), locations of the gray background (bottom-left) and Gaussian amplitudes (bottom-right). For the top-left panel, average error of the radial symmetry method ranged from 3–8 pixels at noise level 18, not shown. For the top-right and bottom panels, the noise level was 0. For the bottom panels, the pixel intensity level of the lighter background section I was 128. For the bottom-right panel, the background location E was 0.

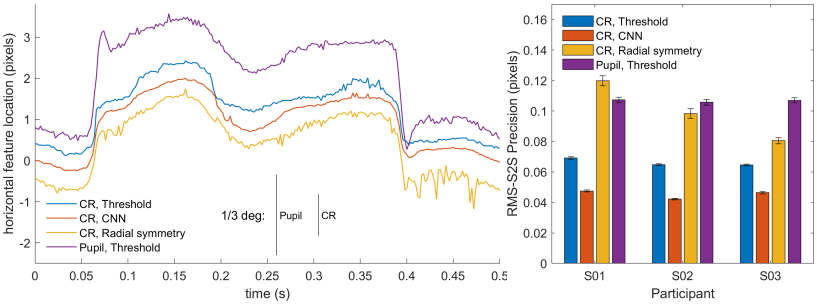


Figure 6: Real eye CR and pupil center signals of dataset one: Left: representative segment of pupil and CR center signals from Subject number three (S03) in camera pixels. For the CR center, the signals produced by three different CR center localization methods are shown. The signals contain two small saccades and have been vertically offset for clarity. RMS precision for the shown segments are 0.081 pixels for the Threshold signal, 0.061 pixels for CNN, 0.096 pixels for Radial symmetry, and 0.121 pixels for the pupil signal. Further, an RMS precision comparison (right panel) between the three methods and the pupil signal on all data of three participants is shown. Error bars depict standard error of the mean.

2.4 Evaluation on Real Eye Images.

2.4.1 Dataset One

Next, we set out to test whether our method is able to perform CR center localization in real eye images and if so, whether it delivers a CR position signal with higher precision than two algorithmic approaches. To test how well our method works on real eye images, we first performed CR and pupil center localization on dataset one, which consisted of 500Hz videos of eye movements made by three participants. CR localization was performed by three methods.

The left panel of Figure 6 shows an example segment of CR center locations estimated using the three methods, along with the estimated pupil center location. A can be seen, the CR center signal from the CNN method appears smoother than the signal from the threshold method, while the signal from the radial symmetry method looks less smooth than the threshold signal. The pupil center signal by and large looks similarly noisy as the CR signal from the radial symmetry method.

To quantify these observations, we calculated the RMS precision of all four signals for all recorded videos of three participants. The results of this analysis are shown in Figure 6 (right panel). While there were differences in overall noise level between participants, a clear pattern in results for the CR center localization methods is seen. The CNN method consistently delivers signals with a better precision (lower values) than the thresholding method, while the radial symmetry method delivers signals with worse precision (higher values). Precision of the pupil center signal is consistently much worse than that of the CNN- or thresholding-based CR center signals. It is important to note here that all methods processed each video frame independently, and that improved precision could thus not be due to any form of temporal information being used from previous or future frames (c.f. Niehorster, Zemblys, and Holmqvist, 2021a; Niehorster, Zemblys, Beelders, et al., 2020).

How does the improved CR center localization of our method impact the gaze signal? To answer this question, we performed a similar analysis as above, but on the calibrated gaze signals. The top panel of Figure

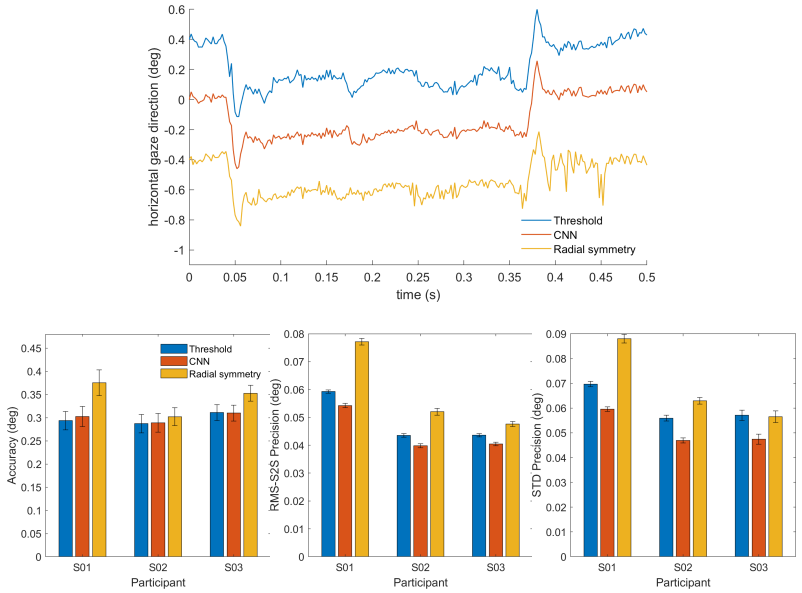


Figure 7: Real eye calibrated gaze signals of dataset one. Top: representative segment of calibrated P-CR signals from S03 as processed by three different CR center localization methods. The signals contain two small saccades and have been vertically offset for clarity. RMS precision for the shown segments are $0.040°$ for the Threshold signal, $0.034°$ for CNN, and $0.046°$ for Radial symmetry. Further, an accuracy comparison (bottom left panel), an RMS precision comparison (bottom middle panel) and an STD precision comparison (bottom right panel) between the three methods on data of three participants are shown. Error bars depict standard error of the mean.

7 shows an example segment of gaze data computed from the three signals. As can be seen, the gaze signals derived from the three different CR localization methods look much more similar than the CR center signals in Figure 6 (left panel). This is likely due to the fact that derivation of the gaze signal involves subtracting the estimated CR center location from the much noisier pupil center location estimate (c.f. Niehorster and Nyström, 2018). The noise in the pupil center location estimate likely is the dominant component of noise in the derived gaze signal, to a large extent swamping the differences in precision between the CR center location signals.

The bottom panels of Figure 7 show the accuracy, RMS-S2S, and STD precision achieved with the three CR center localization methods. While there were small differences between the participants, no systematic differences in accuracy between the three methods were observed. Overall, both the RMS-S2S and the STD precision of the gaze signals derived from the CR center localization estimates of the CNN was a little lower than for the gaze signal derived from the threshold-based CR center, while that for the gaze signal derived from the radial symmetry method for CR center localization showed worse precision.

2.4.2 Dataset Two

To examine how our method performs on real eye images across a wider range of participants with different eye physiology and for lower resolution eye images, we have collected a new set of 17 participants (one of which was excluded from analysis, see methods) and analyzed the videos captured both at full and at half resolution.

Figure 8 shows the calculated RMS precision of the CR signal processed using three different methods, and the pupil signal, for all recorded videos of all participants. The same pattern emerges for the full resolution and the half resolution videos. As we saw for dataset one, while there were differences in overall noise level between participants, a clear pattern of results emerges where the CNN method delivers signals with a better precision (lower values) than the thresholding method, and the

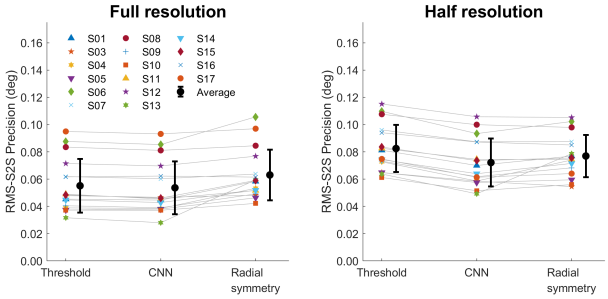


Figure 8: RMS-S2S precision of the raw signals for dataset two. An RMS precision comparison between the CR center signals derived from the three methods and the pupil center signal is shown for all participants (colored symbols) along with the mean across participants (black circles) for analyses run both at full video resolution (left panel) and at half resolution (right panel). Error bars depict standard error of the mean.

radial symmetry method performs worse (higher values) than the CNN method. As before, the precision of the pupil center signal is worse than that of the CNN- or thresholding-based CR center signals.

To examine how the CR center localization methods impact the resulting calibrated gaze signals, we computed the RMS-S2S and STD precision, and the accuracy of the calibrated gaze signals of each participant for both the full resolution and half resolution video analyses. As for dataset one, in most cases there were only small differences in RMS-S2S and STD precision (Figure 9, top and middle rows) between the three CR center localization methods for both video resolutions, with the CNN method showing slightly better precision (lower values) than the other methods. Only for the half-resolution analysis was the STD precision of the gaze signal derived from the threshold method clearly worse (higher values) than for the signals derived from the CNN and radial symmetry methods. Accuracy did not vary systematically between the three methods.

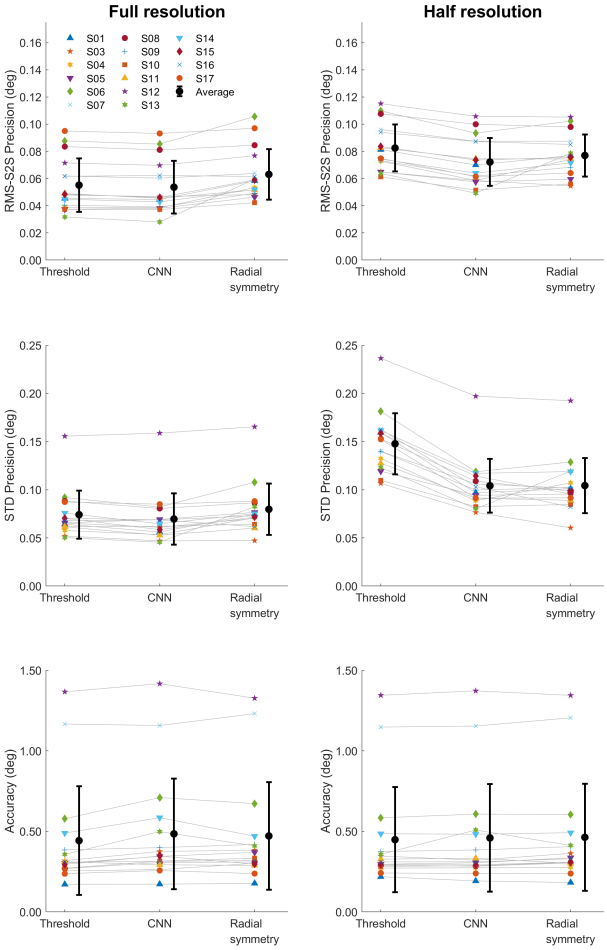


Figure 9: Data quality of calibrated gaze signals of dataset two. RMS-S2S precision (top panels), STD precision (middle panels) and accuracy (bottom panels) comparisons of the calibrated gaze signals derived from the three CR center localization methods is shown for all participants (colored symbols) along with the mean across participants (black circles) for analyses run both at full video resolution (left panels) and at half resolution (right panels). Error bars depict standard error of the mean.

2.5 Concluding Remarks

In this chapter, we have presented a CNN architecture and training procedure for localizing single CRs in eye images. Additionally, we have scrutinized the spatial accuracy and precision achieved with this novel method using both synthetic and real eye images. In relation to the research queries outlined in the introduction, the primary contributions of this paper are as follows:

1. We propose a straightforward method, utilizing only synthetic images, to train a CNN for CR center localization. Furthermore, we showcase that a CNN trained through this method can effectively perform CR center localization in real eye images.

2. We illustrate that our approach can pinpoint the CR center with a degree of accuracy comparable to a routinely employed algorithmic strategy when tested on synthetic data.

3. We demonstrate that, in terms of spatial precision, our method surpasses algorithmic strategies for CR center localization when employed on real eye images.

Through our experiments we have shown that our CNN-based method consistently outperforms the popular thresholding method for CR center localization as well as the radial symmetry method that was recently adopted by R.-J. Wu et al. (2022). As Nyström, Diederick C Niehorster, Andersson, Hessels, and I. T. C. Hooge (2022) have recently shown, binarizing an eye image using a thresholding operation reduces the CR center localization accuracy compared to methods that use the full range of pixel intensity values in the image of the CR c.f. also Helgadottir, Argun, and Volpe, 2019, who show this in the context of microscopy. The radial symmetry method Parthasarathy, 2012 uses the full range of intensity values and has been shown to outperform thresholding for localization of the center of image features R.-J. Wu et al., 2022; Helgadottir, Argun, and Volpe, 2019; B. Midtvedt, Pineda, et al., 2022. However, these results were obtained with features displayed on a uniform background. Our

simulations show that the radial symmetry method is consistently considerably worse when used on images with a background consisting of two regions with different luminance.

It is therefore not suitable for use in more general eye tracking scenarios, where the CR is often overlaid on a non-uniform background, such as the iris or the edge of the pupil. In contrast, our CNN was trained on highly simplified images inclusive of such backgrounds, and it exhibits robust performance against their presence in both synthetic and real eye images. This underscores that, when properly trained, the CNN approach can utilize the pixel intensity data inherent in the CR image to pinpoint its center, effectively sidelining the background. Our CNN methodology consistently surpassed the alternative methods across assessments conducted on two distinct datasets. Furthermore, it maintained its superiority even when provided with eye images that had been downsampled to half their original resolution. This attests to the method's versatility across diverse participants with varied eye physiology and its adaptability to lower resolution eye imagery.

Our results show that while our method offered significantly reduced RMS precision in the CR center signal (28.0% to 34.9% lower than thresholding for dataset one, and on average 13.0% and 41.5% lower for the full and half resolution analyses of dataset two, respectively), this translated to an improvement in RMS precision of the gaze signal that ranged only between 7.2% to 8.6% for dataset one, and on average 2.9% for the full resolution and 13.0% for the half resolution analysis of dataset two. Indeed, for a gaze signal that is derived using the P-CR principle, CR center localization performance is only half the story. P-CR eye trackers typically use the vector between the pupil and CR centers and as such noise in the pupil signal also plays an important role in determining the precision of the gaze signal. As shown in our results, for dataset one the noise in the pupil signal was between 55% to 66% higher than in the CR center signal based on thresholding (average 127% at full and 32.1% at half resolution for dataset two). This ratio only worsens to between 126% to 151% when considering the CR center signal produced by our CNN method for dataset one (average 165% at full and 130% at half resolution

for dataset two). As such, further improvements in CR center localization precision will be of little practical use for P-CR eye trackers until the precision of pupil center localization is also improved.

This paper has demonstrated that simple simulations can be used to effectively train deep learning models that work on real eye images, raising questions about the need for heavy data augmentation techniques and time-consuming data collection as well as hand labeling efforts or reconstruction methods. However, it is worth emphasizing that we have so far employed this approach only on high quality eye images (see Figure 2) encountered in high-end lab-based eye tracking scenarios where researchers are interested in microsaccades and other fixational eye movements, as well as other aspects of eye movements that require high data quality, such as slow pursuit.

While our approach shows promise for these research scenarios, other scenarios in which eye trackers are frequently applied such as virtual reality or wearable eye tracker settings face eye images of significantly worse quality. Our approach should thus be tested on more challenging targets (e.g. localizing the center of the pupil or iris), more complex situations (e.g. involving multiple CRs and spurious reflections) and images of lower quality to further test the hypothesis that effective gaze estimation methods in a broader context can be trained using simple simulations alone.

In summary, our results indicate that our method for training deep learning models for eye tracking applications using only simple synthetic images shows great promise. Which we further explore with full P-CR pipelines in the following chapter.

Chapter 3

The LEyes Framework

[1].

Following the promising outcomes presented in the previous chapter where a single corneal reflection (CR) was synthesized through modeling, we shift our focus to the analogous approach of generating synthetic images. These images represent the 2D light distributions captured by a video-based eye-tracking system, intended for training neural network models. In this chapter, we delve into creating comprehensive P-CR pipelines and subject our methodology to the intricate task of pupil localization in virtual reality (VR). In this chapter, we champion a pioneering strategy that deviates from the conventional approach of crafting photorealistic images. We choose, instead, to leverage the inherent simplicity of eye images. Eschewing the painstaking reproduction of every visual nuance, our emphasis is on simulating the light distributions pivotal to the key features within an eye image, essential for eye tracking. Our findings suggest that images generated through the LEyes technique are not only straightforward to produce but are also rapid in generation. Crucially, our LEyes-based method yields results with superior accuracy compared to other synthetic data methodologies across a spectrum of eye

[1]This discussion is grounded in a co-authored manuscript Sean Anthony Byrne, Virmarie Maquiling, Marcus Nyström, et al. (2023). *LEyes: A Lightweight Framework for Deep Learning-Based Eye Tracking using Synthetic Eye Images*. arXiv: `2309.06129 [cs.CV]` which is currently a pre-print on Arxiv, where the author of this dissertation is the first author.

tracker configurations. We have christened this data creation technique as "Light Eyes", abbreviated as LEyes

The deployment of deep learning algorithms has significantly enhanced the accuracy and robustness of gaze estimation techniques, as evidenced by multiple studies (Fuhl, Santini, Kasneci, and Kasneci, 2016; Fuhl, Santini, Kasneci, Rosenstiel, et al., 2017; Fuhl, Weber, and Eivazi, 2023; Maquiling, Byrne, Nyström, et al., 2023; Kothari et al., 2022b; Kim et al., 2019; Nair et al., 2020). Deep learning algorithms address issues present in conventional algorithmic approaches, which are vulnerable to unpredictable factors like blinks or reflections in the recording (Kothari et al., 2022a). Yet despite these benefits, the incorporation of deep learning algorithms continues to pose challenges, principally due to the complex task of gathering data for training the model (Garbin et al., 2020; Palmero et al., 2021). This data procurement obstacle in gaze estimation can be detailed as follows:

1. **Data scarcity:** While data scarcity is a common issue across many deep learning domains (Bansal, Sharma, and Kathuria, 2022), this challenge is particularly acute in the field of eye-tracking research. Collecting a sufficient amount of training data for the development of deep learning models in this area demands significant time and resources (Byrne, Nyström, et al., 2023; Garbin et al., 2020).

2. **Annotated datasets:** The second challenge involves the necessity for annotating segmented regions within eye images. This annotation is essential for creating labels for supervised learning algorithms that train deep learning models. It is a process that not only is time-consuming but also technically demanding, often requiring manual labeling by an experienced researcher (Garbin et al., 2020; Palmero et al., 2021).

3. **Differences in recorded eye images:** The third challenge stems from disparities in eye images found in the limited amount of publicly available datasets. Differences can occur not just across recording setups, but also from variation in eye image attributes like iris brightness, which lead to pixel level differences that contribute to

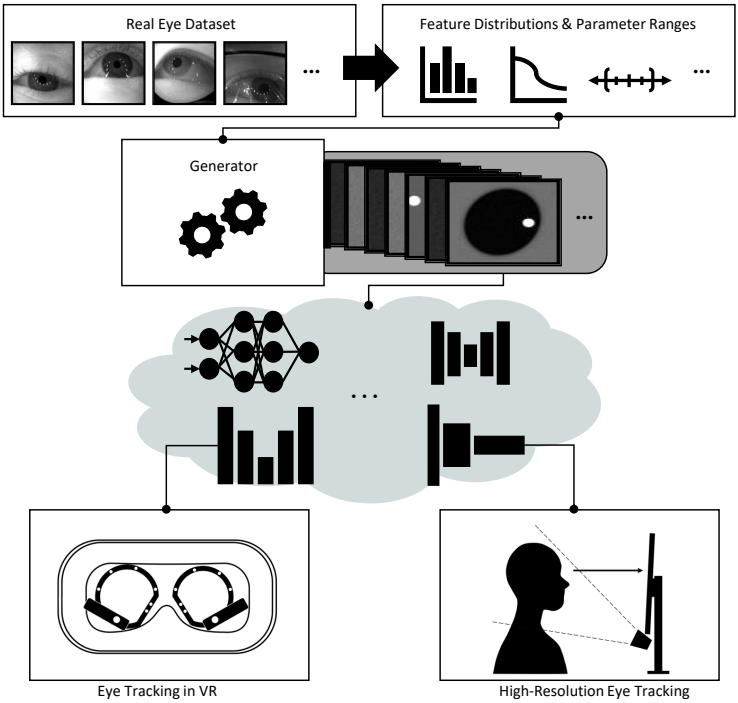


Figure 10: A. Images from the four datasets we used to test the LEyes framework. **B.** The LEyes synthetic training sets corresponding to the real eye datasets in A. These images are based on the light distributions of the real eye datasets. **C.** This shows the predictions of the LEyes trained model on the real eye images. **D.** An overview of our approach: First, we establish a set of parameters based on the distributions of the collected data. These distributions pertain to pixel-level details like the iris and pupil intensity. Next, we employ a generator to efficiently produce new synthetic images from these parameters. The generated images are used to train a neural network which is then tested on real eye images recorded from the same device.

sub-optimal network performance (Nair et al., 2020). This is a major issue as slight differences can have a substantial impact model performance and generalizability.

A proposed solution to these challenges is the use of synthetic datasets which allows for the generation of vast amounts of annotated images (Kim et al., 2019; Byrne, Nyström, et al., 2023; Maquiling, Byrne, Nyström, et al., 2023; Nair et al., 2020). Synthetic data has been used successfully to train deep neural networks in fields such as medical imaging (Gao et al., 2023), autonomous driving (Osiński et al., 2020), and microscopy (Helgadottir, Argun, and Volpe, 2019). Typically in the field of gaze estimation, synthetic eye images creation methods aims for photorealism by employing a 3D model of the human eye and surrounding facial region to produce 2D images akin to those captured by eye-trackers, using render software or game engines such as Blender or Unity. The goal of such processes is to match the synthetic dataset's underlying distribution with the variability seen in real-world eye images (Wood et al., 2015a; Nair et al., 2020). The photorealistic synthetic data approach, however, is not without limitations. One key challenge is the complexity of generating synthetic datasets that accurately emulate the distribution of real eye images. Additionally, concerns exist regarding the potential for achieving state-of-the-art outcomes when compared to models trained on genuine eye images. A study illustrated a decline in model accuracy by 1° when comparing a model trained on photorealistic synthetic images to one trained on a subset of real eye images using a neural network (Kim et al., 2019; Nair et al., 2020). We hypothesize that numerous intricate features must be precisely constructed during synthetic dataset creation, and even minor deviations in design can significantly impact a model's inference capabilities during testing.

3.0.1 Overview of the LEyes Framework

Previous research (Byrne, Nyström, et al., 2023; Nyström, Niehorster, Andersson, Hessels, and Hooge, 2023; Maquiling, Byrne, Nyström, et al., 2023) has shown that key features in eye images relevant for eye tracking

can be effectively represented using 2D Gaussian distributions. Creating a synthetic dataset of eye images necessitates the accurate portrayal of such features, including the pupil, reflections, and pixel-level characteristics such as iris brightness (Kim et al., 2019; Nair et al., 2020; Kothari et al., 2022b), which can be affected by specific lighting conditions altering dimensions and luminosity of features located in the image such as the iris or pupil. Emulating essential hardware attributes, such as lighting conditions and camera parameters, is vital for replicating real-world situations (Kim et al., 2019; Nair et al., 2020; Kothari et al., 2022b). LEyes shows that by generating abstract images using 2D Gaussian distributions that contain the relevant features for an eye tracker, one can effectively capture both eye features and camera attributes for neural network training. The approach is outlined as follows:

First, to model key features such as the pupil, iris, and CRs, luminance attributes are derived by calculating the distributions of recorded data on a given device setup. To ensure generalizability of the model for a wide range of participants we use a larger parameter range than is derived from the distributions. Subsequently, these parameter ranges calculated from the distributions are utilized to craft images by layering and combining the parameter inputs through mathematical operations, achieving simple but realistic portrayals of eye features and noise within the created image. The images are then scaled and discretized to align with standard 8-bit camera output. Refer to the Synthetic Data Methods & Models (3) section for a complete description of the process of generating LEyes images along with full descriptions of each model architecture used in the paper.

To turn the feature parameters into images to be used to train the deep learning models we again utilize the generator function from the DeepTrack 2.1 package (Midtvedt, Helgadottir, et al., 2021). The use of a generator combined with the relatively simple images created from 2D Gaussian distributions enables swift creation of customized synthetic images at reduced computational cost compared to photorealistic models. For example, the NVGaze dataset required 30 seconds to create each image; to create the entire dataset, would take approximately 3.8-years on

a single GPU. In practice, this was reduced to a week as the researchers had access to a supercomputer (Kim et al., 2019). Our models require no special computational resources and can be trained on platforms such as Google Colab, making them accessible to a wider group of researchers. The generator function also keeps track of both the image and corresponding label during training, which allows the generator to discard images after one pass to prevent over-training. Importantly, no images need to be pre-generated and occupy disk-space when a generator function is used (Helgadottir, Argun, and Volpe, 2019).

3.1 Results

3.1.1 Pupil Localization

We begin our analysis by considering the performance of the LEyes framework in a pupil center localization task in a VR setting, a common task for video based eye trackers (Kim et al., 2019). To test our model we selected the widely used 2019 EDS challenge dataset (OpenEDS 2019) (Garbin et al., 2020). We chose this dataset to run a comparative analysis as it has been used extensively to assess the accuracy of other methods in gaze estimation tasks (Nair et al., 2020; Chaudhary et al., 2022; Kim, Lee, Yang, et al., 2019; Kothari et al., 2022b; Kothari et al., 2022a).

OpenEDS 2019 (Garbin et al., 2020) was collected using a VR head-mounted display equipped with dual eye-facing cameras, capturing images at 200 Hz under controlled lighting conditions. The dataset encompasses eye-region video footage from 152 participants for a total of 12,759 images featuring pixel-level annotations derived from human-annotated key points of the iris, pupil, and sclera. For a complete description of the data, refer to the original paper (Garbin et al., 2020).

Various deep learning architectures have been proposed for eye segmentation tasks and LEyes simulations are model agnostic, yet, in light of their prevalent use and proven efficacy in eye tracking tasks (Chugh et al., 2021; Wang, Wang, and Guo, 2023), we chose to train a U-Net model with a ResNet-34 backbone. The model takes a grayscale eye image as

its input and outputs a probability map indicating the location of the pupil in the image. To determine the center of the pupil we threshold this mask and employ a center of mass algorithm on the pupil region in the resulting binary image.

We compare our results with other state-of-the-art models and frameworks, including Pistol (Fuhl, Weber, and Eivazi, 2023), PuRe (Santini, Fuhl, and Kasneci, 2018), the EllSeg framework (Kothari et al., 2022a; Kothari et al., 2022b), and DeepVog (Yiu et al., 2019). These models employ a variety of methods, ranging from conventional ellipse fitting to deep learning architectures. Note that we stress the difference between model and framework where a "model" is a specific representation trained to make predictions, while a "framework" is a set of tools and libraries used to develop, train, and deploy such models.

Estimation accuracy in eye-tracking applications is often evaluated using the cumulative detection rate, which shows how much of the pupil locations estimated by a method are within a given distance from the ground truth pupil center (Kim et al., 2019; Kothari et al., 2022a; Fuhl, Santini, Kasneci, Rosenstiel, et al., 2017; Santini, Fuhl, and Kasneci, 2018). Performance is often specifically assessed as the percentage of images for which the pupil location was estimated within a 5-pixel distance from the ground truth (Kim et al., 2019; Fuhl, Santini, Kasneci, Rosenstiel, et al., 2017). However, recent algorithms have demonstrated superior performance on VR datasets, often reaching ceiling performance well below this 5-pixel threshold. Consequently, we have narrowed our analysis to examine performance for errors up to just 2 pixels. As illustrated in Figure 11, we achieved a 2-pixel error rate of 75.8%, which surpasses EllSeg (model trained on all datasets) at 71.8% and is markedly superior to Pure (65.6%), DeepVOG (60.9%), and Pistol (55.4%). The violin plots in the bottom section of Figure 11 indicate that the distribution of performance across participants in the testing set at the 2-pixel level for a model trained on LEyes is comparable to other models. Notably, its median value at this error level is 80%, outperforming the next best model by 7%.

We underscore comparisons with the different variants of the EllSeg

framework (Kothari et al., 2022a; Kothari et al., 2022b), one of the few public frameworks leveraging synthetic training data, like ours. The architecture used in the the EllSeg framework, named DenseElNet (Kothari et al., 2022b), has a comparable number of trainable parameters (2.24 million) compared to our model. The orange line in Figure 11(B) shows the EllSeg variant trained across multiple eye datasets. Notably, OpenEDS 2019 is one of the datasets included in its training set. Astonishingly, our LEyes model still surpasses this variant, even though there is evident data leakage with 88.6% of the training samples present in the test set. Two further comparisons of EllSeg models trained on purely synthetic datasets (RITeyes (Nair et al., 2020) and NVGaze (Kim et al., 2019)) show that the LEyes framework consistently outperforms other publicly available models that use only synthetic data.

Taken together, the results highlight that LEyes achieves higher performance against other methods tested on the EDS 2019 dataset. In line with earlier observations on domain discrepancies and generalization in gaze estimation (Kothari et al., 2022b; Kothari et al., 2022a; Nair et al., 2020), our results demonstrate that models exhibit optimal performance when trained on datasets analogous to their respective test distributions, something that is easily achieved within the LEyes framework. Notably, this efficacy persists even when there are discernible differences, from a human perspective, between the training and test datasets.

3.1.2 Simultaneous Pupil and Corneal Reflection Localization

Pupil and CR localization can be treated as two separate problems, yet since their positions in the eye images co-vary in a systematic way, it is advantageous to consider them together. In this section, we present a new P-CR eye tracking pipeline, trained entirely using LEyes images. Eye trackers often use several light sources to guarantee at least a pair of reflections for every gaze position, and for robust eye tracking it is necessary to reliably associate at least two corneal reflections with their specific light source across all anticipated eye movements (Chugh et al.,

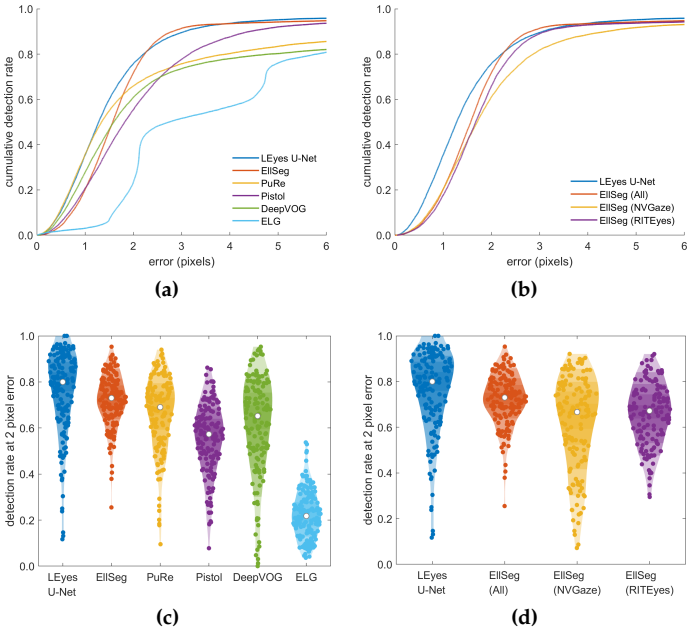


Figure 11: A. We compare the cumulative detection rate on the OpenEDS 2019 dataset of a U-Net model trained using the LEyes method at different pixel errors against PuRe (Santini, Fuhl, and Kasneci, 2018), Pistol (Fuhl, Weber, and Eivazi, 2023), DeepVOG (Yiu et al., 2019), ELG (Park et al., 2018). **B.** We make special comparisons with several models trained using the EllSeg Framework (Kothari et al., 2020; Kothari et al., 2022b). **C & D:** The corresponding violin plots for panels A and B respectively, showing the detection rate at 2 pixel error for each participant in the testing set achieved by LEyes compared with the aforementioned models.

2021). Therefore, our pipeline is not only able to localize the pupil and CR centers in an eye image, but also match the CRs to specific light sources. While previous work has developed models that locate the pupil and CRs simultaneously and perform CR matching (Niu et al., 2021; Maquiling, Byrne, Nyström, et al., 2023), we introduce a novel method that importantly streamlines the process of robust P-CR eye tracking by using the maximum value of the model output to select only the 'best' two CRs. This ability of our method to robustly select CRs for gaze estimation is especially important due to the complex reality often encountered in eye images where CRs may be missing or additional, unwanted, reflections are often present.

Through our novel pipeline, illustrated in Figure 12, we aim to demonstrate the power of LEyes in such challenging scenarios. First, since LEyes requires input images of a certain size that contain the pupil, we have developed a novel adaptive cropping strategy. Second, we demonstrate the success of our strategy to select the two 'best' CRs in an eye image.

We test our new P-CR pipeline using two VR datasets. First, we use a dataset compiled by Chugh, et al. (2021) (Chugh et al., 2021) which contains eye images of 15 participants captured from a VR headset with an eye tracking attachment. The dataset includes manually annotated (x, y) coordinates for the pupil-center and centers of the CRs. Second, we test on the OpenEDS 2020 Challenge dataset (OpenEDS 2020) (Palmero et al., 2021), which consists of 80 participants and includes manually annotated segmentation labels for the pupil for 5% of the data, amounting to a total of 2605 images. The images were captured at a frame rate of 100 Hz under controlled illumination using a VR headset. Since only pupil, but no CR annotations, are provided with the Open EDS 2020 dataset (Palmero et al., 2021), we provide illustrative examples instead of a quantitative comparison. Through these examples, we want to highlight that our model appears to provide accurate predictions also for CRs in this dataset, despite have more CRs (eight instead of five) with a different spatial configuration compared to the Chugh, et al. (2021) dataset. Figure 13 (bottom) illustrates how our model performs on a representa-

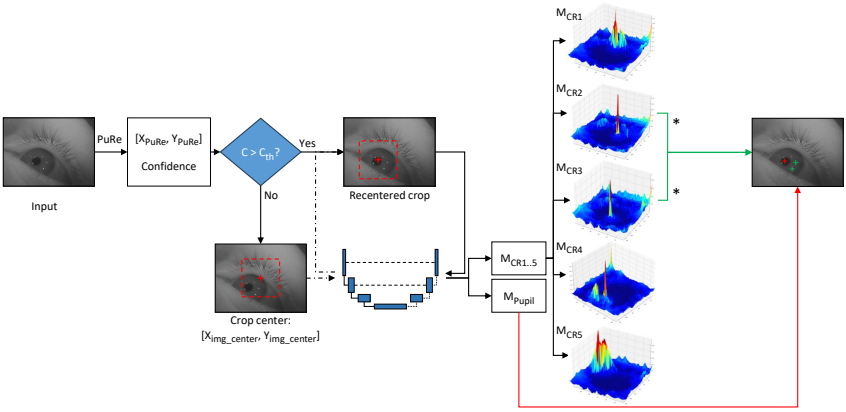


Figure 12: Flowchart of the simultaneous P-CR pipeline: Using an adaptive cropping strategy the center of the crop is determined using PuRe's pupil center prediction ($[X_{PuRe}, Y_{PuRe}]$) if the confidence metric for PuRe's prediction (C) is above a given confidence threshold (C_{th}), otherwise, the crop is determined by the pupil prediction of the LEyes-trained model given a naive center crop ($[X_{img\_center}, Y_{img\_center}]$). The pupil-centered crop is passed through the model, which outputs logits representing likely feature locations for each prediction, illustrated here as heat maps (M) for both the pupil (M_{Pupil}) and for each CR ($M_{CR1\ldots5}$ in this example). For each CR map, the highest value is located. These peaks are compared between maps and the two highest values across all the maps determine which CRs are selected. The asterisks signify which maps contain the two highest values in this example. However, if the exclusion criteria are met, the image is deemed invalid (see text).

tive selection of eye images from the OpenEDS 2020 dataset. Predictions from all eye images are available in the repository associated with this paper.

Adaptive Cropping Strategy

Before inputting the eye image into the model, it needs to be cropped to the input size expected by the model in such a way that the pupil is in the crop. A naive cropping strategy assumes that the pupil is in the center of the eye image. However, this is not always the case and such a crop may exclude parts of, or even the entire pupil from the crop. To solve this challenge, we employ PuRe (Santini, Fuhl, and Kasneci, 2018), a well known lightweight open-source pupil detection method based on ellipse fitting, to create a 128×128 pixel image centered on its detected pupil center. We ran PuRe over the two datasets and found that PuRe had average pixel errors of 27.39 and 20.77 in the OpenEDS 2020 and Chugh et al. 2021 datasets, respectively. Next, providing these crops based on PuRe's pupil center estimate to LEyes still yielded high average errors of 10.25 and 5.27 pixels in those datasets due to cases where PuRe failed to locate the pupil. Therefore, we adopted an adaptive cropping strategy using PuRe's confidence metric. This confidence, ranging between 0 and 1, is based on various metrics outlined in detail in the paper (Santini, Fuhl, and Kasneci, 2018), with 0 indicating a poor ellipse outline. In our cropping method, if PuRe's confidence is larger than or equal to a threshold, the crop used as input to the LEyes U-Net is based on PuRe's pupil center estimate. If the confidence is below this threshold, we instead use the naive center crop on the image with no guarantee that the pupil will be present. This hybrid cropping strategy significantly improves our model accuracy. For the OpenEDS 2020 dataset the lowest average pupil error was 2.32 pixels, achieved when using a confidence threshold of 0.90 (the largest average pixel error was 2.58 for confidence thresholds between 0.50–0.95). For the Chugh et al. 2021 dataset, an average pixel error of 2.5 pixels was achieved at the 0.70 confidence threshold (largest error 3.56).

Table 1: Mean Pupil Pixel Error at Different PuRe Confidence Thresholds

PuRe Confidence Level	OpenEDS 2020 Avg. Pixel Error	Chugh et al. 2021 Avg. Pixel Error
0.55	2.58	2.81
0.6	2.57	2.74
0.65	2.58	2.62
0.7	2.52	2.50
0.75	2.46	2.65
0.8	2.42	2.71
0.85	2.38	2.72
0.9	2.32	2.81
0.95	2.44	3.26
0.99	2.45	3.56

Note: If PuRe confidence falls below the indicated threshold, a naive crop is applied.

Selecting the 'best' CRs using model output

The LEyes U-Net model takes a grayscale eye image as input and produces output maps for each feature (the pupil and each CR) that correspond to the confidence the model has that a given feature's center is located at a given position in the input image. We will represent these unnormalized output values, which we will refer to as logit values, in the form of a heatmaps. The maximum value of each heatmap corresponds to where the model is most confident of the prediction for the pixel location of each eye feature's center. To robustly select the two CRs the model is most confident about, we choose the two CRs with the highest corresponding logit values across the output heatmaps. Figure 13 shows the heat maps of each CR and their associated max values derived from real eye images from both datasets along with the predicted locations of the selected CRs overlaid onto the eye image. To exclude eye images clearly unsuitable for eye tracking, for instance images that contain a blink or when both cropping strategies failed to capture the pupil, our method excludes images that fail to produce at least two heatmaps where the

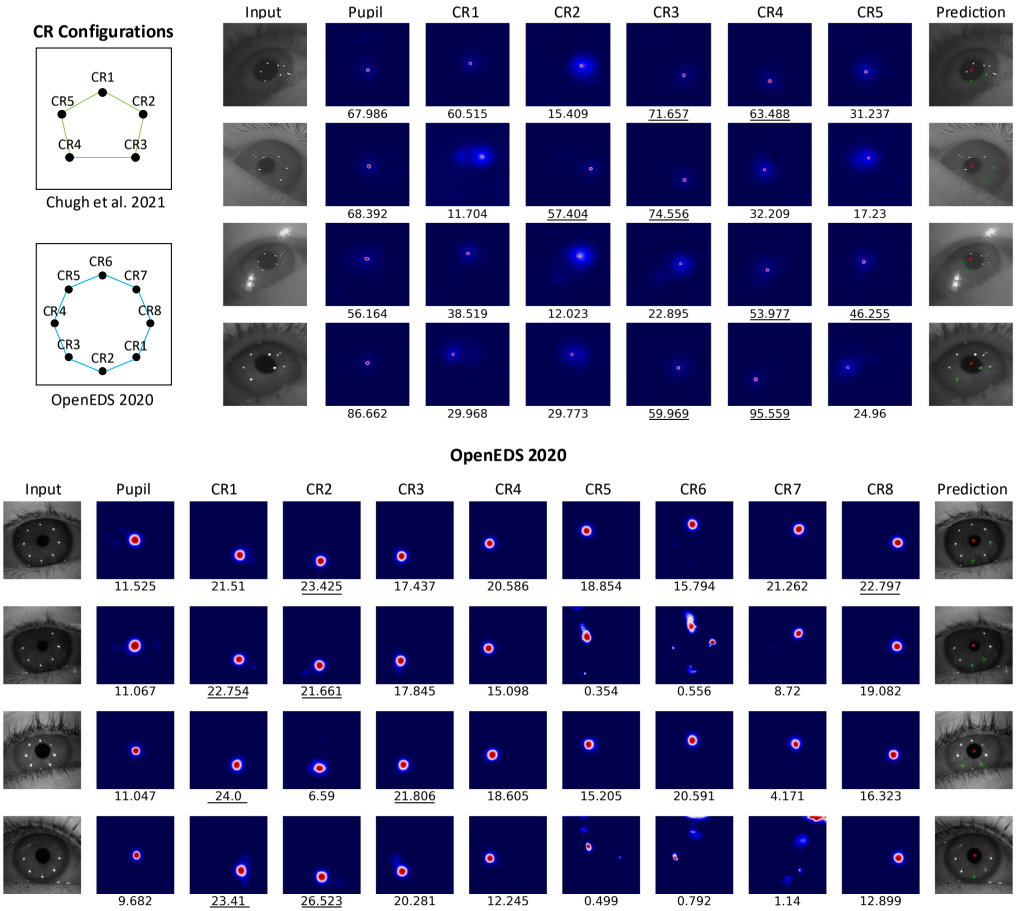


Figure 13: Heat maps for both the Chugh et al. 2021 dataset and the Openeds 2020 dataset. The maximum of the corresponding logit value is shown under each heat map. In the Chugh et al. 2021 dataset, the labeling of the CRs starts at the top-most IR reflection and then proceeds clockwise (top right). In the OpenEDS 2020 dataset, the labels used when training the model start at the lower right CR and proceed clockwise. Our algorithm selects the two highest logit values from the CR maps along with the pupil value for a complete robust P-CR pipeline. The last column shows the prediction locations of the centers of the pupil and selected CRs on the corresponding eye image.

max value are greater than or equal to one.

To assess our model, we compared its average pixel error to the CR annotations in the Chugh et al. 2021 dataset (Chugh et al., 2021). They achieved successful matches of at least two CRs within five pixels for 91% of the images in their test set and an average error of 1.5 pixels on these images. It is worth mentioning that Chugh et al. 2021 had to sacrifice 88% of the dataset for both training and validation of the model (Maquiling, Byrne, Nyström, et al., 2023), so their results include only a small part (12%) of the whole dataset. In contrast, since LEyes is trained on synthetic images, we can evaluate our model on the entire dataset. Therefore, direct comparisons between the two models are not straightforward since they are evaluated on a different number of images and use different exclusion criteria. To make the results more comparable, we apply our exclusion criterion that the maximum value of at least two heat maps is larger than one in conjunction with Chugh et al. (2021)'s criterion that evaluates model performance only on the images where the predicted locations of at least 2 CRs were less than 5 pixels away from the ground truth. Using these criteria, our model exhibited an average pixel error of 1.47 across all the CRs. Focusing solely on the best two CRs, this error was reduced by 11% to 1.31 pixels. Further, using both exclusion criteria we retain 84% of images from the dataset.

3.1.3 High-Resolution Gaze Tracking

The Pupil-Corneal Reflection (P-CR) eye tracking method, often employed in controlled lab settings for gaze estimation, requires accurate identification of both the pupil and Corneal Reflections (CRs) (Hooge, Holmqvist, and Nyström, 2016; Nyström, Niehorster, Andersson, Hessels, and Hooge, 2023; Fuhl, Santini, Kasneci, Rosenstiel, et al., 2017). When estimating the smallest and fastest of eye movements, an eye tracker with high spatial and temporal resolution is required. This typically requires sub-pixel localization of the pupil and CR(s).

To address these requirements, we developed a dual Convolutional Neural Network (CNN) model trained on LEyes images. One CNN fo-

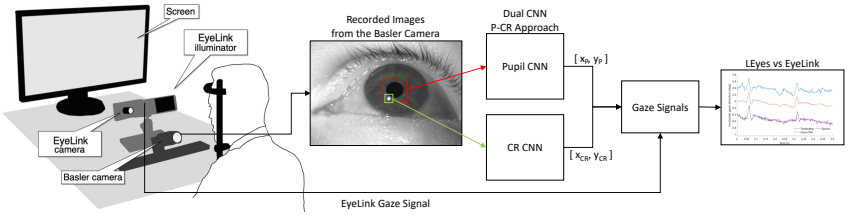


Figure 14: Experimental setup: In a co-recorded setup we acquire eye images from the FLEX setup and gaze signals from the Eyelink 1000 Plus. We analyzed the eye images which we recorded from expert participants using a dual CNN approach. The pupil CNN localized the pupil center, while the CR CNN localized the center of the CR located in the eye image. Both CNNs achieved sub-pixel pixel error. Image of co-recording setup adapted from (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2023).

cuses on locating the pupil center, while the other locates the CR center; an illustration of our setup is in Figure 14. Our model was compared with traditional thresholding methods, the LEyes U-Net model used in OpenEDS 2019 but with different parameters used in generator to account for the dataset, and a state-of-the-art commercial eye tracker (SR Research EyeLink 1000 Plus).

The data for this high-resolution study was captured in a co-recorded experiment using our custom-built FLEX system (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2023; Hooge, Niehorster, Hessels, Cleveland, et al., 2021) and the EyeLink 1000 Plus eye tracker. Such co-recording was required since the eye images captured by the EyeLink are not accessible and a direct comparison of its image processing to the LEyes method is thus not possible. The EyeLink's illuminator was used to deliver illumination for both systems. This setup resulted in eye images containing a single CR. Both the FLEX system and the EyeLink acquired data at 1000 Hz. Since the focus of this comparison is on eye tracking signal quality, data was recorded from 4 expert participants who performed a series of fixation and saccade tasks during eight minutes. To provide additional variation in the luminance profiles of the eye images, and thereby test the robustness of our model, the four participants were

recorded a second time with the FLEX system configured to a sampling rate of 500 Hz. The captured eye images were brighter at this lower sampling rate due to the longer possible exposure time.

The eye images captured by the FLEX system were first processed using a standard thresholding operation (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2023) to provide an initial localization of the pupil and CR centers. We then took 180x180 pixel crops centered on the pupil and the CR features from the original images and fed these to the LEyes CNNs. As shown in Figure 15, both thresholding and in particular the LEyes CNNs provided a significant improvement in stability of the pupil signal compared to using the LEyes U-Net model, which therefore was omitted from further analysis.

Example raw pupil and CR signals resulting from the thresholding operations and the LEyes CNNs are shown in Figures 16a (1000 Hz) and 16c (500 Hz). As can be readily appreciated, the sample-to-sample variation in both the pupil center and the CR center signal is lower for the LEyes method than for the standard thresholding method for data acquired at both sampling rates. To formalize this observation, the precision in the form of root mean square of sample-to-sample deviations in the signal (RMS-S2S (Holmqvist, Nyström, and Mulvey, 2012; Niehorster, Santini, et al., 2020; Niehorster, Zemblys, Beelders, et al., 2020)) was computed across the dataset and plotted in Figures 16b (1000 Hz) and 16d (500 Hz). This analysis confirms that for both the 1000 and the 500 Hz data sets, the LEyes CNNs consistently demonstrated superior precision (lower values) than the thresholding method.

Researchers using eye tracking are rarely interested in the individual pupil and CR signals, but instead use the gaze signal derived from them. Does the improved precision of the pupil and CR center signals lead to an improved gaze signal? To examine this, we derived P-CR gaze signals using pupil and CR centers estimated by thresholding or by the LEyes CNNs and compared both with the gaze signal delivered by the EyeLink. Each signal was calibrated using standard methods and example segments are plotted in Figures 17a (1000 Hz) and 17b (500 Hz). Again, it can be readily appreciated that the gaze signal derived from the

LEyes CNNs is smoother and more stable than that derived from standard thresholding operations or delivered by the EyeLink. To quantify this observation, we computed the RMS-S2S precision of these signals which quantifies short-timescale smoothness, as well as the STD precision (Niehorster, Zemblys, Beelders, et al., 2020) which quantifies the spatial spread of the signal and indicates its stability. These evaluations are presented in Figures 17c–e (1000 Hz) and 17f–h (500 Hz). This analysis confirms that the dual LEyes CNN method consistently demonstrated superior RMS-S2S precision (lower values) than the thresholding method and the results from the Eyelink 1000 Plus. It is important to note that all methods processed each video frame independently, without using any temporal information from preceding or future frames. Thus, the increased precision seen in the CNN method cannot be attributed to the use of temporal information (Niehorster, Zemblys, and Holmqvist, 2021b; Niehorster, Zemblys, Beelders, et al., 2020). The signal stability (STD precision) achieved by the LEyes method was on par with the EyeLink for the 1000 Hz dataset and slightly better than the EyeLink for the 500 Hz dataset, and consistently outperformed the thresholding method for both datasets. The accuracy achieved did not systematically differ between the three methods, indicating that the gains in precision did not come at the cost of reduced accuracy.

3.2 Concluding Remarks

Our study has limitations: First, the models were trained on simulated data but tested on real data. We did not investigate any potential learning differences between synthetic and real eye datasets. Future research may benefit from analyzing these differences to further improve the quality of the synthetic data generation. Second, we aim to explore "Domain Adaptation" techniques such as fine-tuning LEyes-trained models with real eye images to assess performance impact. Third, the low participant count in our high-resolution experiment, while sufficient for our purpose of demonstrating the power of a LEyes trained model, potentially limits the generalization of our findings for this specific test. Despite seeing

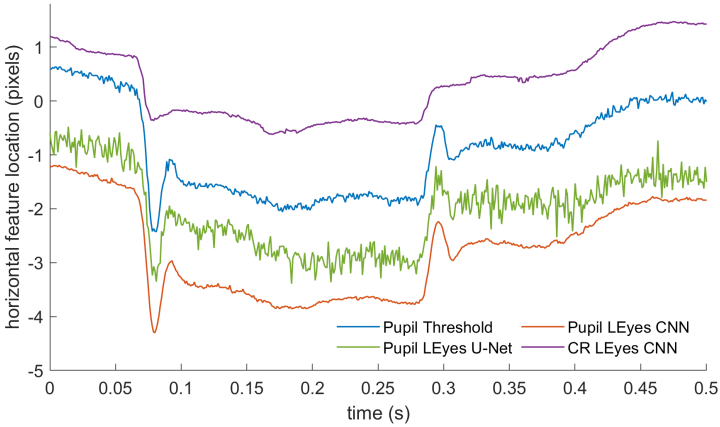


Figure 15: Representative segment of pupil and CR center locations derived from 1000 Hz eye images. The pupil center was determined using three different methods; thresholding (blue), a U-Net trained using the LEyes framework and derived from the EDS2019 U-Net (green), and a CNN trained for pupil center localization using LEyes images (red).

promising results with the LEyes framework and good generalizability across large participant samples in the other tests, recruiting a broader participant base that encompasses both experts and novices can be seen as a worthwhile further study.

We developed a novel framework named LEyes for training gaze estimation algorithms, achieving cutting-edge results for both virtual reality (VR) and high-resolution, lab-based eye-tracker setups. LEyes outperformed other methods in a pupil center localization task by a margin of at least 4%. In a high-resolution setting, LEyes exceeded the performance of the industry-standard EyeLink 1000 Plus eye tracker across two lighting conditions in a co-recorded experiment. Additionally, we introduced a novel LEyes-trained P-CR pipeline that both simplifies and improves CR detection by considering only the two best CRs in the recorded image. Overall, our results emphasize both the accuracy and flexibility in design of the LEyes framework, highlighting its applicability across gaze estimation applications.

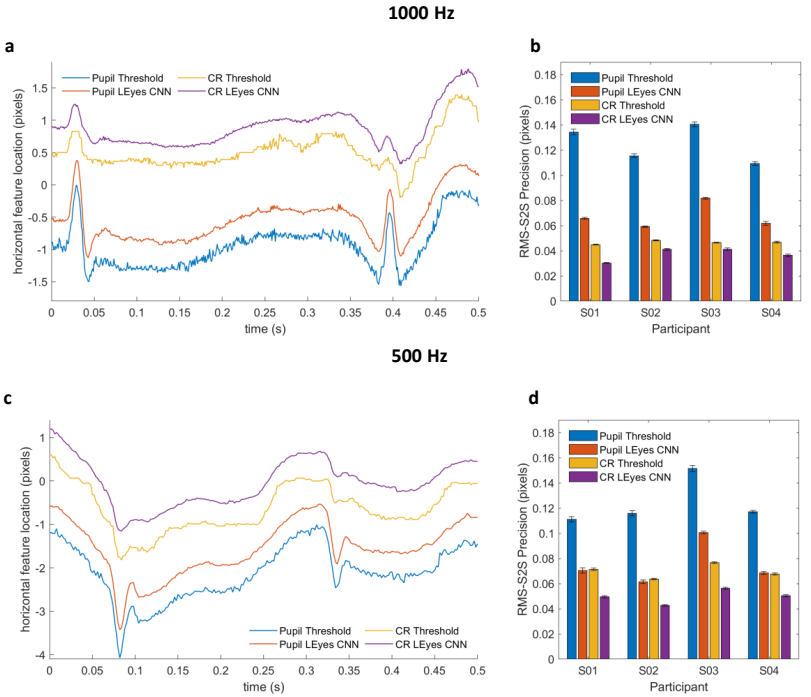


Figure 16: CR and pupil center signals. Left column: representative segment raw pupil and CR center signals derived from eye images recorded at 1000 Hz (a) and 500 Hz (b). Right column (panels b and d): an RMS precision comparison between the thresholding and LEyes CNN methods for the pupil and CR signals on all data of four participants. Error bars depict standard error of the mean.

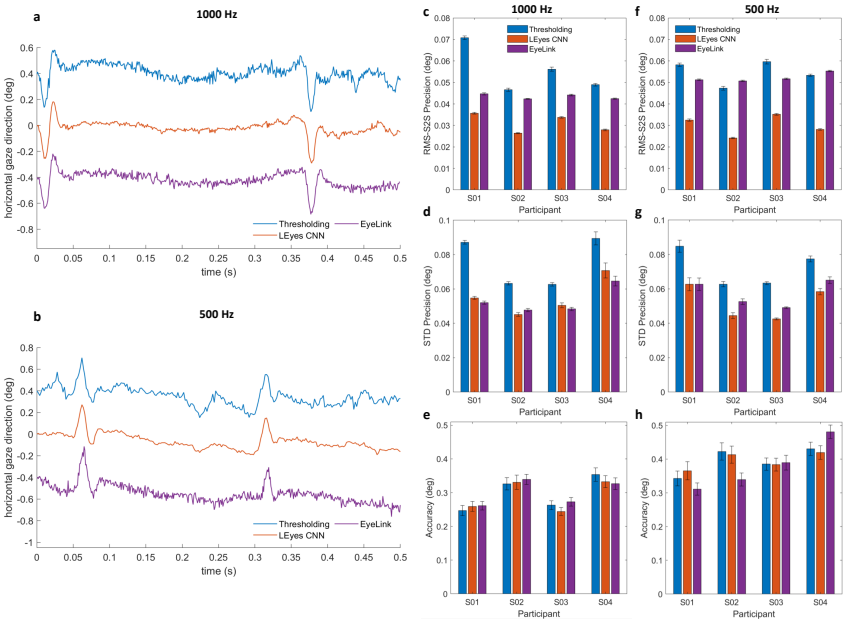


Figure 17: Calibrated gaze signals. Left column: representative segment of calibrated P-CR signals derived from 1000 Hz data (a) and 500 Hz data (b) as derived from pupil and CR center locations determined using either thresholding or the dual LEyes CNN strategy, along with the EyeLink. The signals in both panels contain two small saccades and have been vertically offset for clarity. Further, an RMS precision, STD precision and an accuracy comparison for the 1000 Hz data (middle column, panels c–e) and the 500 Hz data (right column, panels f–h) between the three gaze tracking methods on data of all participants are shown. Error bars depict standard error of the mean.

Chapter 4

LEyes Methods & Models Used

[1]

4.1 Model Architecture and Training for Detecting Corneal Reflections Used in Chapter Two

In a preliminary test, we implemented the original DeepTrack CNN model as described in Helgadottir, Argun, and Volpe (2019). This model consisted of three convolutional layers and two dense layers, and we em-

[1] This chapter presents a thorough examination of the technical methodologies required to replicate the findings detailed in Chapters Two and Three, reflecting their close interrelation. It acts as an exhaustive guide, detailing the specific techniques, algorithms, and tools utilized in the research. These methodologies are essential not only for reproducing the results of this thesis but are also comprehensively described in corresponding academic publications: Sean Anthony Byrne, Marcus Nyström, et al. (Dec. 2023). "Precise Localization of Corneal Reflections in Eye Images Using Deep Learning Trained on Synthetic Data". In: *Behavior Research Methods*. ISSN: 1554-3528. DOI: 10.3758/s13428-023-02297-w. URL: https://doi.org/10.3758/s13428-023-02297-w and Sean Anthony Byrne, Virmarie Maquiling, Marcus Nyström, et al. (2023). *LEyes: A Lightweight Framework for Deep Learning-Based Eye Tracking using Synthetic Eye Images*. arXiv: 2309.06129 [cs.CV]. Additionally, all the necessary code for these methods is accessible at the following links: https://github.com/dcnieho/Byrneetal_CR_CNN and https://github.com/dcnieho/Byrneetal_LEyes, facilitating practical application and exploration allowing the reader to freely adapt our methods for their own studies.

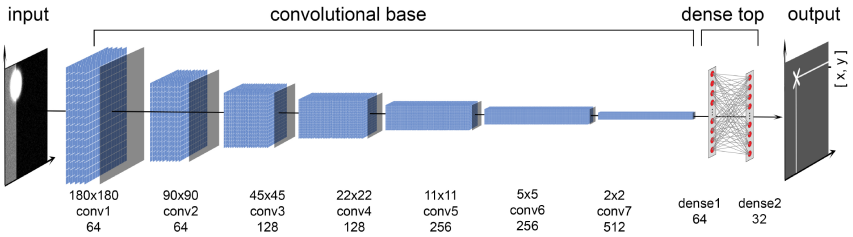


Figure 18: Overview of our method: A CNN model with seven convolutional layers that increase in filter size from 64 to 512 and two dense layers returning the Cartesian coordinates of the CR center.

ployed the same optimizer and hyperparameter choices as described in the original work. However, when we evaluated this model on our synthetic images that simulated a corneal reflection captured in a video-based eye tracking setup, we were unable to achieve sub-pixel level accuracy. The minimum validation error we reached was 2.95 pixels. To enhance the accuracy of our predictions, we developed our own CNN model. Our model included seven convolutional layers connected to two dense layers. The input to our model is 180 x 180 pixels grayscale images, and it outputs the subpixel location of the corneal reflection center. Figure 18 provides a visual representation of the complete architecture of our model.

We implemented a two-stage training approach for our model to achieve sub-pixel level accuracy. In the first stage, to ensure good generalization, we trained the model on a broader range of CR center locations than the model would typically encounter during inference. We describe the process of generating the images in detail in the following section of the paper. We utilized the Adam optimizer (Kingma and Ba, 2017) and a mean squared error (MSE) loss function along with a very small batch size of four. The training was conducted for a maximum of 700 epochs, with an early stopping function implemented to prevent overfitting, achieving a validation error of 0.2338 pixels after 127 epochs. The second stage of training was performed on a dataset containing a smaller range of synthetic CR center locations. During this stage, we fine tuned the model

by freezing the first two convolutional blocks while all subsequent layers of the model were set to trainable (i.e unfrozen), and we initialized the model with the weights from the first training stage. For selecting the layers to freeze, we followed an iterative process where we gradually increased the number of trainable layers. We initiated the process with a fully frozen model and subsequently unfroze the layer closest to the model head at each iteration. Additionally, we lowered the learning rate of the Adam optimiser from $1e^{-4}$ to $1e^{-6}$. The second stage of training resulted in a sub-pixel accuracy of 0.085 after 187 epochs on the validation set.

The DeepTrack 2.1 (Midtvedt, Helgadottir, et al., 2021) package provides a generator function which we used to efficiently generate and feed images into the model for training. We set up the generator such that the model only saw each training image one time, meaning that every image the model saw for training was unique. We additionally generated 300 synthetic images for the validation set. The fully trained model was saved and model evaluations were conducted on an Intel Xeon W-10885M CPU @ 2.40GHz with a prediction time of 13ms per image.

4.2 Generating Synthetic Images Used in Chapter Two

Model of Image information for CR center localization

Since the aim of the current work is to develop a high-accuracy CR center localization method, it is important to develop a model of what accuracy an optimal localization method could achieve. As shown by (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2022), CR localization accuracy depends on the number of pixels spanned by the CR in the image as well as the shape of the light distribution. Theoretically, the lower the spatial pixel resolution or bit-depth of the CR image, the lower is the maximum achievable localization accuracy of the CR center. This follows from the logic that the coarser the digital representation of the CR, the bigger the change in its position needs to be before an observable

change occurs in the CR image (*cf.*, Mulligan, 1997). Therefore, to provide a benchmark for the results presented in this paper, we determined the theoretically optimal center localization performance as a function of CR size and Gaussian amplitude (i.e. tail width). To do so, we took a set of CR images generated with the Cartesian product of $r = \{2, 4, 6, ..., 18\}$ and $A = \{10, 50, 200, 1000, 10000\}$, i.e., the same parameters as used for the evaluation on synthetic images (see the section "Evaluation Criteria for Synthetic Images" below). The center of each CR image was then estimated as the center-of-mass of all the pixels in the discretized CR image, using their intensity values (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2022; Shortis, Clarke, and Short, 1994). Unlike the synthetic images used for model evaluation, these images had a completely black background such that only pixel intensity values associated with the CR would influence the center estimate.

Synthetic Images used First & Second Training Stages

During the first training stage, the following parameters were used. Where possible, the parameters were set to ranges significantly larger than the set used for evaluation.

1. CR radius r was drawn from a uniform distribution with range $[1, 30]$ pixels. This was chosen to be wider than our testing range of $[2, 18]$ (like was used in Nyström, Niehorster, Andersson, Hessels, and Hooge, 2022) and also encompasses the range of CR sizes one may reasonably expect to encounter in real eye images.

2. Location: Horizontal and vertical CR center locations were drawn from uniform distributions. To ensure that the CR would not be significantly cut off by the edge of the image, the range of both uniform distributions depended on the CR size (r). Specifically, they spanned $[r, 180 - r]$ pixels, where 180 pixels is the image size.

3. Gaussian amplitudes A were drawn from a uniform distribution with range $[2, 20000]$. The range of this parameter was decided by

means of manual inspection of the output to provide a range of different tail widths (c.f. Figure 1).

4. The horizontal and vertical coordinates of a point on the line dividing the two sections of the background were drawn from a normal distribution centered on the CR center location and spanning a standard deviation of $-1.5r$. A random orientation of this line was then drawn from a uniform distribution with range $[0, 2\pi]$. The edge between the two segments was smoothed with a raised cosine profile spanning 4 pixels. The pixel intensity value of the dark section of the background was drawn from an exponential distribution with its scale parameter set to 10 pixel intensity values, and offset 1 (so that full black did not occur). The pixel intensity level of the lighter section I was drawn from a uniform distribution with a range of $[32, 153]$ pixel intensity values.

5. The pixel noise σ_n was drawn from a uniform distribution with range $[0, 30]$ pixel intensity values.

In the second training stage, all parameters except CR location were set to the same ranges as were used in the first stage. Since the CNN will only be used on image patches where the CR has already been centered, horizontal and vertical CR center locations in this second pass were drawn from uniform distributions with ranges spanning 1.5 pixels around the center of the output image, i.e. $[89.25, 90.75]$.

Features of the Synthetic Images

As in previous work (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2022), the light distribution of the CR in an eye image is modeled as a 2D Gaussian distribution, as is supported by optical modeling (Wu et al., 2022). CRs in real eye images have at least two further important features: 1) The CR in an eye image is normally heavily oversaturated (Wu et al., 2022; Holmqvist et al., 2011); and 2) depending on the physical geometry of the setup and the orientation of the eye, the CR is often overlaid on a non-uniform background, such as the iris or

the edge of the pupil. We, therefore, extend the approach of Nyström, Diederick C Niehorster, Andersson, Hessels, and I. T. C. Hooge (2022) by introducing saturation that truncates the Gaussian distribution and leaves it with an area of maximum brightness surrounded by shallow tails, and by introducing a non-uniform background.

More formally, the saturated CR is generated from a Gaussian distribution

$$G(x, y) = Ae^{-\left(\frac{(x-x_c)^2+(y-y_c)^2}{2\sigma_w^2}\right)} \tag{4.1}$$

where the following parameters are varied in the simulation:

1. The center of the input light distribution (x_c, y_c).
2. The amplitude A of the Gaussian distribution. Saturation is achieved when A is set to amplitudes larger than 1 since image values are limited to 1 at the end of the image generation pipeline (see below). Figure 1 (top-left) shows CRs at three different amplitudes. Notice that larger amplitudes lead to shallower tails.
3. The radius r of the resulting CR. This is parameterized such that for a given value, the radius of the saturated portion of the CR is kept constant irrespective of the amplitude (A) of the underlying Gaussian. This is achieved by setting $\sigma_w = r/\sqrt{-2\log\frac{1}{A}}$.

Two further aspects were varied to generate the final synthetic images. First, to simulate the pupil-iris border, a background was generated that consisted of two sections of different luminance, and the line dividing the two sections was randomly placed near the CR and randomly oriented. The image of the synthetic CR was added to this background using the following operation: $max(CR, background)$. The top row of Figure 1 shows synthetic CRs on various example backgrounds. Furthermore, noise was added to the images by adding a value drawn from a Gaussian distribution $X \sim \mathcal{N}(0, \sigma_n^2)$ for each individual pixel of the image. The parameter σ_n^2 was varied (see Figure 1, top-right). Finally, the intensity values in the resulting images were limited to the range $[0, 255]$, scaled to the range $[0, 1]$ and the image was discretized to 256 levels, corresponding to 8-bit camera images. .

4.2.1 Generating Light Simulations Used in Chapter Three

Five different simulations modeling the light distribution of eye images were used for training the U-Net for OpenEDS 2019, the U-Net models with attention used on the OpenEDS 2020 and Chugh et al.'s 2021 datasets and the CR and pupil CNNs. Here we first present features shared between these simulations, and then detail the individual simulations in order of complexity. The full code to generate LEyes images is available at our GitHub Repository linked to this paper.

Common features

Following previous work (Byrne, Nyström, et al., 2023; Maquiling, Byrne, Nyström, et al., 2023), we developed simulated images that model the light distributions of the relevant aspects of an eye image that the given model would have to deal with during inference. Blob-like features, such as the pupil and CR were modeled as 2D Gaussian distributions using the Gaussian equation 4.1.

where

$$a = \frac{\cos(\theta)^2}{2\sigma_\alpha^2} + \frac{\sin(\theta)^2}{2\sigma_\beta^2}, \tag{4.2}$$

$$b = \frac{\sin(2\theta)}{4\sigma_\alpha^2} - \frac{\sin(2\theta)}{4\sigma_\beta^2}, \tag{4.3}$$

$$c = \frac{\sin(\theta)^2}{2\sigma_\alpha^2} + \frac{\cos(\theta)^2}{2\sigma_\beta^2}, \tag{4.4}$$

and where θ is the orientation of the 2D Gaussian and σ_α and σ_β its spread along the minor and major axes, respectively.

The luminance of the pupil was determined per simulation by analyzing the eye images on which inference would be run, while the luminance of a CR was always set to full white. Regardless of the Gaussian amplitude A of the feature, which was varied to create differently steep edges, the minor and major axis radii of the luminance plateau in each feature (the dark part of a pupil, or the bright part of a CR) were kept

constant by parameterizing

$$\sigma_r = r/\sqrt{-2\log\frac{1}{A}}, r \in \{\alpha, \beta\}. \tag{4.5}$$

To create the final simulated image, first the relevant features were layered onto a background luminance distribution that differed between simulations. These layers were then collapsed into a single image by subtracting dark features (such as pupils) from the background, and by adding bright features to the collapsed image of the preceding layers using the operation $max(image, background)$. Pixel noise was added to the final image by adding a value from a Gaussian distribution $X \sim \mathcal{N}(0, \sigma_n^2)$ to the image that was drawn independently for each pixel. Finally, the resulting image was limited to the range $[0, 255]$, scaled to the range $[0, 1]$ and discretized to 256 levels, corresponding to 8-bit camera images.

CR 500 Hz & CR 1000 Hz

The CNN for CR center localization used for the 500 Hz data was the same as presented in previous work(Byrne, Nyström, et al., 2023) in Chapter 2. As such, only the key points of this simulation are described. Circular CRs ($\sigma_\alpha = \sigma_\beta \in [1, 30]$, $A \in [2, 20000]$) were placed on a background that was made up of two parts, divided by a randomly oriented straight line representing the pupil-iris border that passed close to the CR. On one side of the line the background was dark, with a luminance drawn from an exponential distribution with its scale parameter set to 10 pixel intensity values, and offset 1. The other part of the background was middle grey (pixel intensity value $L_{CR} = 128$). The standard deviation of image noise was varied per generated image, with $\sigma_n \in [0, 30]$.

The simulations used for training the CNN for determining CR centers in the 1000 Hz eye videos were identical to those used for the 500 Hz data, except that the middle-grey part of the background varied in luminance between $L_{CR} \in [32, 153]$.

Pupil 500 Hz

The simulated light distributions used for training the CNN for locating pupil centers differed from the simulations for the CR CNNs in a few ways. First, the simulated images contained a 2D Gaussian representing the darker pupil. Second, the images contained one or multiple bright 2D Gaussians representing CRs that were randomly positioned and could thus overlap the pupil. Third, instead of a background consisting of dark and grey segments separated by a straight line, the background now consisted of a uniform field at a range of grey levels, representing the iris at various illumination levels.

Specifically, a randomly oriented dark 2D Gaussian with minor axis radius $\alpha_p \in [20, 60]$ pixels, major axis radius $\beta_p \in [1\alpha_p, 1.3\alpha_p]$ and amplitude $A_p \in [2, 20000]$ was used to represent the pupil. Its luminance L_p was drawn from an exponential distribution with a scale parameter of 10, and offset 1. Between 1 and 4 corneal reflections (CRs) were generated with minor axis radius $\alpha_c \in [4, 12]$ and major axis radius $\beta_c \in [1\alpha_c, 1.1\alpha_c]$ and $A_c \in [2, 20000]$ and randomly positioned. Overlap between CRs was avoided by removing CRs whose center location was closer to another CR than 1.25 times the sum of the major axis radii of the two CRs, and replacing it with a new randomly positioned CR. The background luminance level representing the iris was $L_{background} \in [64, 179]$ pixel intensity values.

Pupil 1000 Hz

The simulations used for training the CNN for determining pupil centers in the 1000 Hz eye videos were identical to those used for the 500 Hz data, except that the background luminance level representing the iris was $L_{background} \in [32, 153]$ pixel intensity values to encompass the iris luminance values in the darker 1000 Hz eye images.

U-Net for OpenEDS 2019

In order to ensure that the U-Net reliably detects the pupil and not the iris, the simulations used for training the U-Net contained several more

features than those for the pupil CNN. Firstly, a bright background representing the sclera with luminance $L_s \leftarrow \mathcal{N}(217,\ 26)$ was generated. On top of this an iris was generated as a randomly positioned and oriented 2D Gaussian ($\alpha_i \in [30, 42.5]$ and major axis radius $\beta_i \in [1\alpha_i, 1.3\alpha_i]$, $A_i \in [20, 500]$ and $L_i \leftarrow \mathcal{N}(77,\ 16)$). Then an irregularly shaped collarette was generated close to the center of the iris consisting of between 13 and 24 vertices arranged around the collarette center at an average distance $r_{col} \in [.3\beta_i, .6\beta_i]$, with the individual distance of vertices varied between $[0.05r_{col}, 0.2r_{col}]$. The resulting polygon was upsampled to five times the number of vertices using periodic cubic spline interpolation to create a shape with a smoothly varying edge, and the resulting polygon was rendered at luminance $L_{col} = [1.25L_i, 1.6L_i]$ with an edge modulated by a raised cosine function over a range of between $[1, 4]$ pixels.

On top of this were layered a randomly positioned and oriented pupil (minor axis radius $\alpha_p \in [10, 30]$ and major axis radius $\beta_p \in [1\alpha_p, 1.3\alpha_i]$, $A_p \in [2, 2000]$ and $L_p \leftarrow \mathcal{N}(34,\ 15)$) and between 1 and 8 randomly positioned and oriented CRs (minor axis radius $\alpha_c \in [0.8, 4]$ and major axis radius $\beta_c \in [1\alpha_c, 1.4\alpha_c]$, $A_c \in [2, 20000]$ and $L_{CR} = 255$), again avoiding overlap.

U-Net for Chugh et al. 2021 dataset

We use a simulation that improves on previous work (Maquiling, Byrne, Nyström, et al., 2023) to perform pupil and CR localization and CR matching. The pupil is represented by a randomly oriented dark 2D Gaussian with a minor axis radius $\alpha_p \in [6, 22.5]$ pixels, major axis radius $\beta_p \in [1\alpha_p, 1.3\alpha_p]$ and amplitude $A_p \in [200, 100000]$. Its luminance L_p is drawn from an exponential distribution with a scale parameter of 10 and offset 1. Five randomly oriented CRs are generated, each having a random minor axis $\alpha_c \in [1, 2.5]$ pixels, a random major axis $\beta_c \in [\alpha_p, 1.1\alpha_p]$, and a random amplitude $A_c \in [200, 100000]$. Each CR has a drop-out rate of 16%. Some spurious (non-CR) reflections may randomly appear in the image. These are generated in the same way as CRs, each with a random minor axis radius $\alpha_s \in [1, 2.5]$ pixels and random major axis radius $\beta_s \in [\alpha_s, 2.5\alpha_s]$. The location of each spurious reflection is generated us-

ing a rejection sampling method with an inverted Gaussian ($1 - G(x, y)_p$, c.f. Eq 4.1) to make them less likely to appear near the pupil center. Randomly, between 0 and 5 of such spurious reflections were generated. A grayscale gradient background was created by drawing two random values from a luminance range of $L_{background} \in [63, 178]$ and smoothly varying the luminance from one side to the other along a random axis. This is to prevent the model from interpreting any dark part of the image as part of the pupil.

As this model not only performs pupil and CR center localization but also matching of CRs to specific illuminators, the positions of the CRs need to follow the same pattern as in the real dataset. Specifically, for Chugh et al.'s 2021 dataset (Chugh et al., 2021), this involves five IR lights that project to a house-shaped polygon that is usually close to the pupil. The polygon is modeled as a rectangle with an additional vertex above the middle of its top edge. The rectangle's base width is randomly sampled from $w \in [0.1d, 0.45d]$ where $d = 128$ pixels, the length of one side of the synthetic image. The rectangle's height is sampled from $[0.5w, 0.6w]$, and the height of the roof from $[0.2w, 0.5w]$. The polygon is randomly rotated between $\pm[0, 45]$ degrees.

In order for the model to learn the matching correctly, the CR positions are always calculated in a certain order, starting from the topmost position and moving clockwise. Training this model was performed in two stages (see below). In the second stage, the maximum number of spurious reflections that could appear in the image is reduced to 3, the dropout probability for individual CRs is reduced to 10% and the range of rotation is reduced to $\pm[0, 35]$.

U-Net for OpenEDS 2020

We reuse the simulation created for Chugh et al. 2021 dataset (Chugh et al., 2021), adjusting the polygon so that it has eight vertices corresponding to the eight IR lights in the dataset, starting from the bottom-right CR and moving clockwise. The polygon's radius is randomly sampled from the range $w \in [0.15d, 0.4d]$ where $d = 128$ pixels. As the OpenEDS 2020 dataset contained forward-facing eye images, the random rotation

of the polygon is reduced to the range $\pm[0, 0.01]$ degrees. Each CR has a dropout rate of 20%. The pupil luminance L_p is drawn from a Weibull distribution with a scale of 25, an offset of 18 and shape parameter of 2, while no other parameters were changed.

4.3 Neural Networks & Training Regimes

U-Net model for pupil segmentation

For the pupil segmentation task, we utilized an off-the-shelf U-Net (Ronneberger, Fischer, and Brox, 2015) from the PyTorch Segmentation Modules library (Iakubovskii, 2019). The encoder part uses a ResNet-34 backbone (Koonce and Koonce, 2021) pre-trained on ImageNet (Deng et al., 2009). The decoder part consists of five convolutional layers of dimensions (256, 128, 64, 32, 16). The trained U-Net model accepts grayscale images of arbitrary dimensions and produces a probability map that represents the pupil segmentation. In total, the U-Net model contains 24,430,097 trainable parameters.

The masks output by the U-Net (range $[0, 1]$) were binarized using a threshold of 0.99, and then postprocessed with OpenCV (version 4.7.0.68) in Python 3.10. Specifically, morphological operations were performed on the resulting binary masks to fill holes, and the pupil was selected based on shape and size criteria (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2023). The center of mass of the blob was then computed and an ellipse was then fit to the selected blob. If the center of mass was closer than the radius of the ellipse's major axis to the edge of the eye image cutout, the cutout was recentered on the center of mass and inference run anew on this cutout.

U-Net with attention mechanism

In both cases, we used a modified U-Net model based on previous work (Niu et al., 2021). The encoder and decoder consist of residual modules producing a feature map with a consistent depth, only decreasing/increasing in size using down-and upsampling respectively. The U-Net contains six

residual modules with a consistent channel size of 256. The output of the U-Net are passed through two convolution blocks which produce the heat maps for the CRs and the pupil respectively. The peak in the heat maps is taken as the pixel location of each eye feature center and is found with an argmax operation. Invalid cases (where a CR is not visible e.g. due to the CR being occluded by the eyelid) return the XY-position $(-1, -1)$.

CNN models for pupil and CR localization

In this task, the model was trained to localize the subpixel center of an eye feature (pupil or CR). Overall, four CNNs were trained: two for localizing the pupil and CR centers in data captured at 500 Hz with the FLEX setup and another two for data captured at 1000 Hz. Each model is composed of seven convolutional layers followed by two dense layers. The CNNs for CR center localization in both 500 Hz and 1000 Hz data as well as the CNN trained to detect the pupil center in 500 Hz data have the following convolution layer dimensions: (64, 64, 128, 128, 256, 256, 512) while the pupil CNN for 1000 Hz data has wider dimensions: (128, 128, 256, 256, 512, 512, 768). The CR CNNs have dense layers with sizes of (64, 32) while the pupil CNNs both have sizes of (64, 64). The CR CNNs both have a total of 6,268,386 trainable parameters while the pupil CNN for 500 Hz and 1000 Hz data have 6,270,530 and 19,671,426 trainable parameters, respectively. Each CNN model was built within the DeepTrack 2.1 library (Midtvedt, Helgadottir, et al., 2021).

Model training regimes

To train the U-Net for the OpenEDS 2019 dataset, we chose the AdamW (Loshchilov and Hutter, 2019) optimizer with an initial learning rate set to $1e^{-4}$ and an exponential decay scheduler. The loss used is a combination of Binary Cross Entropy loss (Ruby and Yendapalli, 2020), Dice loss (Li et al., 2019), and Focal loss (Lin et al., 2017). During the training phase, the model was shown 1000 new simulated images per epoch and the validation set consisted of 400 pre-generated simulated images. The

model training ran for 100 epochs reaching a natural plateau.

Following our previous work (Byrne, Nyström, et al., 2023; Maquiling, Byrne, Nyström, et al., 2023), the U-Net model used for Chugh's dataset (Chugh et al., 2021) is trained in two stages. The first stage consisted of a broader range of challenging examples, aimed at enhancing the model's robustness to large variations in eye data while the second stage consisted of images that more closely represent the images captured by the eye tracker. Similar to the U-Net model for EDS 2019, we used the AdamW optimizer with an initial learning rate of $1e^{-4}$ in the first stage and $1e^{-5}$ in the second stage, an exponential decay scheduler, and a combination of Binary Cross Entropy loss (Ruby and Yendapalli, 2020), Dice loss (Li et al., 2019), and Focal loss Lin et al., 2017 for the loss. The generator was first configured to present the model with 20000 unique images per epoch. In the second stage, the generator is reconfigured to show 1000 images. We let the model train for 30 epochs in the first stage and 20 epochs in the second stage.

Similarly, the U-Net model for EDS 2020 (Palmero et al., 2021) is trained in two stages. We incorporated early stopping with a patience of 30 for both stages. In the first stage, a weight of 100 is added to the Binary Cross Entropy Loss, while the rest of the parameters for both the first and second stages remain the same as the first U-Net model. In both stages, the generator was configured to produce 1000 unique images per epoch, early stopping after 175 epochs in the first stage and 81 epochs in the second stage.

Similar to the above, we adopted a two-stage approach for training each CNN, training first on simulations with harder examples and then honing in on cases that are closer to the dataset. The generator was configured to present the model with 1000 unique samples per epoch, with batch sizes of 4 for the CR CNNs, 16 for the pupil CNN at 500 Hz, and 8 for the pupil-CNN at 1000 Hz. The batch size was further reduced to 4 for both pupil CNNs during the second stage of training. Additionally, a set of pre-generated synthetic images was used for validation, with a validation set size of 300 for the CR CNNs and 600 for the pupil CNNs. We employed the mean absolute error (MAE) loss function and the mean ab-

solute error to assess model performance. To train the models, we used the Adam (Kingma and Ba, 2014) optimizer for the CR CNNs and the pupil CNN at 1000 Hz, while AdamW was used for the pupil CNN at 500 Hz. In the first stage, the initial learning rate was set to $1e^{-4}$, which was subsequently decreased to $1e^{-6}$ in the second stage. An exponential decay scheduler was used for the learning rate in all training regimes.

The CR CNNs at 500 Hz and 1000 Hz were trained for a maximum of 700 epochs for the first and second stages, incorporating early stopping with a patience of 40 in the first stage and 120 and 20, respectively, in the second stage. The first stage of the CR CNN at 500 Hz converged after 286 epochs while the second stage required 555 epochs. The 1000 Hz model reached convergence in 167 epochs for the first stage and 307 epochs for the second stage.

In the first stage, the pupil CNNs are allowed to train up to 500 epochs with a patience of 20. In the second stage, the 500 Hz model is trained for up to 30 epochs with a patience of 5 while the 1000 Hz model is trained for up to 100 epochs with a patience of 10. The 500 Hz model reached convergence after 99 epochs in the first stage and 25 epochs in the second stage. The 1000 Hz model achieved convergence after 88 epochs in the first stage and 36 epochs in the second stage.

In the second stage, the first convolutional layer of each model is frozen and we used an iterative approach to determine which layers to freeze. We chose to freeze the first two layers of the CR CNNs and only froze the first convolutional layer of the pupil CNNs.

4.3.1 High-Resolution Eye-Tracking Data Collection

High-resolution eye images were recorded from the first, third and last author of the current paper and one further experienced participant with the FLEX setup Nyström, Diederick C Niehorster, Andersson, Hessels, and I. T. Hooge, 2023; I. T. C. Hooge, Diederick C Niehorster, Hessels, Cleveland, et al., 2021. Eye movement data were simultaneously recorded with the EyeLink 1000 Plus (SR Research Ltd., Ottawa, Canada). The setup is shown in Figure 14. The EyeLink illuminator was used to illu-

minate the eye and create the corneal reflection used by both the EyeLink and the FLEX setups. The FLEX setup used a Basler ace acA2500-60um camera equipped with a 50-mm lens (AZURE-5022ML12M) and a near-IR long pass filter (MIDOPT LP715-37.5) that was positioned 50 cm from the participant's eyes.

Two datasets were collected using the same participants and tasks: the FLEX 1) acquired images at 1000 Hz and 2) acquired images at 500 Hz. Camera and illuminator settings for the two data sets were as follows:

1. *1000 Hz*. 8-bit images were captured at 672 x 340 pixels, with camera exposure set to 882 μs and gain to 12 dB. EyeLink illuminator power was 100%.

2. *500 Hz*. 8-bit images were captured at 896 x 600 pixels, with camera exposure set to 1876 μs and gain to 10 dB. EyeLink illuminator power was 75%.

Videos were captured with custom software that streamed the recorded frames to mp4 files using libavcodec (FFMpeg) version 5.1.1 and the libx264 h.264 encoder (preset: veryfast, crf: 17, pixel format: gray).

For both datasets, simultaneous binocular eye movement recordings were performed at 1000 Hz with an EyeLink 1000 Plus (host software 5.12) in desktop setup using the center-of-mass pupil tracking mode. The EyeLink camera sensor was located 56 cm away from the participant's eyes. To synchronize the acquisition of eye images from the FLEX with eye movement data from the EyeLink, TTL triggers were sent to the EyeLink Host computer at the onset and offset of each FLEX image recording trial. The recordings took place in a dark room with no windows.

Several tasks were shown on an Asus VG248QE monitor at 60 Hz (viewing distance 79 cm). Participants performed the following tasks while stabilized on a chin- and forehead rest:

1. Nine 1-second fixations in random order on a 3×3 grid of fixation points positioned at $h = \{-7, 0, 7\}$ deg and $v = \{-5, 0, 5\}$ deg.

2. One 30-second fixation on a point positioned at $h = 0$ deg and $v = 0$ deg while the background luminance alternated between black and white at a cycle time of 3 seconds.

3. Three 30-second fixations on points positioned at $h = \{-3.5, 0, 3.5\}$ deg and $v = 0$ deg on a middle grey background, with each position repeated 2 times.

4. Five rightward step-ramp pursuit stimuli from $h = -10$ deg to $h = 10$ deg at a speed of 2 deg/s following a 200 ms leftward step.

5. Fixations on a dot that was presented for 1 second at positions $(x, 0), x \in \{-7, -3.5, 0, 3.5, 7\}$ deg, with each position repeated 6 times.

6. Fifteen fixations in random order on a dot that was presented for 1.5 seconds at positions $h = \{-7, -3.5, 0, 3.5, 7\}$ deg and $v = \{-5, 0, 5\}$ deg, with each position repeated 6 times.

The fixation point consisted of a blue disk (1.2 deg diameter) with a red point (0.2 deg diameter) placed on its center.

The total recording time for each participant was approximately 8.5 min, resulting in a database containing approximately 437500 FLEX eye images per participant at 1000 Hz and 219300 images at 500 Hz, along with the EyeLink data

High resolution eye image analysis

Image analysis was performed frame-wise and adapted from (Nyström, Niehorster, Andersson, Hessels, and Hooge, 2023) and (Byrne, Nyström, et al., 2023) (Reported in chapter 2). In a first stage, pupil and CR centers were localized using the thresholding method. Briefly, fixed thresholds and analysis ROIs were manually set per participant to identify the pupil and CR in the images. The analysis was performed at different pupil and CR thresholds for each participant, and the threshold that resulted in the best precision pupil and CR signals were used. Using these thresholds the images were binarized and after morphological operations to fill

holes, the pupil and CR were selected based on shape and size criteria. The center of mass of these binary blobs were then computed; these will be referred to as the pupil and CR centers localized using the thresholding method. For the pupil an ellipse was furthermore fit to the binary pupil blob.

In a second stage, for both the pupil and the CR, 180×180 pixel cutouts centered on the pupil and CR center locations identified by the thresholding method were made. To determine the CR center with the CNN, as was done in (Byrne, Nyström, et al., 2023), a black circular mask with a 48-pixel radius was applied to the cutout before feeding it into the CR CNN. Similarly, before providing the pupil cutout to the pupil CNN, a middle gray elliptical mask was applied to the cutout that was 1.4 times larger than the pupil ellipse determined in stage 1. RMS-S2S precision (Holmqvist, Nyström, and Mulvey, 2012; Niehorster, Santini, et al., 2020; Niehorster, Zemblys, Beelders, et al., 2020) of the pupil and CR center locations estimated by both the thresholding and CNN methods was computed in a 200 ms window moved over the signals, after which each trial and signal's median RMS values were determined (Hooge, Niehorster, Hessels, Benjamins, et al., 2022; Hooge, Niehorster, Nyström, et al., 2018; Niehorster, Hessels, and Benjamins, 2020).

We computed calibrated gaze signals by subtracting the CR center location from the pupil center location and calibrating the resulting vector with data from the 3x3 grid of fixation points from the first task. We used second-order polynomials in x and y with first-order interactions to calibrate these P–CR signals (Cerrolaza et al., 2012; Stampe, 1993). To examine the quality of the resulting calibrated gaze data, we computed accuracy as the offset between the estimated gaze location and the target location for the gaze data from task 6, which involved repeated fixations on 15 targets. We determined the RMS-S2S precision of the calibrated gaze signals for all recorded trials in the same way as for the pupil and CR center signals, and computed the standard deviation of the signals using the same sliding window technique.

Center of mass calculations

In order to determine the center of mass or centroid of a feature, specifically the pupil within an image, we employed the following equations (Shortis, Clarke, and Short, 1994):

$$CoM_x = \sum_{j=1}^{m}\sum_{i=1}^{n} j \cdot I(i,j) / \sum_{j=1}^{m}\sum_{i=1}^{n} I(i,j) \tag{4.6}$$

$$CoM_y = \sum_{j=1}^{m}\sum_{i=1}^{n} i \cdot I(i,j) / \sum_{j=1}^{m}\sum_{i=1}^{n} I(i,j) \tag{4.7}$$

where $I(i,j)$ represents the pixel intensity value at row i and column j in an image I, and (m,n) denote the dimensions of the image.

These equations were also used to determine the pupil center location from the annotations provided in the OpenEDS 2019 dataset (Garbin et al., 2020) and OpenEDS 2020 dataset (Palmero et al., 2021).

Chapter 5

Scanpath Feature Engineering for Image Classification Models

[1]

The field of Gaze Analysis, vital in understanding cognitive behaviors through eye movement events, has grown substantially. To manage this expansive field, our research zeroes in on the scanpath tool, a critical aspect in gaze pattern analysis. The scanpath traces the eye's movement patterns, and. although machine learning methods have boosted the ability to classify these scanpaths, finding the most effective way to use scanpaths as input for supervised learning algorithms is still largely unstudied. Our research aims to fill this gap by examining how scanpaths can be best configured for these models, focusing on their representation and classification efficacy in diverse applications. Over the next two chapters we explore how feature engineering can enhance the

[1]This discussion is grounded in a co-authored manuscript Sean Anthony Byrne, Virmarie Maquiling, Adam Peter Frederick Reynolds, et al. (May 2023). "Exploring the Effects of Scanpath Feature Engineering for Supervised Image Classification Models". In: *Proc. ACM Hum.-Comput. Interact.* 7.ETRA. DOI: 10.1145/3591130. URL: https://doi.org/10.1145/3591130 which is published in Proceedings of the ACM on Human-Computer Interaction (2023) .

performance of these models in the domain of normal-form economic games and how models trained on optimally configured scanpaths outperform other traditional methods of analysis (Byrne, Reynolds, et al., 2023; Byrne, Maquiling, Reynolds, et al., 2023; Polonio, Di Guida, and Coricelli, 2015; Krol and Krol, 2017).

Gaze Analysis pertains to the high-level extraction of intricate cognitive behaviors from eye movement events. Subsequent chapters delve into a particular tool used to analysis gaze patterns, namely the scanpath. A scanpath represents the sequence of eye movements triggered by distinct tasks. It grants a deeper understanding of these intricate strategies by showcasing either their temporal, spatial attributes, or both. Evaluating scanpaths facilitates categorization based on sequence resemblances, transition occurrences, and so forth. Moreover, categorizing using scanpath traits can foresee the group or task a scanpath associates with, relying on the patterns recognized for particular groups. The classification of scanpaths has proven effective in distinguishing between experts and novices (Castner, Kuebler, Scheiter, Richter, Eder, et al., 2020; Hosp, Schultz, et al., 2021; Hosp, Yin, et al., 2021), recognizing neurological disorders (Tseng et al., 2013; Tao and Shyu, 2019; Startsev and Dorr, 2019; Crabb, Smith, and Zhu, 2014), and discerning cognitive conditions (Braunagel, Rosenstiel, and Kasneci, 2017; Lotz and Weissenberger, 2018). These instances merely touch upon the potential uses of scanpath classification. Consequently, the quest for innovative scanpath classification methods remains ongoing.

Research developments in scanpath classification methodologies have rapidly evolved in recent years, striving to stay abreast with advancements from diverse disciplines. For example, insights from computer vision, specifically saliency models, and bioinformatics. In more contemporary developments, the domain of scanpath classification has reaped significant benefits from strides made in machine learning and deep learning. Such models have elevated the capabilities of scanpath classification, enabling it to manage high-dimensional data and efficiently detect patterns (Arulkumaran et al., 2017). Nevertheless, with these advancements, the manner in which scanpaths are represented gains paramount

importance. An emerging perspective that's increasingly being adopted in scanpath classification is the realm of image classification. This investigation delves deeply into the representation of scanpaths as images, ensuring they serve as optimal feature inputs for image classification models.

Supervised image classification models have seen growing popularity due to advancements like Keras (Chollet et al., 2015) which democratizes their implementation. Improvements such as transfer learning and fine-tuning strategies address issues of limited data and extensive configuration time (Weiss, Khoshgoftaar, and Wang, 2016; Mathew, Amudha, and Sivakumari, 2020; Moolayil, Moolayil, and John, 2019; Dung et al., 2019). As a result, their adoption spans from medical fields to cybersecurity, often surpassing traditional analysis methods (Banerjee et al., 2019; Roopak, Tian, and Chambers, 2019). This widespread use has spurred innovative data representations as images, like transforming music frequencies into spectrograms for CNNs (Costa, Oliveira, and Silla Jr, 2017), converting ECG signals into images (Jun et al., 2018), and algorithms turning tabular data into visuals (Zhu et al., 2021). This broadens the reach of image classification models across various scientific domains.

Representing scanpath data as an image aligns with a 2D spatial comprehension, providing a normalized view aligned with prevalent visualization techniques like heatmaps, prominent in scanpath assessment (Isokoski, Kangas, and Majaranta, 2018; Fuhl, Kuebler, et al., 2018; Privitera and Stark, 2000). Incorporating saliency models[2] adds semantic depth, such as scene comprehension (Fuhl, Kuebler, et al., 2018; Rajashekar et al., 2008; Geisler et al., 2020). This fusion of scene semantics with attentional influences offers deep learning inputs in line with human perception (Simonyan, Vedaldi, and Zisserman, 2013). Current advancements in deep learning for image comprehension further elevate automated semantic extraction in scanpath analysis (Sugano, Ozaki, et al., 2014; Barz and Sonntag, 2016). Nevertheless, focusing image classification models on the scanpath, not the image, emphasizes attention models beyond specific image traits.

[2]Models in computer vision mirroring human visual scene processing.

The domain of eye-tracking has witnessed the evolution of image classification models, achieving exemplary performance across free-viewing and task-based viewing experiments. We direct readers to the related work section for an in-depth exploration of image classification models. While these models exhibit competitive results compared to traditional methods, we identify a literature gap regarding the formulation of input feature space when utilizing gaze data in image classification algorithms. To navigate this gap, we conduct experiments, initially creating varied scanpath images from identical data, each differing in gaze data representation. We systematically examine the incorporation of three fundamental gaze data features into the images: saccades, fixations, and Areas of Interest (AOIs). Furthermore, we explore aspects like sequential coloring of saccadic information or aggregating fixations, enhancing feature salience for kernel-based models. Subsequently, we assess the impact of feature addition on input, reporting on model accuracy and other metrics. Employing a pre-trained VGG-16 and a simplistic SVM model, we conduct model tests, comparing metrics like Accuracy, F1-score, and AUC to evaluate performance disparities across varied scanpath images. To date, there is only one scientific report that delves into feature engineering of scanpath data. (Yin et al., 2021) examine different scanpath feature engineering approaches for CNN input. They tested multiple models including a VGG model pre-trained on Imagenet, where they found that the VGG model and training regime works well for scanpath classification. Our work differs from their approach as we examine features such as saccades and fixations that are more familiar and therefore more accessible to the wider eye-tracking audience, while (Yin et al., 2021) use more data driven methods.

In this exploration of scanpath feature engineering and model evaluation, we employ data published by (Marchiori, Di Guida, and Polonio, 2021), originating from a task-based viewing experiment wherein participants engage in normal-form matrix games against a computer, which employs a strategy founded on the Nash Equilibrium. The Nash Equilibrium, a game theory concept, implies no player can gain by altering their strategy given the counterpart's known strategy (Fudenberg

and Levine, 2016). This task was transformed into a binary classification task, aiming to categorize whether or not the participant selects the Nash Equilibrium. Additional dataset details are available in methods section 7. We selected this experiment as its sparse environment ensures relatively simple scanpaths, and the association between choices and gaze behavior is well-documented (Devetag, Di Guida, and Polonio, 2016; Polonio and Coricelli, 2019; Polonio, Di Guida, and Coricelli, 2015; Li and Camerer, 2020; Li and Camerer, 2021). For decades, economists have leveraged eye-tracking studies, employing gaze data as a surrogate measurement for cognitive processes. A multitude of classification methods has been deployed to ascertain the correlation between gaze patterns and the cognitive intricacies of individuals, especially when deciphering their decision-making strategy in economic games (Knoepfle, Camerer, and Wang, 2009; Devetag, Di Guida, and Polonio, 2016; Marchiori, Di Guida, and Polonio, 2021; Krol and Krol, 2017; Li and Camerer, 2020). Initial efforts gravitated towards methods like logistic regression or cluster analysis of fixation points (Polonio, Di Guida, and Coricelli, 2015). However, contemporary research has capitalized on the advancements of machine learning. For instance, (Li and Camerer, 2020) utilized a Salience Attentive Model (SAM) with saliency maps, aiming to classify equilibrium choices in two-by-two normal-form games. Concurrently, (Krol and Krol, 2017) employed a Multilayer Perceptron (MLP) to discern if a game played by a participant was either ”predictable” or ”unpredictable.”.

5.1 Evaluation Criteria

Our work underscores the significance of leveraging traditional eye-tracking features to enhance classification accuracy in supervised image classification models. By integrating standard model architectures with feature engineering rooted in scanpath domain knowledge, we aim to make these models more accessible and interpretable, especially for eye-tracking researchers. The emphasis on familiar scanpath features provides a tangible layer of model explainability. Our primary contributions center on:

1. We demonstrate that easy to implement transfer learning strategies can be applied successfully to gaze data. This finding is important because it shows that high classification accuracies can still be achieved without complex, specialized training regimes or custom architectures.

2. We suggest that the traditional features in the eye-tracking literature are useful priors in the realm machine learning based scanpath classification.

3. Our results hold for both deep learning and machine learning methods suggesting robustness in our scanpath design approach.

4. The feature engineering strategy we employ can be easily understood and replicated across many other datasets and experiments by eye-tracking researchers wishing to use image classification models.

5.1.1 The Dataset

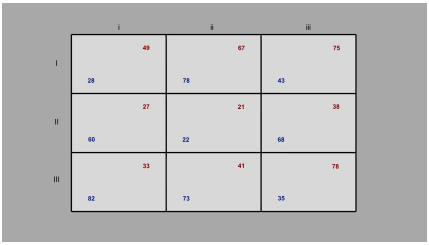


Figure 19: An example of a gameboard used in the experiment by (Marchiori, Di Guida, and Polonio, 2021). The payoffs of the participant are in blue, the payoffs of the other player, which is a computer algorithm, are in red. As the participants are made aware that the computer will always select a choice consistent with the Nash Equilibrium, it stands that any choice the participant makes that is not consistent with the Nash Equilibrium will lead to a sub-optimal outcome. In order for the participants to maximise their payoff, they must perform a complex visual search across the gameboard to find the Nash Equilibrium which is located at position [Row 2, Column 3].

To test the effects of using different scanpath designs, we use data from (Marchiori, Di Guida, and Polonio, 2021). The data was recorded during a behavioral experiment where the eye movements of participants are captured while playing a set of economic games. The data contains 243 participants (81 males, mean age = 24.1, SD age = 4.6). The participants played a set of two-person 3x3 matrix games presented in normal-form against a computer programmed to always play the action consistent with an optimal strategy that in game theory is named the Nash Equilibrium strategy (Nash Jr, 1950). Participants were informed that the computer would always play rationally and try to maximize its own payoff. The payoffs of both player and computer in the game matrix were presented in different colors to facilitate comprehension. To identify the Nash Equilibrium in the selected games, participants must use what is known in behavioral game theory as strategic sophistication (Polonio, Di Guida, and Coricelli, 2015), which is the attempt to predict the other's decisions by taking their own incentives into account and best respond to it (Costa-Gomes, Crawford, and Broseta, 2001). The Nash Equilibrium action was randomized across the games in an effort to avoid any patterns which could be identified by the participants.

During the experiment, the participants played 15 independent games. For the current study, we model scanpaths using ten of these games to generate a total of 2430 scanpaths. We did not consider the other five games because they did not require strategic sophistication and the participants could play optimally without considering the other player's incentives. As this is a supervised classification task, the physical choices made by the participant using a keyboard are used as the labels in our experiments. The participants used the keyboard to select one of the three rows each game knowing that the payoff they would receive would depend on the selection made by the computer. For instance, in Figure 19 the highest payoff for the participant is located in row three. However, this payoff should be seen as unattainable by the participant as they know that a rational opponent will never select the first column. In order to best respond to the actions of the computer knowing that it will play rationally, the participant should choose row two under the assumption

that the computer will choose column three. In this experiment, only one Nash Equilibrium exists in each game. To frame this experiment as a binary classification problem, we label if the participant selects the Nash Equilibrium choice as one class and any other choice as the second class. The experiment was conducted at the Experimental Psychology Laboratory of the University of Trento (Italy) and lasted around one hour.

5.1.2 Gaze Data and Scanpath Creation

The gaze data was garnered with a sampling rate of 1000 HZ, facilitated by the Eyelink 1000 tower mount (SR research, Ontario, Canada). The raw gaze recordings underwent a transformation into scanpath visuals via the PyGaze library (Dalmaijer, Mathôt, and Stigchel, 2013). Methodically crafting the scanpath sets, we amalgamated various combinations of prevalent eye-tracking attributes, specifically saccades, fixations, and AOIs, into each image iteration. Central to our methodology was the hypothesis postulating enhanced classification accuracy with the incorporation of expansive gaze details as image features. Moreover, we delved into strategies poised to accentuate the prominence of gaze attributes for image classifiers, encompassing distinct color schemes and fixation aggregation techniques. Subsequent sections elucidate the representations of saccades and fixations within the images, and our endeavors to augment the saliency of these features for image-based classifiers. Collectively, our examinations can be categorized into four distinct domains.

The first category consists of the the baseline cases (see Fig. 20, Category 1). We compare our results against our two baseline cases, both of which involve simply plotting each recorded gaze-point. For one baseline set, we plot the raw gaze data as white colored dots onto a black background (Fig. 20 ii.). For the other baseline set, we test the effect of the game environment by illustrating the gaze as green colored dots over the gameboard (Fig. 20 iii.); we chose green as it does not occur anywhere on the background. To avoid running too many redundant experiments, we exclude the gameboards from all other scanpath design except in one test where we remodel the best performing scanpath de-

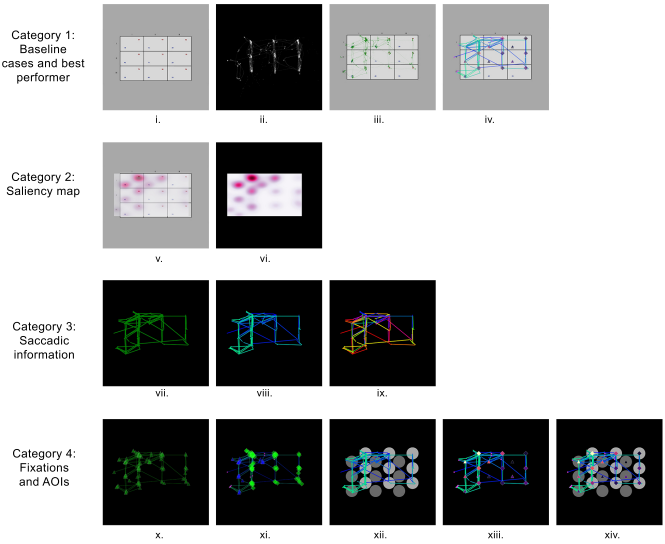


Figure 20: Scanpath sets arranged by incorporated visual data. Category 1 (from left to right): the empty gameboard serving as the stimulus in the experiment, the raw gaze data, the raw gaze data overlayed on the gameboard image, the best performing scanpath set overlayed on the gameboard image. Category 2: a saliency map overlayed on the gameboard image, a simple saliency map with a black background. Category 3: saccadic information, sequentially-colored saccadic information, non-sequentially-colored saccadic information. Category 4: Non-aggregated fixations with saccades (with uniform shape and color), non-aggregated fixations with sequentially-colored saccades, sequentially-colored saccades over AOIs, sequentially-colored saccades with aggregated fixations, sequentially-colored saccades 5and aggregated fixations over AOIs.

sign with the gameboard background included. It should be noted that we changed the opacity of the scanpath so that the gameboard is visible in this trial.

In our second category, we tested different ways of representing the fixation data as saliency maps. While saliency maps are not scanpaths, as they do not contain any temporal dimension regarding the gaze data, they have been used in the past to represent gaze behavior as an image

(Li and Camerer, 2020; Fuhl, Kuebler, et al., 2018). Saliency maps are created by plotting the density of fixations across an image. Since they are technically a different class of image, we decided to plot them with and without the game background for completeness (Fig. 20 v.-vi.).

In the third category, we tested different methods of representing saccadic information. In all cases, the saccadic information was plotted using linear saccades that occur between two fixations. In our first saccade design, we plotted the saccades in a single green color against a black background (Fig. 20 vii.). Next, we attempted to make the temporal dynamics of the saccades salient to the model by plotting the saccades using a sequential colormap from the matplotlib library (Hunter, 2007)(Fig. 20 viii.). Using this colormap, the lightness value increases monotonically, saccades formed at the start of the recording are plotted in a dark blue color and with later saccades being plotted in a light green color. To implement a counterfactual test, we also tested a scanpath design plotting the saccades using a qualitative or non-sequential colormap where the colors have no order or relationship (Fig. 20 ix.).

In the fourth category, we experimented with the representation of fixations and AOIs centered on the 18 payoffs. First, we plotted all of the fixations using a single color and shape (Fig. 20 x.), then added sequential colormapping on the saccades (Fig. 20 xi.). We also displayed saccades with temporal information where we colored just the AOIs depending on if they belonged to the participants or the computers payoffs (Fig. 20 xii.) Next, to try and control overcrowding and overlapping fixations in the scanpath design, we plotted a single shape located in the center of the AOI where the fixations occurred (Fig. 20 xiii.-xiv.). We use a triangle if the participant was fixating on a payoff that he could receive and diamond for when the participant gazes at the counterpart's payoff. In this design, we again made use of sequential colormapping; this time to count the number of times an AOI was visited. Fixations that occur outside of the AOIs are represented by fuchsia dots that are 57% the size of the fixation shapes within an AOI.

5.1.3 Model Selection and Training Regime

To reduce the possibility of a chance result, our primary experiment consists of running the VGG-16 model using 5-fold cross-validation with a 80:20 split. We set the model to run for 10 epochs each fold. We also balance the training set by under-sampling the majority class, removing a total of 28 images from the majority class. We report both the the average result across the 5-fold and the best fold for each model. We set the learning rate of 1×10^{-3} with a decay factor of 0.1. We report both the average result across folds and the best performing fold for each dataset.

As a second experiment, we split the data using a 70-20-10 train-validate-test split and ran the VGG-16 model on each set to see how it would perform on a holdout set. We split the data at the participant level to avoid any contamination leaking from the training set. Meaning, the scanpaths recorded from a given participant could only appear in a single split. During these experiments, we did not drop any of the recorded data, thus keeping a slight class imbalance, which is more indicative of real-world problems (Johnson and Khoshgoftaar, 2019; Wang et al., 2016). Additionally, we incorporated an early-stopping mechanism into the model with a patience of 3 and set the maximum amount of epochs the model could run for to 15. However, no model reached this number of epochs before the early stopping halted the training. We used the same optimiser and learning rate as in the primary experiment.

For the final stage of analysis, we compared the results of the VGG-16 model to a Support Vector Machine (SVM) image classifier. The model is created using the popular SKlearn library (Pedregosa et al., 2011). Similar to its VGG-16 counterpart above, we used a 5-fold cross-validation and focused our attention on its average results. For each scanpath set, we apply an exhaustive parameter grid search to select the values of the hyperparameters from the following options: Regularization parameter: $C = \{0.1, 10, 100\}$, Kernel coefficient: $gamma = \{10, 0.1, 0.0001\}$, and a default $Kernel = \{rbf\}$. Out of the thirteen scanpath datasets, the majority resulted in parameters $C = 10, gamma = 0.0001$, but the two variants that include only the raw gazepoints (gazeraw, gazeraw_wimg) only de-

viated at parameter $C = 100$. While the primary focus of our paper is not to compare various CNN models or training methods, but rather to investigate the impact of utilizing feature engineering on input data, we employed a ResNet-50 model on the best performing scanpath design to see if our results hold across another variety of deep learning model. We conducted a small ablation study comparing our favored transfer learning approach with frozen weights to the one with unfrozen weights and a random weight initialization. Our findings revealed that the frozen weights transfer learning approach produced comparable results to the unfrozen weights and random weight initialization approaches, while necessitating considerably less computational resources and much easier to devise and implement training regimes. Consequently, we elected to emphasize models with frozen weights from Imagenet and a transfer learning approach in this study. This approach strikes an ideal balance between computational efficiency and performance, is straightforward to set up, and may surprise many readers since the target dataset of scanpaths differs substantially from the domain dataset of Imagenet. Similar findings have been documented in the medical imaging domain in (Raghu et al., 2019), which demonstrated that models trained on Imagenet perform similarly to custom lightweight models. We argue that this approach will serve the eye-tracking community better since deploying pre-trained models necessitates less computational expertise than creating a custom model, although future research could investigate scanpath feature engineering in the context of these lightweight models.

5.2 Results

We evaluated our experiment using the following metrics: Accuracy, F1-score, and Area Under the Curve (AUC) which can be seen in the tables below. True Positive/ False Positive Rates are also reported in the form of confusion matrices, which can be seen in Figure 21. When considering all model tests across both classifiers, the scanpath design containing sequentially colored saccades and aggregated fixations stands out as the best-performing. However, the scanpath design containing sequen-

tially colored saccades and AOIs proved to be a competitive adversary throughout all the tests, with almost equivalent scores across all models and, in some tests, even outperforming the scanpath with sequentially colored saccades and aggregated fixations. We chose the former to insert a gameboard underlay, as it made for a less crowded and crisper image compared to the scanpath design which includes AOIs. The colored saccades/aggregated fixations design performed second best in our primary experiment (the VGG-16 configured for k-fold cross-validation) in terms of average accuracy, with a score of 75.98%, narrowly missing out on first place to the sequentially colored saccades with AOIs design by a margin of 0.75% . Further, it achieved an average F1-score of 75.19% and AUC of 83.62%. In terms of best performing fold in our cross-validated model, this design scored third place, losing again to colored saccades with AOIs by a narrow margin of 0.37%. The first place is won by colored saccades/aggregated fixations with a gameboard underlay by a margin of 0.83%. This design also landed on the top three when using the SVM image classifier and scored the best in terms of average accuracy across the folds with a score of 73.94%, and again landed in third place in terms of the best-performing fold, losing against saliency maps with a gameboard underlay and sequentially colored saccades with non-aggregated fixations.

In terms of incorporating the gameboard image into the design of the baseline case, it performed the worst across all tests using the pre-trained VGG-16 model with a score of 51.16% in average accuracy, 54.89% in best accuracy and 70.40% in test accuracy in the model with a train-validate-test split. Regarding the SVM, it outperformed the raw gaze data plotted on a black background, but only marginally with both sets in the bottom five worst performers. Indeed in all tests, the baseline cases of plotting the raw gaze data on a black background or gameboard always performed badly and placed within the bottom five results.

The results for the trial's saliency maps were most surprising, as they performed much better than expected in the tests with the VGG-16. In all cases, the saliency maps with the background outperformed the ones on a black background. The saliency maps with the gameboard as a

Figure 21: Confusion matrices representing the average performance of each scanpath dataset on a 5-fold cross-validated pre-trained VGG-16 model. Each row and figure number corresponds to the four categories and numbering as defined in Figure 20 excluding the simple gameboard (i) as it is not used as input for the models.

Dataset	Accuracy	AUC	F1-Score
Saccades_Temporal_AOI$^{\Delta\star}$	**0.7674**	**0.8423**	**0.7568**
Saccades_Temporal_Fixations$^{\Delta}$	0.7598	0.8362	0.7519
Saccades_Temporal$^{\Delta}$	0.7589	0.8361	0.7510
Saccades_Temporal_Fixations_with_Background$^{\Delta}$	0.7585	0.8292	0.7480
Saccades_Temporal_Fixations_AOI$^{\Delta\star}$	0.7456	0.8242	0.7343
Saccades_Temporal_Non-Aggregated_Fixations$^{\Delta}$	0.7435	0.8131	0.7336
Saccades	0.7369	0.8126	0.7273
Saliency_Map_with_Background $^{\diamond}$	0.7361	0.8064	0.7282
Raw_Gaze	0.7223	0.7952	0.7112
Saccades_Fixations_Single_Shape_Single_Color	0.7173	0.7956	0.7015
Saccades_Temporal_NonSequential_Colormap	0.7090	0.7916	0.6993
Saliency_Map	0.7049	0.7732	0.7154
Raw_Gaze_with_Background $^{\diamond}$	0.5116	0.4964	0.0036

Table 2: Average results of a five-fold cross validation on a VGG-16 model with pre-trained weights sorted from highest to lowest accuracy. Δ denotes temporal information via a sequential color map. $\star$ AOI is included. $\diamond$ With gameboard image placed under the scanpath

background scored an average accuracy of 73.61% and best accuracy of 75.88% when using the k-fold cross-validation VGG-16 model. We found this result so noteworthy, since comparable research (Li and Camerer, 2020) did not find a statistically significant results when attempting to resolve whether salience affects how often people choose the equilibrium strategy in two-by-two matrix games with a similar experimental structure. They used saliency maps plotted onto a game background as input to a Salience Attentive Model (SAM), As SAMs are usually pre-trained models that fine-tuned on open-access salience datasets such as SALICON (Jiang et al., 2015) this raises questions over how specific training regimes impact model performance when analysing eye-tracking data.

In our primary experiment, we compare the average accuracy across each fold of the VGG-16 configured for k-fold cross-validation model with each different scanpath set as input. The top four performing scanpaths all scored within one percent of each other, with the best score of 76.72%, which outperforms the baseline case of the raw gaze data with the gameboard as a background by over 25%. All top four scanpath designs contain saccades created with the temporal colormap. In all cases, a scanpath design that incorporated some fixation features in combination with temporal saccades became the best-performing model. However, it remains unclear from this experiment how to best represent these fixations because, depending on the experiment, the AOIs with aggregated and non-aggregated fixations all exhibited the highest accuracy. The full results can be seen in Table 2. Our hypothesis that using sequential colouring to helps form meaningful representation for the model is supported because the performance becomes worse when saccadic information is encoded using non-sequential colormaps. These scanpaths performed worse than the baseline raw gaze data in every test using the VGG-16 model, suggesting that convolutional filter extracts meaning from the colorschemes. However, further research is needed to confirm these findings.

Figure 21 shows the confusion matrices based on the average results of all 13 scanpath sets from the primary experiment. Here, it is clear to see that raw gaze data with background image (See Figure 20 iii.)

Dataset	Accuracy	AUC	F1-Score
Saccades_Temporal_Fixations_with_Background$^{\Delta}$	**0.7879**	**0.8590**	**0.7661**
Saccades_Temporal_AOI$^{\Delta\star}$	0.7833	0.8582	0.7792
Saccades_Temporal_Fixations$^{\Delta}$	0.7796	0.8564	0.7730
Saccades_Temporal$^{\Delta}$	0.7771	0.8533	0.7757
Saccades_Temporal_Fixations_AOI$^{\Delta\star}$	0.7713	0.8470	0.7727
Saccades_Temporal_Non-Aggregated_Fixations$^{\Delta}$	0.7692	0.8287	0.7589
Saccades	0.7651	0.8299	0.7490
Saliency_Map_with_Background $^{\diamond}$	0.7588	0.8231	0.7495
Saliency_Map	0.7542	0.8098	0.7511
Raw_Gaze	0.7354	0.8014	0.7406
Saccades_Fixations_Single_Shape_Single_Color	0.7277	0.8074	0.7265
Saccades_Temporal_NonSequential_Colormap	0.7256	0.8126	0.7295
Raw_Gaze_with_Background $^{\diamond}$	0.5489	0.5174	0.0181

Table 3: Best results of a five-fold cross validation on a VGG-16 model with pre-trained weights sorted from highest to lowest accuracy. Δ denotes temporal information via a sequential color map. $\star$ AOI is included. $\diamond$ With gameboard image placed under the scanpath

performed the worst with TPR only at 0.18%. While the raw gaze (ii), non-sequentially colored saccades (ix) and single-colored saccades with non-aggregated fixations (x) performed relatively better, they still rank lower compared to scanpath sets containing meaningful representation.

In a second experiment, we trained the VGG-16 model and tested it on a holdout set the design with temporal saccades, aggregated fixations and AOIs scored highest with an accuracy of 78.80%, AUC 87.93%, and an F1-score of 76.23%, making it an excellent classifier by all standards.

The 5-fold cross-validated SVM model generally yielded comparatively worse results than the VGG-16 variant as well as with a train-validation-test split evaluated on the VGG-16 model. The SVM shows a maximum average accuracy of 73.94% from the scanpath set that includes sequentially colored saccades and aggregated fixations and the maximum best accuracy of 76.79% from sequentially colored saccades with non-aggregated fixations. In comparison, the cross-validated VGG-16, achieved a higher average accuracy by about 3% and a marginally higher best accuracy of about 2%.

Comparing the k-fold cross-validation on the VGG-16 model and SVM

Dataset	Accuracy	AUC	F1-Score
Saccades_Temporal_Fixations_AOI	**0.7880**	**0.8792**	**0.7623**
Saccades_Temporal$^{\Delta}$	0.7880	0.8582	0.7535
Saccades_Temporal_AOI$^{\Delta\star}$	0.7760	0.8583	0.7407
Saccades_Temporal_Fixations_with_Background$^{\Delta}$	0.7680	0.8612	0.7563
saccades	0.7560	0.8317	0.7382
Saliency_Map_with_Background $^{\diamond}$	0.7440	0.8336	0.7168
Saccades_Temporal_Non-Aggregated_Fixations$^{\Delta}$	0.7400	0.8596	0.7368
Saccades_Temporal_Fixations$^{\Delta}$	0.7400	0.8719	0.7410
Saccades_Fixations_Single_Shape_Single_Color	0.7400	0.8387	0.7257
Saliency_Map	0.7360	0.7971	0.7402
Saccades_Temporal_NonSequential_Colormap	0.7280	0.8291	0.7302
Raw_Gaze	0.7160	0.7968	0.7102
Raw_Gaze_with_Background $^{\diamond}$	0.7040	0.8501	0.7176

Table 4: Results of a VGG-16 model with pre-trained weights tested on a hold-out set sorted from highest to lowest accuracy. Δ denotes temporal information via a sequential color map. $\star$ AOI is included. $\diamond$ With gameboard image placed under the scanpath

in terms of the F1-score, the scanpath set with sequentially colored saccades and AOIs is on top for VGG-16 with an average accuracy of 75.68% and best accuracy off 77.92% and remains in the top five for both the average and best results of the SVM. The dataset with sequentially colored saccades, aggregated fixations, and a gameboard background takes first place in the SVM results with an average accuracy of 73.63% and best accuracy of 76.41% while similarly still remaining in the top five from the VGG-16 results. The dataset containing sequential saccades, aggregated fixations as well as AOIs also consistently place in the top 5 for both SVM and VGG-16 tests.

Additionally, we performed a small ablation analysis on the best performing scanpath design. We used the scanpath design that was the sequentially colored saccades and aggregated fixations, to test a VGG-16 and a ResNet-50 model under various initialization conditions, such as random initialization and models starting with Imagenet weights with frozen and unfrozen layers. We followed the methodology of our second experiment, training each model and evaluating its performance on a holdout set. Although our paper primarily focuses on the impact of

feature engineering on performance, we conducted this additional analysis to emphasize that our transfer learning strategy, can be an effective training regime for scanpath images. The VGG model pretrained on Imagenet with frozen layers returned an accuracy of 0.7880. The model with random weight initialization performs almost equivalently with a score of 0.7800. When running the model with unfrozen layers meaning that model can adjust the weights and biases, we see a drop of performance decrease of 6% to 0.7280. Moving to the ResNet-50 – another popular architecture, we see less stable results, especially when moving to random weights hence further supporting our strategy choice. The following accuracies can be reported for the Resnet-50 Model 0.7680 (unfrozen), 0.6840 (frozen) and 0.6280 (random initialization). Our findings are in line with previous research, for instance, (Bhattacharya et al., 2020) used scanpaths with a comparable design and from a similar-sized dataset into multiple CNN architectures and found that the VGG architecture slightly outperformed the rest.

Dataset	**Accuracy**	**AUC**	**F1-Score**
Saccades_Temporal_Fixations	**0.7394**	**0.7390**	**0.7300**
Saliency_Map_with_Background$^{\diamond}$	0.7385	0.7398	0.7354
Saccades_Temporal_Fixations_AOI$^{\Delta\star}$	0.7381	0.7383	0.7336
Saccades_Temporal_NonAggregated_Fixations$^{\Delta}$	0.7369	0.7370	0.7318
Saccades_Fixations_Single_Shape_Single_Color	0.7369	0.7371	0.7315
Saccades_Temporal_Fixations_with_Background$^{\Delta}$	0.7294	0.7304	0.7363
Saccades_Temporal_AOI$^{\Delta\star}$	0.7273	0.7274	0.7156
Saccades_Temporal$^{\Delta}$	0.7156	0.7161	0.7027
Saccades	0.7140	0.7144	0.7051
Raw_Gaze_with_Background$^{\diamond}$	0.7115	0.7109	0.6894
Saccades_Temporal_NonSequential_Colormap$^{\Delta}$	0.7086	0.7085	0.6990
Saliency_Map	0.7073	0.7070	0.7001
Raw_Gaze	0.7069	0.7062	0.6870

Table 5: Average results of a five-fold cross validation on a simple SVM model sorted from highest to lowest accuracy. Δ denotes temporal information via a sequential color map. $\star$ AOI is included. $\diamond$ With gameboard image placed under the scanpath

5.3 Limitations and Future Research

Our study has limitations. First, the dataset we used for this study was handpicked as it provides a much sparser environment than most eye-tracking datasets meaning that the constructed features are more pronounced than they may be in a dataset that contains natural images. This sparseness may contribute to the improvements in accuracy as the image becomes fuller so to speak, as we include more eye-tracking features. Second, our list of engineered features is not exhaustive and there may well be scanpath configurations that yields better results. In future studies, we aim to test the effects of feature engineering in the design of scanpath images on multiple datasets for both free and task-based viewing in order to come up with some guiding principles as currently this work only provides the reader with an example case. Third, we did not explore all of the many different types of image classifiers such as Vision Transformers that may perform better without any feature engineering. Finally, another avenue that needs to be explored in future research involves how to best optimise a training regime that could impact the model performance on any of the given scanpath sets containing engineered features.

5.4 Concluding Remarks

Feature engineering, specifically the transformation of raw gaze data into saccades and fixations, significantly augments the efficacy of machine learning models. While prior works have affirmed the enhancement of model performance through feature engineering in other realms (Rawat and Khemchandani, 2017; Shah, Wang, and He, 2020; Jha et al., 2018), our findings consolidate the importance of traditional eye-tracking metrics in this process.

Leveraging image classifiers for eye-tracking data offers a wealth of benefits, such as circumventing issues with sequence padding and the straightforward incorporation of viewing context. With the availability of established architectures like VGG (Simonyan and Zisserman, 2015),

Resnets (He et al., 2015), and Vision Transformers (Dosovitskiy et al., 2020), researchers have an expansive architectural palette to extract optimal results. The utility of pre-training on datasets like Imagenet for scanpath classification is underlined by works like (Yin et al., 2021; Byrne, Reynolds, et al., 2023), resonating with phenomena observed in the realm of medical imaging (Shin et al., 2016).

Eye-tracking research's multidisciplinary nature signifies a vast gamut of technical proficiencies. Our emphasis is not on positing an exhaustive approach to constructing scanpaths for image classifiers, but rather, we aim to elucidate a blend of domain-specific and task-centric knowledge to assist a diverse cohort of researchers.

Chapter 6

Prediction of User Intention Using Scanpath Images

[1]

The use of eye-tracking as a means of information acquisition for automated systems is ever-increasing due to technological advances such as the ability of front-facing cameras on smartphones (Valliappan et al., 2020; Krafka et al., 2016) and laptops (Zhang et al., 2015; Papoutsaki, Laskey, and Huang, 2017) to accurately record eye-movements at scale. Other developments, including the growing presence of virtual and augmented reality devices in everyday life (Xiong et al., 2021), have also contributed to the development of eye-tracking software and devices. Further, recent advances in Machine Learning (ML) techniques have led to a significant increase in the accuracy of prediction when modelling gaze data (Kümmerer, Theis, and Bethge, 2015; Valliappan et al., 2020; Sims and Conati, 2020; Kümmerer and Bethge, 2021). ML techniques have been applied to eye-tracking data to model human cognition in a variety of settings, including – but not limited to – detecting sarcasm (Mishra and Bhattacharyya, 2018), identifying when a participant is in a state of

[1] This discussion is grounded in a co-authored manuscript Sean Anthony Byrne, Adam Peter Frederick Reynolds, et al. (2023). "Predicting choice behaviour in economic games using gaze data encoded as scanpath images". In: *Scientific Reports* 13.1, p. 4722 which is published in Nature Scienfic Reports (2023).

confusion (Sims and Conati, 2020), classifying the relevance of a passage text to a user (Bhattacharya et al., 2020), and predicting where a participant will focus their attention during location-based games (Li and Camerer, 2021). Further, humans are more frequently interacting with automated systems when engaging in strategic contexts, which is a phenomenon that has been noticed by policy makers (March, 2021; Borges et al., 2021). Taken together, these factors seem to suggest that gaze data may soon be incorporated into online automated systems that will be able to anticipate future decisions of a user based on their gaze behaviour.

Numerous studies have demonstrated that gaze patterns can be effectively used to model the decision-making process of individuals operating in strategic contexts (e.g., economic games), which can, in turn, be used to form a prediction regarding if they will choose an optimal strategy or not (Li and Camerer, 2020; Krol and Krol, 2017; Polonio, Di Guida, and Coricelli, 2015). As eye-tracking becomes more widespread in user applications, it follows that an Artificial Intelligence (AI) system will be able to incorporate gaze data in order to anticipate the choices of a human agent across a broader range of strategic contexts introducing an information asymmetry in favor of the automated system and those who control the technology. The onset of gaze-aware systems may bring many potential benefits to the user. An example of this benefit is providing a means to create early detection systems which can be used to warn users before they make sub-optimal choices or to provide, in training scenarios, adaptive feedback based on the analysis of the users' pattern of eye-movements (Kümmerer and Bethge, 2021; Van der Gijp et al., 2017; Kübler, Kasneci, and Rosenstiel, 2014).

In this chapter, we delve deeper into the intricacies of economic games, building upon the tightly controlled framework discussed in Chapter 5. Here, we delve into a method that involves modeling an individual's decision strategy by examining their eye movement patterns. Our analysis is based on the gaze data collected during an experiment in which participants engaged in a series of games frequently employed in the field of economics to investigate strategic interactions. Participants play

the games on a computer screen against an algorithm that simulates the behavior of a rational player who aims to maximize their profit by choosing the Nash equilibrium strategy, which is the optimal strategy under the assumption that the counterpart is also rational and profit-oriented. The participants are informed that the algorithm is rational and profit-oriented before the game starts. Therefore any deviation from the equilibrium solution, which assumes rationality and profit maximization from both agents, can be exclusively attributed to the participant's inability to identify the optimal strategy (Marchiori, Di Guida, and Polonio, 2021). In this setting, deviations from equilibrium can be systematically categorised as different types of sub-optimal strategies. To model the decision-making process associated with different decision strategies, we create images that express gaze patterns. These images are generated from raw eye-tracking data, which capture the temporal sequences of eye fixations and are commonly referred to as *scanpaths* (Bao and Chen, 2020; Fuhl, Bozkir, Hosp, Castner, Geisler, Santini, et al., 2019b). We consider each scanpath image as a viable proxy measurement for the temporal evolution of the decision process with which participants acquire and integrate information.

We use the scanpath images in two machine learning Classification Tasks (CTs). In CT 1, we set up a simple binary classification task aimed at determining if the participant chooses the equilibrium strategy or not. In CT 2, we classify the exact strategy of the participant from the three available options in the game (see Results Section for a detailed description of how these options relate to strategic profiles). To demonstrate a clear example of how this method can produce an information asymmetry, we expand our analysis by running a series of machine learning experiments after each CT by using the trained models to classify scanpaths created from only a subsequence of the available gaze data starting from the beginning of the recording (see Figure 22 for an overview of our approach). The results of these experiments provide evidence that the gaze patterns of participants who chose the equilibrium action, as well as actions consistent with different strategic profiles, are detectable with very little gaze data and critical to our hypothesis, using data recorded

before the participant committed to a choice in the game. This, in principle, would allow an algorithm capable of processing this information in real-time to anticipate the future choice of a player. We achieve these results by predicting the choices of participants unseen by the model during training and validation since the model has only been trained using scanpaths generated from full sequences. Using a modeling method that only requires gaze data, we provide evidence that the decision strategy of an individual is detectable independently of the specific strategic structure of the game under consideration. Moreover, we show that our approach in principle would be able to predict the decision strategy of an individual also in strategic environments that were never seen by the algorithm.

Many different classification approaches have been previously applied to gaze patterns in order to model the decision-making process of individuals and classify their decision strategy in economic games (Knoepfle, Camerer, and Wang, 2009; Polonio, Di Guida, and Coricelli, 2015; Devetag, Di Guida, and Polonio, 2016; Marchiori, Di Guida, and Polonio, 2021; Krol and Krol, 2017; Li and Camerer, 2020). We contribute to this literature by investigating if presenting the data as a scanpath image, and thus accounting for the spatiotemporal patterns of the data, provides any additional benefits to increasing the accuracy of prediction. More generally, significant work has gone into creating systems with the capability of adaptive feedback using gaze data as model input. Notable work includes distinguishing between novice and expert dentists (Kümmerer and Bethge, 2021; Castner, Kasneci, et al., 2018), a tool for training radiologists (Van der Gijp et al., 2017), and detecting drivers' state while operating a vehicle (Braunagel, Geisler, et al., 2017; Tafaj et al., 2013). We contribute to this literature concerning human scanpath classification by demonstrating that in some environments, scanpaths generated from subsequences, or in other words partial scanpaths, are sufficient for a model trained on full scanpaths to classify a user's strategy in complex strategic contexts.

Our study has three objectives. First, we aim to present a new method to analyse eye patterns recorded during economic game playing with

a higher classification accuracy when compared to traditional methods. Second, we intend to provide a clear example of gaze data being used to create an information asymmetry in favour of those that in the future might develop this technology in real-time scenarios. From a policy perspective, we hope to highlight the clear need for regulation surrounding how gaze data and, more generally, biometric data are processed and used as technology becomes rapidly more immersive, making this type of data more available to AI systems. Third, we aim to present a method of early detection via the creation of scanpath images from subsequences of the data, which may be useful to the more general literature focused on creating an adaptive online system that incorporates eye-tracking data.

6.1 Results

6.1.1 Models of Choice and Behavioral Results

The games used in this study are characterized by a unique game theoretical optimal solution known as the Nash Equilibrium (Fudenberg and Levine, 1993). We used data from participants playing with an algorithm that always selects the equilibrium strategy. At the beginning of the experiment, participants are informed that the algorithm will play rationally by trying to maximize its own payoff and that it cannot modify its strategy during the experiment nor adjust its choices to those of the human player. In all games, equilibrium play requires the participant to use strategic sophistication. Strategic sophistication is defined as an attempt to predict the behavior of the counterpart by taking its incentives into account (Costa-Gomes, Crawford, and Broseta, 2001). Equilibrium play in this type of games is commonly associated with an information search pattern focused on evaluating the incentives of the counterpart, detection of the possible presence of dominant strategies, and identification of the other player's action with the highest average payoff. Deviations from equilibrium are consistent with decision rules such as Naïve and Coordination strategies. A Naïve strategy predicts if the participant selects the action with the highest average payoff. This strategy is commonly

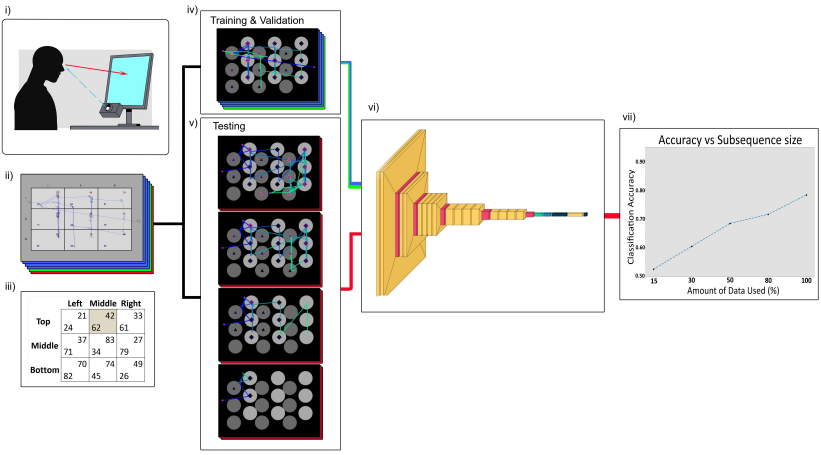


Figure 22: i.) Using a 1000 HZ tower mount eye-link, we tracked participants' gaze behavior during computer game sessions. ii.) Game visuals featured Row player payoffs in blue and Column player payoffs in red, maximally spaced for clear distinction. An illustrative raw gaze overlay emphasizes the disparity with scanpath model inputs. iii.) Representing two-player strategy in normal-form games, the Row player selects from "Top", "Middle", or "Bottom"; the Column player, from "Left", "Middle", or "Right". Their choices intersect, determining respective payoffs in a cell—bottom-left for Row, upper-right for Column. Game equilibrium is gray-highlighted. iv.) Data-wise, 70% of participants formed the training set, 20% for validation, with the remaining 10% as a hold-out test set. v.) In testing, we generated abbreviated scanpaths based on various criteria, e.g., time constraints. vi.) These scanpaths underwent evaluation in our trained model for predictive efficiency. vii.) Model accuracy served as our primary metric; findings showed minor accuracy drops despite substantial scanpath data reductions.

associated with a search pattern focused on the payoffs of the participant while the payoffs of the counterpart are barely considered. A Coordination strategy predicts if the participant selects the action consistent with the outcome that yields the largest payoff sum for the two players minimizing the difference between the two payoffs. This strategy requires the participant to compare the payoffs of the two players for each possible outcome of the game (Devetag, Di Guida, and Polonio, 2016; Polonio,

Di Guida, and Coricelli, 2015; Polonio and Coricelli, 2019; Zonca, Coricelli, and Polonio, 2019; Zonca, Coricelli, and Polonio, 2020; Marchiori, Di Guida, and Polonio, 2021).

Looking at the entire pool of data (243 participants, 2 430 choices) the proportion of choices consistent with the Equilibrium strategy is equal to 0.48, whereas the proportion of choices consistent with Naïve and Coordination strategies is equal to 0.31 and 0.21, respectively. The average response time is 15 980 ms (SD = 13 320 ms) but this value changes significantly based on the strategy used by the participant. Equilibrium choices (18 050 ms, SD = 13 200 ms) take longer than choices consistent with Naïve (13 770 ms, SD = 16 780 ms) and Coordination (15 660 ms, SD = 15 400 ms) strategies. This is supported by the results of a mixed-effects logistic regression with Equilibrium Response (1, 0) as the dependent variable, logarithmic transformation of the Response Time data as the independent variable, and Subject as Random effect (B = 0.38, p ¡ 0.001). This difference in the average response time is well documented and reflect the different level of complexity of the three strategies. Response times are not affected by the game type: on average, participants take 15 840 ms (SD = 13 380 ms) to make a decision in games that are solvable through iterated dominance and 16 120 ms (SD = 13 840 ms) in games that are not.

In terms of search patterns, equilibrium responses are characterized by a high proportion of transitions (defined as eye movements from one payoff to the next) among the payoffs of the counterpart (0.29) and a lower proportion of transitions among the payoffs of the participant (0.18). Choices consistent with the Naïve strategy are characterized by a higher proportion of transitions among the participant's payoffs (0.36) and a lower amount of transitions among the payoffs of the counterpart (0.11). Finally, choices consistent with the Coordination strategy are associated with a more balanced proportion of the two types of transitions (participant = 0.26; counterpart =0.22). The analysis of fixation times yields results that are identical to those obtained with transitions: the average amount of time participants spend looking at the payoffs of the counterpart is longer than the amount of time they spend looking at their

own payoffs when they play equilibrium (own =5 516 ms; other = 7 695 ms), shorter when they play the Naïve strategy (participant =5 996 ms; counterpart = 3 549 ms), and almost identical when they play the Coordination strategy (participant = 5 822 ms; counterpart = 5 520 ms).

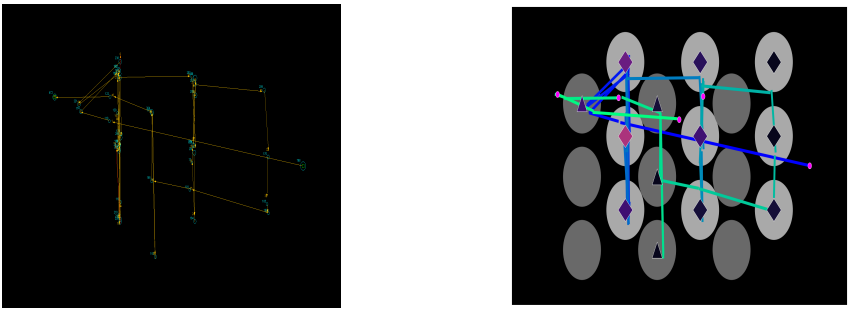
6.1.2 Modelling Cognition in Games Using Scanpaths

To untangle the participants' decision-making process we first transform the raw eye-tracking data into scanpath images. We use the following techniques to increase the salience of features that are predictive of choice behavior in the scanpath images. This is the best perfoming scanpath design from the Chapter5. These features include the temporal evolution of the visual analysis and the number of times a particular piece of information is acquired. We first define 18 Areas of interest (AOIs) centered on the matrix payoffs and distinguish between fixations that happened inside or outside of these areas. AOIs centered on player's own payoffs (payoffs the participant could receive) are light-gray circles and AOIs centered on the counterpart's payoffs are dark-gray circles. See Figure 23 for an illustration of a standard representation of eye-tracking data and how we represent recorded data into a scanpath.

When creating the scanpaths a black background was used with the AOIs superimposed on top. We provide no information regarding the game environment other than the spatial coordinates of the AOIs, as this method allows us to isolate the relationship between the information acquisition process of a player and the resulting choice without taking into account any feature of the game structure. This is important because often the last fixations of a player fall on the expected outcome for the player, or on the label of the chosen option. We believe that this would be the primary source of information used by the model to make its prediction. Therefore, we hypothesized that providing information relating to the last fixation made on the game matrix would facilitate the prediction of the model when an image with the full visual pattern is available. However, when the images are generated from subsequences, where the last fixations are missing, this task would be much more difficult.

To prevent overcrowding in the scanpath design, a single shape located in the center of each AOI represents the fixations occurring within each AOI. We use a triangle for own payoff and a diamond for when the participant gazes at the counterpart's payoff. We use changes in colour from a perceptually uniform sequential colour map to represent when multiple fixations occurred within an AOI; this means that the lightness value increases monotonically throughout the colourmaps (Hunter, 2007). For example, if there was a single fixation in an AOI, the colour of the fixation shape would be black, and if there were ten fixations in the area of interest, the colour of the shape would be pink, following the "Magma" colour map, as illustrated in Figure 24g. All fixations outside of the AOIs are represented by Fuchsia Dots that are 57% the size of the fixation shapes within an AOI. We use linear saccades to represent the transitions between consecutive fixations taking the direct linear distance between fixation points. To capture the temporal evolution of the decision process, we colour-code each saccade. Earlier saccades are coloured a dark blue, RGB (0, 0, 255), and later saccades are coloured a light green RGB (0, 1, 254.5) as illustrated in Figure 24b. The saccades were designed using start-screen position (pixel), end-screen position (pixel), start-time of the saccade (ms), and end-time of the saccade (ms). This was the "best" performing scanpath desing from the previous chapter.

To create the sets of partial scanpaths we use the following methods. First, we create scanpaths using a percentage of the total gaze at the following intervals 15%, 30%, 50% and 80%, See Figure 3 (24c, 24d, 24e,24f) for an example of how partial scanpaths at each percentage interval compares to the participant's full scanpath as illustrated in Figure 24a. This method ensures that all the test scanpaths are equally reduced but with the drawback of needing to be calculated post-hoc. Second, partial scanpaths were created based on the length of time (2s, 5s, 10s and 15s). While this method is a more realistic representation of an online setting, it comes with a disadvantage that the amount of time it takes each participant to play a single game varies. This problem leads to situations occurring, such as when a participant finishes in a given game within the first 8 seconds, the partial scanpaths in the 10 seconds and 15



(a) (b)

Figure 23: 23a.) A prototypical representation of a scanpath displaying fixation locations, fixation duration, and saccades. 23b.) An example of how we represent the data using scanpaths to increase the salience of information to the models. The circles represent the location of the payoffs for both the participant (light grey) and the opponent (dark grey). We use sequential colourmaps to represent the temporal evolution of the linear saccades and to display information regarding fixations to the model.

seconds sets are identical and are indeed full scanpaths. This feature of creating partial scanpaths by the length of time makes an overall comparison between the experiments difficult. See Figure 3 (24h, 24i, 24j,24k) for an example of how partial scanpaths created at each recorded time interval for a given participant compares to the participant's full scanpath as illustrated in Figure 24a. The average time taken per trial was 16 seconds, with the median time per trial being 11 seconds. To minimise the situation where test scanpaths are identical across test sets, the timings were chosen to be below the average time taken per trial.

6.1.3 The Dataset

Using the same dataset as in the previous chapter, we use data from 243 individuals, with each participant playing 10 games, leading to a total of 2 430 scanpaths which we use as input to the model. We do not discard any of the 2 430 scanpaths from our study. We implement an approximate 70-20-10 train-validate-test split at the participant level. Previous

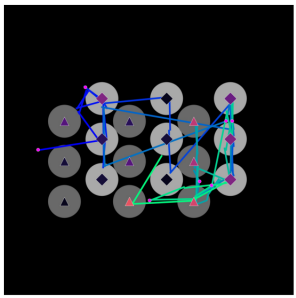
studies have shown that both the strategy and visual analysis used by the same participant are consistent in different games. We observed this consistency in our data (see Supplementary Table 4), therefore we decided to split the data at the participant level. Moreover, we hypothesized that scanpaths recorded from the same participant and game type could cause an accidental leakage of signal across the training and testing sets. For an even comparison, we use the same data split for each experiment across our analysis, as we hypothesized that some participants would create harder-to-classify scanpaths. 1 700 scanpaths were randomly placed into the training set, 480 scanpaths into the validation set, and 250 scanpaths were placed into the test set.

	VGG-19 (CT 1)	VGG-19 (CT 2)	SVM (CT 1)	SVM (CT 2)
Accuracy	**0.783**	**0.648**	0.728	0.612
AUC	**0.824**	**0.78**	0.734	0.682
F1- Score*	**0.757**	**0.616**	0.727	0.575
Percentage Accuracies				
80% Percent	**0.716**	**0.648**	0.696	0.596
50% Percent	0.684	0.564	**0.704**	**0.592**
30% Percent	0.604	0.436	**0.608**	**0.476**
15% Percent	0.524	0.328	**0.576**	**0.376**
Time Cut Off Accuracies				
15 seconds	**0.776**	**0.64**	0.756	0.616
10 seconds	**0.728**	0.604	0.696	**0.620**
5 seconds	**0.684**	**0.544**	0.652	0.524
2 seconds	**0.608**	0.32	0.592	**0.384**

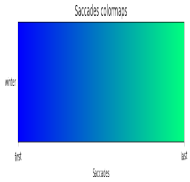
Table 6: * For Classification Task 2, weighted average metrics are reported. Total number of test observations = 250. The best result of each trial is highlighted in bold.

6.1.4 Model Selection, Tasks and Performance Metrics

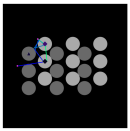
After generating the scanpaths we use the resulting images as model input into a VGG-19 model pre-trained on the Imagenet dataset. VGG-19



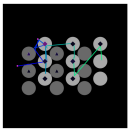
(a)



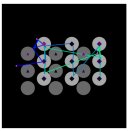
(b)



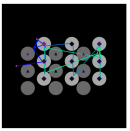
(c)



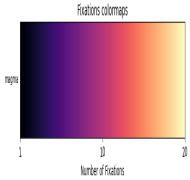
(d)



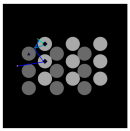
(e)



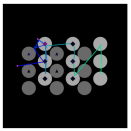
(f)



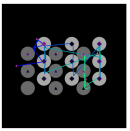
(g)



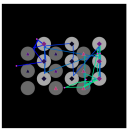
(h)



(i)



(j)



(k)

Figure 24: Scanpaths generated from the full sequence and subsequences of the data from one participant in a single game. In total, there are 8 sets of test scanpaths made subsequences. 24a.) Example of a full image, 24b.) The colour-map used for saccades. The left side would correspond to colours of earlier saccades with the right side corresponding to later saccades. 24c, 24d, 24e, 24f.) Images generated from subsequences stemming from the same participant-game at increasing percentage intervals. 24g.) The colourmap chosen for fixations. The upward threshold of 20 fixations was chosen because 99% of the area of interest across trials across participants had 20 fixations or less. 24h, 24i, 24j, 24k.) Images generated from subsequences stemming from the same participant-game at increasing time intervals.

is a variant of the VGG architecture of convolutional neural networks consisting of 16 convolutional layers and is a standard model architecture used in computer vision tasks due to excellent performance that was conducive to its success in the 2014 Imagenet Challenge (Simonyan and Zisserman, 2015). Much like the VGG-16 used in the previous chapter, the VGG-19 architecture was chosen as it is a relatively shallow network with multiple small kernels which we hypothesised would be optimal for capturing any nuanced differences between the input images. Our hypothesis stems from results in an earlier paper that tests similarly a scanpath design on a wide range of out-of-the-box neural network models during a reading task (Bhattacharya et al., 2020). To further validate our preferred VGG-19 method, we also report the results of two benchmark cases: a Support Vector Machine (SVM) configured for image classification, commonly used in scanpath classification tasks and a logistic regression model which is the most common method of traditional analysis to test the link between gaze data and choice behavior in games presented in normal-form.

We configure all the models employed for two independent Machine Learning Classification Tasks (CTs). CT 1 is a binary classification task, where the models are tasked with detecting if a participant selects Nash-Equilibrium or not. CT 2 is a multi-class classification problem where the models are asked to identify the exact strategy profile of the participant (Naïve, Coordination, Nash-Equilibrium). Importantly, the equilibrium location change in different games and each decision strategy is independent of the three available actions (”Top”, ”Middle”, ”Bottom”). A full description of the games, the spatial location of the equilibrium and of the three different strategies are available in Figure 2 of the Supplementary Material. A convenient feature of our dataset is that in CT 1 the class labels are almost naturally balanced, making it easier to train the model (Johnson and Khoshgoftaar, 2019). As an initial preprocessing step in CT 1, we create a fully balanced dataset by randomly under-sampling the majority class, removing a total of 46 scanpaths (corresponding to 1.89% of the data). In CT 2, the labels create an imbalanced dataset with almost double the responses in the majority class (Nash-

Equilibrium) compared to the other two classes. This second CT allows us to test our approach on a small imbalanced dataset, which is more indicative of real-world problems (Johnson and Khoshgoftaar, 2019; Wang et al., 2016; Johnson and Khoshgoftaar, 2019).

We assess model performance using a set of widely used accuracy measurements including Accuracy, Area Under the Curve (AUC), and F1-Score. Model accuracy is a metric that reflects the proportion of correct predictions made by a model and is calculated by the number of correct predictions divided by the total number of predictions made by the model. F1 score is a metric that combines both precision and recall, with higher scores indicating better performance. Precision is a measure of the accuracy of a model when it correctly predicts a positive outcome, while recall is a measure of the ability of a model to find all of the positive cases within the dataset. The area under the curve (AUC) measures the ability of a classifier to distinguish between positive and negative classes. It does this by comparing the true positive rate (TPR) against the false positive rate (FPR) at various classification thresholds. Like the F1 score, AUC ranges from 0 to 1, with a higher value indicating a better classifier (Pedregosa et al., 2011). We also present a full table of results for the logit model, along with confusion matrices for all models in the Supplementary Material.

In CT 1, the VGG-19 model achieves a test accuracy well above chance with a classification accuracy of 78% on the test set. The VGG-19 model can be considered an excellent discriminator between the two classes, with an AUC score of 0.8239 on the test set. Moving to the SVM, we observe a lower accuracy score (73%) and a lower AUC (0.7380). Logistic regression provides the worst performance with an accuracy of 67% and an AUC of 0.6796. When comparing the two image classification models, the VGG-19 achieves an F1-score of 0.7567 compared to 0.7265 from the SVM classifier. When analysing data from the test set during CT 2, we observe a similar pattern of results. We found that, in both cases, the test accuracy was well above chance, with the VGG-19 achieving an accuracy score of 65% and SVM scoring 61%, note that the weighted Average Area Under the Curve (wAUC) is reported for CT 2, where we can report

wAUC scores of 0.7800 and 0.7114, respectively. Logistic regression falls short again with a performance of 0.5787 (wAUC) and an accuracy of 60%. Table 6 depicts the results for all the performance metrics for both CTs for VGG-19 and SVM. A full table of results for the logit model (Supplementary Tables 1-3) along with confusion matrices (Supplementary Figures 3-8) for all models are reported in the Supplementary Material.

When evaluating our trained models' ability to classify partial scanpaths, we note the trend of a small decrease in model accuracy relative to the amount of data removed from the analysis, as illustrated for CT 1 in Figure 25 below. Results for CT 2 are similar and reported in Supplementary Figure 1 in the Supplementary Material. The results suggest that the gaze behaviour of different strategies forms distinguishable patterns early on in the visual search process despite the stochastic nature of inter-participant eye movements during game play. In CT 1, during model experiments using the sets of scanpaths created from a percentage of the total recorded sequence, we observe model accuracies at the following intervals: 80% interval (71% Accuracy, 0.7799 AUC), 50% interval (68% Accuracy, 0.7486 AUC), 30% interval (60% Accuracy, 0.6338 AUC), and the 15% interval (52% Accuracy, 0.4978 AUC). To put these findings into perspective, we observe a reduction of only 10% Accuracy when comparing the 50% interval to full scanpaths from the test set and perform better than chance from the 30% interval on-wards. During the experiments where arbitrary time points were formed, we observe a model accuracy of 72.8% with the VGG-19 model when using 10 seconds as the cutoff point, a reduction of close to 5%.

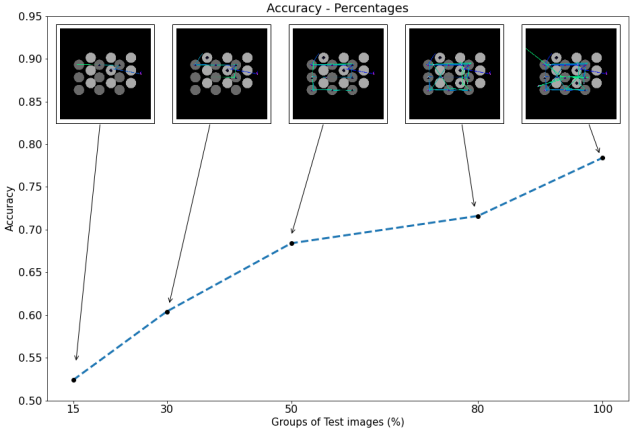
We are able to report a similar trend for the SVM classifier: 80% interval (69% Accuracy, 0.6925 AUC), 50% interval (70% Accuracy, 0.6882 AUC), 30% interval (60% Accuracy, 0.5576 AUC), 15% interval (58% Accuracy, 0.5074 AUC). We note that with very little data in both the 2-second trial and the 15 percent trial the SVM outperforms the VGG-19 model in terms of accuracy. While it is out of the scope of this paper, it remains an open question as to why this occurs.

We observed similar results during CT 2 where we found a 9% decrease in model accuracy when using the scanpaths created with 50%

of the data and VGG-19 model. Due to the class imbalance, the small sample size of our data, and more nuanced differences between playing types, CT 2 is a much harder classification task for both models. During CT 2, when conducting the model experiments using the sets of scanpaths created from a percentage of the total recorded sequence, we observe model accuracies at the following intervals for the VGG-19 model: 80% interval (65% Accuracy, 0.7569 wAUC), 50% interval (56% Accuracy, 0.7141 wAUC), 30% interval (44% Accuracy, 0.6489 wAUC), and the 15% interval (33% Accuracy, 0.5406 wAUC). Again, we are able to report similar results of the SVM outperforming the VGG model when using shorter subsequences for the SVM observing model accuracy at the following intervals: 80% interval (59% Accuracy, 0.6714 wAUC), 50% interval (59% Accuracy, 0.6761 wAUC), 30% interval (48% Accuracy, 0.6069 wAUC), and the 15% interval (37% Accuracy, 0.5343 wAUC).

6.1.5 Interpretation of Model Predictions

We aim to identify the features that contribute most significantly to the confidence of the model's predictions in the two classification tasks (CT 1 and CT 2). To do so, we evaluate the relationship between the model's Confidence Estimation (CE) and three features expressing the type and amount of information acquired: (1) the proportion of fixations on the other player's payoffs, (2) the proportion of transitions between the other player's payoffs, and (3) the natural logarithm of the response time as a proxy for the total amount of information acquired by the participant. Confidence estimation (CE) involves assessing the level of certainty that a deep learning model has in its own predictions, which is commonly achieved via computing a probability score (Hendrycks and Gimpel, 2016). We expect the CE to be directly linked to the characteristics of the visual analysis and only partially linked to the amount of information acquired (RT). In CT 1, we find that the CE of the model is positively correlated with the proportion of fixations on the other player's payoffs (Pearson's r = 0.76, p ¡ 0.001) and with the proportion of transitions between the other player's payoffs (Pearson's r = 0.66, p ¡ 0.001). We also



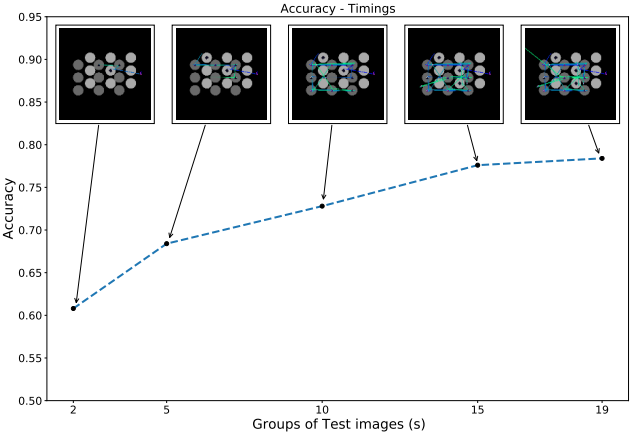


Figure 25: Accuracy of VGG-19 model in CT 1 using subsequences via percentages (25a) and time points (25b).

find a significant, yet weaker, correlation between CE and response times (Pearson's r = 0.31, p ¡ 0.001).

In CT 2, we observe similar results when the model predicts a choice consistent with the equilibrium strategy: the CE of the model is positively correlated with both the proportion of transitions between the other player's payoffs (Pearson's r = 0.44, p ¡ 0.001) and the proportion of fixations on the other player's payoffs (Pearson's r = 0.37, p ¡ 0.001). However, we do not find a positive relationship between response times and CE in this case (Pearson's r = -0.16, p = 0.06). When the model predicts a choice consistent with the Naïve strategy, we find that the proportion of transitions between the other player's payoffs is negatively correlated with CE (Pearson's r = -0.36, p ¡ 0.001), while the proportion of transitions between the participant's payoffs is positively correlated with CE (Pearson's r = 0.40, p ¡ 0.001). No relationship is observed between response times and CE in this case (Pearson's r = -0.06, p = 0.56). Finally, we find that none of the features examined is correlated with CE when the model predicts a choice consistent with the Coordination strategy. This is in line with previous results showing that participants using a Coordination strategy devote the same amount of attention (expressed in terms of the proportion of fixations and transitions) to the incentives of the two players. Overall, our results are consistent with previous findings and suggest that the level of attention given to the incentives of the other player is a predictive factor in the decision-making strategy used (Coricelli, Polonio, and Vostroknutov, 2020).

Finally, we test the hypothesis that the model makes better predictions in some games compared to other games and that the model is more accurate in predicting one strategy than another. We test the first hypothesis in CT 1 by running a mixed-effect logistic regression with accuracy of the model as dependent variable, game type (dominant solvable games and games that do not contain a dominant strategy) as independent variable, and Subject as random effect. Results of the model show no effects of the game type (B = 0.49, p = 0.12). We test the second hypothesis in CT 2 by running another mixed effect logistic regression analysis with the accuracy of the model as dependent variable, the type of strategy

used by the participant (Equilibrium, Naïve, and Coordination) as independent variable, and Subject as random effect. Results show that the model is more efficient in predicting choices consistent with equilibrium strategy (Mean = 0.86) than choices consistent with Naïve (Mean = 0.67; B = -1.11, p = 0.004) or Coordination (Mean = 0.20; B = -3.24, p ¡ 0.001) strategies. In particular, results show that the model underestimates the number of times participants choose in accordance with "Coordination" strategy (only 5% of the time compared to 18% of choices observed). This may be due to the fact that the visual analysis required to implement the Coordination strategy can also be used to implement different strategies (Polonio, Di Guida, and Coricelli, 2015).

6.2 Concluding Remarks

In this work, we were able to successfully represent the cognitive process of participants' playing economic games by transforming the recorded gaze data into scanpath images thus accounting for the spatiotemporal sequence within the data. This post-processing approach enables machine learning methods to accurately classify subsequences of the data from new participants using a model trained on only full sequences stemming from the training set. We were surprised by the large amount of data we could remove to form a subsequence and the resulting relatively small decrease in accuracy of prediction, further confirming that distinguishable patterns are formed very early on in the decision process (Polonio, Di Guida, and Coricelli, 2015; Polonio and Coricelli, 2019).

Generating scanpaths from the data seems to come with certain advantages when analyzing games presented in normal form. In one of the most comparable studies (Li and Camerer, 2020) the authors were unable to find a statistically significant relationship when investigating if they could identify the equilibrium choice in two by two games using saliency maps as input to a Salience Attentive Model (SAM). What makes this comparison so strong is that the games the authors used were created with the same experimental design as in our experiment. Our hypothesis as to why the authors of this study were unable to identify

equilibrium choices using a saliency model is due to the loss of temporal information that occurs when creating saliency maps as fixations tend to be aggregated over the temporal dimension (Assens et al., 2018). In a separate study (Krol and Krol, 2017) a Multilayer Perceptron (MLP) was used with the following variables as input: reaction time, gaze dispersion and pupil deviation. The aim of the model was to detect whether a single trial is of the "predictable" or "unpredictable" type. The authors were able to classify these types with an accuracy of 67%, which is the most comparable analysis between the two papers. That being said, the two studies we compare our approach against were conducted in the context of two-by-two games which are characterized by a lower relational complexity of the payoff structure (Zonca, Coricelli, and Polonio, 2020), which may have effects on the classification difficulty of the patterns generated from recorded eye-movements.

When examining traditional modelling methods applied to players' type classification, including cluster analysis and mixed model approaches do not reach the same level of accuracy observed with our approach (Costa-Gomes, Crawford, and Broseta, 2001; Costa-Gomes and Weizsäcker, 2008; Polonio, Di Guida, and Coricelli, 2015; Devetag, Di Guida, and Polonio, 2016; Polonio and Coricelli, 2019). Further, in these studies the accuracy of the models strongly depends on the strategic environment under consideration and the complexity of the decision rule used by the decision-maker. On average, in the contest of three-by-three games, the highest level of accuracy reported is around 68%, (Polonio and Coricelli, 2019). Results from our baseline regression model are in line with these results achieving an accuracy of 67%, an 11% deficit when compared to our augmented VGG-19 model.

Much like in the field of vision research, experimental economists are searching for methods that can be used to predict across domains (Hargreaves Heap, Rojo Arjona, and Sugden, 2014; Li and Camerer, 2021). We are especially keen to use our scanpath design in different games to investigate the stability of levels of cognition across strategic environments of different types and complexity. Future research could simply attempt to classify scanpath images created from games with different

structures using the fully trained model presented in this experiment, or build more complicated ML models to analyse the data. To date, limited research has been conducted which incorporates ML techniques, into economic games with a few notable exceptions (Krol and Krol, 2017; Li and Camerer, 2020; Li and Camerer, 2021). We hope to continue to see developments of new analysis types in this field, as historically game paradigms have provided a much needed laboratory setting to study cognition in strategic environments which enables developments in theory that are then used to explain patterns in naturally occurring data (Camerer and Ho, 2015; Crawford, 1997).

Outside of economic games, our method contributes towards the creation of interactive systems that incorporate gaze data (Castner, Kuebler, Scheiter, Richter, Eder, et al., 2020). Considerable resources have already been allocated to incorporating eye-tracking research into website design (Nielsen and Pernice, 2009), making the possibility of using a similar approach in more general domains by crafting AOIs into websites a priory quite a feasible task. We can envisage our approach being adapted to contexts such as shopping websites, online games played using virtual reality headsets, and in other strategic contexts where the cognitive state of the user is relevant such as stock trading, where this approach could be adapted to create a cooling off system to prevent traders from over-trading when losses start to occur.

Scanpath analysis is receiving increased recent attention due to the rise of Virtual and Augmented Reality systems (Rai, Le Callet, and Guillotel, 2017). While we made use of a VGG-19 pre-trained on the imagenet dataset largely for reasons of convenience, a number of deep learning techniques could be used to drastically improve the performance of our model (Assens et al., 2018). Additionally, it remains an open question outside the scope of this paper as to why the SVM outperforms the VGG-19 model when minimal data is available. Outside of model selection, there are a few limitations to our study. First, our study suffers from a small sample size, for a study that uses ML methods. Second, there is a quite serious concern surrounding the generalizability of our experiment. It is widely accepted that task semantics greatly affect gaze

behaviour (Dewhurst et al., 2018), it remains uncertain if our method of using subsequences to model cognition would achieve the same result in more noisy and complex environments outside of a laboratory setting (Assens Reina et al., 2017). While we are encouraged by results from previous studies (Kübler et al., 2017; Kübler, Kasneci, and Rosenstiel, 2014) which demonstrate the ability to classify subsequences of eye-tracking significantly above chance in a variety of settings, and in one instance using a similar kernel-based method, this remains an open question. A promising area of research involves applying transformer models like BERT (Devlin et al., 2018) directly to raw eye-tracking data instead of generating scanpath images. The benefit of using a model like BERT lies in its pre-training on a vast corpus of tokens and its large number of parameters, which endow it with zero-shot capabilities beyond those of simpler models like VGG. Such an approach has been explored by (Unger, Wedel, and Tuzhilin, 2023), who investigated its application in the complex and more realistic scenario of purchasing a computer with multiple attributes. The potential for employing this model in even more realistic and possibly real-time settings is an intriguing question that remains unanswered. Further, cognitive strategies in real-world settings will most probably be more nuanced and therefore harder to differentiate and classify using our approach.

From a policy perspective, our work demonstrates that gaze data can be used to provide an information advantage to AI-endowed players in strategic settings. As with all technology, it is not hard to imagine cases where eye-tracking data can be used to produce both good and bad outcomes for society. Trends such as the increase in the number of workers participating in the gig economy via smartphone apps (Amankwah-Amoah et al., 2021), the increased adoption of VR headsets, and the integration of eye-tracking data into online training and adaptive feedback systems, make it easy to envisage a world where eye-tracking data affect everyday life for a large percentage of the population. Our work here is just the tip of the iceberg so to speak, projects with a larger sample size could easily analyze gaze patterns at the participant level to customize user-based strategies. Much consideration and awareness of potential is-

sues are needed going forward if eye-tracking data is to be successfully regulated as it becomes part of everyday life.

Chapter 7

Scanpath Methods & Models

[1]

7.0.1 Experimental Procedure

We use data from the entire pool of participants (243) who took part in the assessment stage of Marchiori et al.(2021) (Marchiori, Di Guida, and Polonio, 2021) experiment (81 males, mean age = 24.1, SD age = 4.6). In the assessment stage, the authors assessed the initial level of strategic sophistication of each participant, and no feedback was provided to them. During this initial phase, each participant played five DO games and five

[1]This chapter contains all the technical details needed to recreate the results mentioned in chapters Two and Three. These methods are also stated in the following papers: 1. Sean Anthony Byrne, Virmarie Maquiling, Adam Peter Frederick Reynolds, et al. (May 2023). "Exploring the Effects of Scanpath Feature Engineering for Supervised Image Classification Models". In: *Proc. ACM Hum.-Comput. Interact.* 7.ETRA. DOI: 10.1145/3591130. URL: https://doi.org/10.1145/3591130 2. Sean Anthony Byrne, Adam Peter Frederick Reynolds, et al. (2023). "Predicting choice behaviour in economic games using gaze data encoded as scanpath images". In: *Scientific Reports* 13.1, p. 4722. Further all code and data needed to create these methods can be found at the following links: https://github.com/vbmaq/ImageMaker & https://github.com/seanbyrne226/Choice-Behaviour-Machine-Learning-Modeling-of-Scanpath-Subsequences & https://osf.io/fhmjy This Chapter also contains the supplementary material results from Chapter 6.

UE games for a total of 2430 games played.

They played a set of two-person 3x3 games presented in normal form against a computer programmed to always play the action consistent with the equilibrium strategy. Participants were informed that the computer would always play rationally and try to maximize its own payoff. To facilitate comprehension, the payoffs of the two players in the game matrix were presented in different colours (see Figure 22). Participants were not subject to a time constraint during the experimental task. At the end of the experiment, three trials were randomly chosen, and the participant was paid based on the outcome of the selected games. The experiment was conducted at the Experimental Psychology Laboratory of the University of Trento (Italy). The assessment stage of the experiment lasted about 15 minutes. The entire experiment lasted about 1 hour. The study was approved by the Human Research Ethics Committee of the University of Trento (protocol title: "Transfer learning within and between brains"). All participants gave informed consent. All experiments were performed in accordance with the relevant guidelines and regulations. Informed consent was obtained from all participants before the start of the experiment.

7.0.2 Eye-tracking Recording

Eye movements were monitored and recorded using an Eyelink 1000 tower mount (SR research, Ontario, Canada) at a sampling rate of 1000 HZ. The authors used a custom-made calibration procedure with 13 points. Points were placed at the center of the nine cells and in the four possible locations of the fixation cross. The point located at the center of the matrix (corresponding to the central cell of the matrix) was repeated twice. After the calibration phase, a validation phase was performed to test the accuracy of the calibration. Calibration was repeated if necessary. During the experimental task, and before the beginning of each trial, a drift correction was performed in order to test whether participants looked at the current fixation location, by means of a target stimulus that was randomly presented in one of four possible locations. The matrix game

was presented to the participant after the target stimulus was fixated for 300 milliseconds. This was done to ensure that each participant, in every trial, was looking at one of four possible random points located outside the matrix before starting the game. To transform the data into scanpaths, the data was processed using the PyGaze library (Dalmaijer, Mathôt, and Stigchel, 2013), with colour maps from Matplotlib (Hunter, 2007).

7.0.3 Model Architectures & Training Procedure

We implemented a transfer learning and fine-tuning strategy using a VGG-19 model pre-trained on Imagenet as the baseline model. In both Classification Tasks, we freeze the convolutional base layers and replace the fully connected layers and final classification layer of the model with a custom head. We used Stochastic Gradient Descent (SDG) as the optimiser. Initially, we implemented Bayesian hyperparameter tuning via the Keras tuner (O'Malley et al., 2019) to select: (i) the number of fully connected layers, (ii) the number of nodes in each fully connected layer, and (iii) the dropout percentage between these layers. To improve classification accuracy, we fined-tuned the model by unfreezing some of the convolutional layers of the model and lowering the learning rate. To choose the optimal parameters during this stage, we used a random grid search which adjusts the number of frozen layers and the learning rate at each iteration.

As the target dataset differs greatly from the source dataset a number of regularization steps were also included in the model to help boost model performance and prevent overfitting. First, a Global Average Pooling (GAP) layer was inserted after the convolutional layers of the model to be used as a structure regularizer (Zhou et al., 2015). Additional measures include a light amount of L1/L2 regularization *(1e-5)* which is applied to the model alongside an early stopping callback. For both CTs, the model was able to train for a maximum of thirty epochs and up to ten further epochs during the fine-tuning stage.

In CT 2, to deal with the class imbalance problem, we implemented a focal loss function with the default values for the alpha and gamma

parameters. The focal loss was designed to deal with class imbalance by down-weighting well-classified examples and focusing on harder examples (Lin et al., 2018). All models were implemented using Tensorflow-Keras library (Chollet et al., 2015), which makes the Imagenet weights available for the VGG-19 model from its applications module, hence reducing the computational cost and time of running our experiments and making our experiments easier to reproduce.

For exploring the effects of scanpath feature engineering, To compare the effects of feature engineering in the scanpath images, we passed the datasets through a VGG-16 model pre-trained on the Imagenet dataset (Simonyan and Zisserman, 2015). We chose this model due to both its popularity and ease to implement, making it natural choice for many researchers, regardless of their level of deep learning experience. The VGG-16 model was created using the Pytorch library (Paszke et al., 2019), which makes the Imagenet weights available for VGG models as well as many other standard architectures. We choose Stochastic Gradient Descent (SDG) as the optimiser with a momentum of 0.9. As we are interested in comparing the impact of different representations of the input data - rather than optimising the model, we opted to train the model using a simple transfer learning strategy, freezing all of the layers of the network up to the classification layer, which we replaced to solve a binary classification problem. We choose this very vanilla training regime as it is easy to reproduce and one of the most commonly deployed training regimes for transfer learning and also produced good results in the previous eye-tracking studies (Castner, Kuebler, Scheiter, Richter, Eder, et al., 2020; Byrne, Reynolds, et al., 2023; Yin et al., 2021).

7.0.4 The Games

To implement our ML classification method we used data from Davide Marchiori, Sibilla Di Guida, and Luca Polonio (2021). "Plasticity of strategic sophistication in interactive decision-making". In: *Journal of Economic Theory* 196, p. 105291. The authors conducted a behavioural experiment recording eye movements of participants playing a set of games charac-

terized by a unique game theoretical optimal solution known as Nash Equilibrium (Fudenberg and Levine, 1993). In their study, the authors included three categories of games. In two of these categories, strategic sophistication is required in order to identify the Nash equilibrium. Strategic sophistication is the attempt to predict others' decisions by taking their incentives into account (Costa-Gomes, Crawford, and Broseta, 2001; Zonca, Vostroknutov, et al., 2021). In the third category, the equilibrium strategy can be identified without taking into account the other player incentives. In our experiment, we excluded games from this third category where the equilibrium strategy and more simple decision strategies overlap focusing on the two categories of games requiring strategic sophistication. Games in one selected category are called Dominant solvable Other (DO) games, and are characterized by the computer having a dominant strategy. This means that the computer has a strategy that provides strictly higher payoffs regardless of the strategy selected by the other player. Games in the second selected category are called Unique Equilibrium (UE) games, these games also include a unique Nash Equilibrium but do not contain a dominant strategy for either player and they are therefore not solvable by iterated dominance. Importantly, the Nash Equilibrium action and the actions corresponding with the Coordination and Naïve strategies were randomized across games to avoid regular patterns which could be recognised and exploited by a participant.

Previous eye-tracking research has shown that the analysis of the lookup patterns can be used to reconstruct, at the trial level, the decision process of different individuals and reveal the rationale behind their decisions (Polonio, Di Guida, and Coricelli, 2015). In games, possible deviations from equilibrium depend mainly on three components: (i) the social preferences of the players, (ii) their strategic thinking ability, and (iii) their beliefs about the expected behavior of their counterparts. In their study, Marchiori et al. (Marchiori, Di Guida, and Polonio, 2021) controlled for the first of these three components by providing information to the participants about the strategy of the computer. By telling the participants that the computer will play rationally with the aim of gaining as much as possible, they limited the participants' propensity to cooperate

by reducing their uncertainty towards the intentions and the strategic abilities of the counterpart. Possible deviations from equilibrium play are still possible and are mainly due to the inability of the player to properly represent the strategic environment, anticipate the other player's behavior, and best respond to it.

7.0.5 Analysis of Information Acquisition, Strategic Play

In our games, equilibrium play requires forming beliefs about the other player's actions. In terms of information acquisition, the belief formation process occurs through the use of saccades between the payoffs of the counterpart. Saccades express rapid eye movements connecting consecutive fixations and can be viewed as a valid and reliable measure of how individuals acquire and integrate different pieces of information (Devetag, Di Guida, and Polonio, 2016; Polonio and Coricelli, 2019). In the two categories of games selected for our experiment, looking at the other player's payoffs is not sufficient to identify the optimal solution of a game. Previous results show that equilibrium play, in games that require strategic sophistication, is associated with a large use of vertical and horizontal saccades between the other player's payoffs which are necessary to form accurate beliefs about the possible action of the counterpart, and vertical saccades between player's own payoffs which are necessary to identify the best response to it (Polonio, Di Guida, and Coricelli, 2015). Conversely, the implementation of a Naïve strategy is characterized by a high proportion of horizontal saccades between the payoffs of the player and more generally a high proportion of fixations on the player's own payoffs. Finally, actions consistent with a coordination strategy are usually associated with a higher proportion of diagonal saccades between the payoffs of the two players for each possible outcome of the game and in general balanced attention between own and other payoffs. Given their relevance for the identification of the decision strategy used by the individual, information about saccades are made salient via the use of colour when creating the scanpaths.

7.0.6 Comparison with Baseline Models

We compared our model to two common approaches used for classification problems, namely, a Support Vector Machine (SVM) image classifier and a logistic regression model. In the cases of the SVM, we implemented a Radial-Basis-Function (RBF) kernel SVM with the SVC estimator from the Python SkLearn library (Pedregosa et al., 2011). We used the same scanpaths as model input. We apply a 5-fold cross-validated, exhaustive parameter grid search to select the values of the hyperparameters. We set the kernel coefficient and regularization penalty to 0.0001 and 10, respectively, in the exact Machine Learning strategy task, the penalty parameter was computed with adjusted weights inversely proportional to class frequencies in the input data. To fit the logistic regression model to CT 1, we used a generalized linear model (GLM) from the `R` package `stats` (version 3.6.0). For CT 2, we applied a multinomial regression model within the `R` package `nnet` (version 7.3.16) (R Core Team, 2013). We fitted different models by regressing the binary outcome on proportions of *'Own'*, *'Other' and 'Intracell'* transitions recorded in intervals of the total duration of the game (at 15%, 30%, 50%, 80%, 100%) or in fixed time spans of 2, 5, 10 and 15 seconds.

7.0.7 Supplementary Figures from Scanpath Prediction Chapter

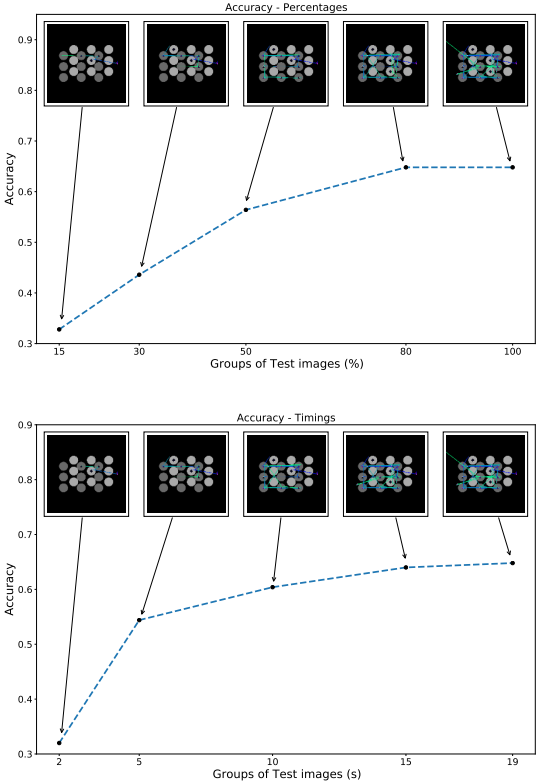


Figure 26: Accuracy of VGG-19 model in CT 2 using subsequences via percentages (a) and time points (b).

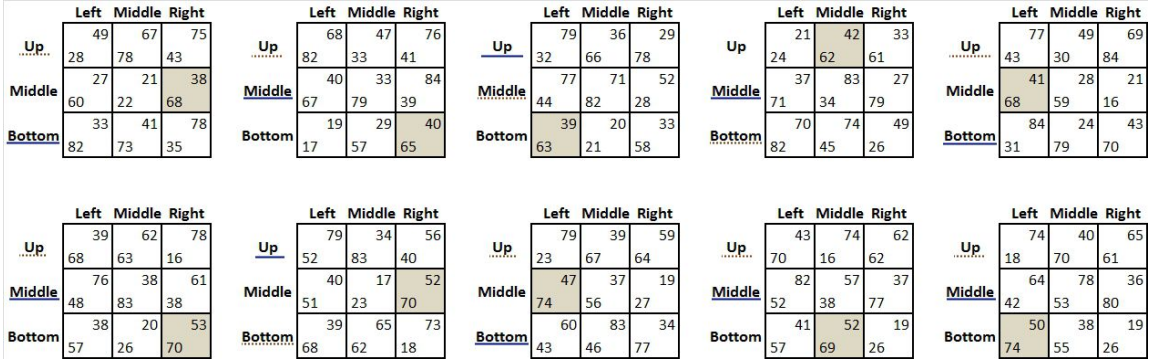
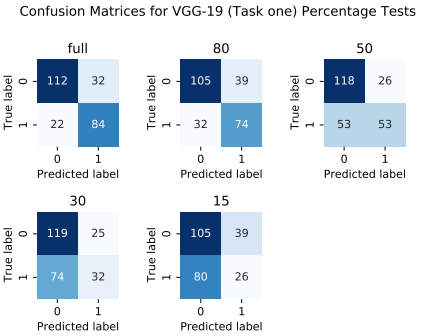
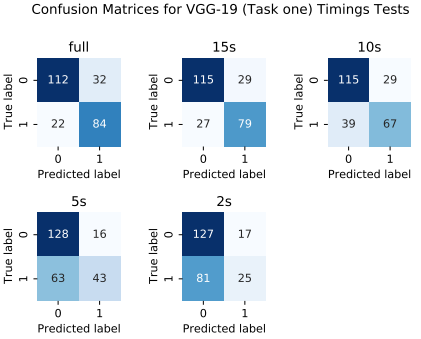


Figure 27: The games used in the experiment grouped by types. The payoffs of the Row/Column player are located in the Bottom-left/Upper-right part of the nine cells. The Nash equilibrium payoffs are indicated in grey. The Naive strategy is underlined with a solid line, and the Coordination strategy with a dashed line.

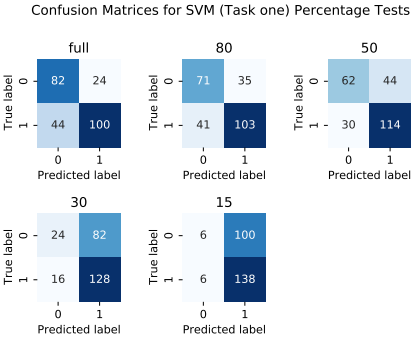


(a) Percentages

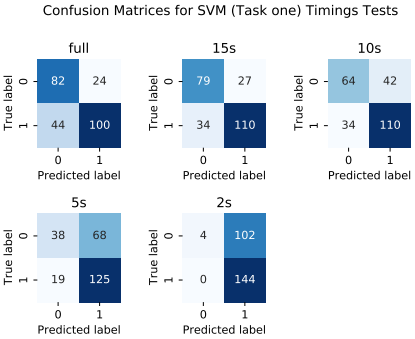


(b) Timings

Figure 28: Confusion Matrices of VGG19-model in CT 1 using subsequences via percentages (a) and time points (b).

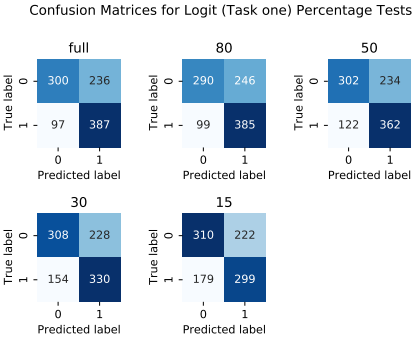


(a) Percentages

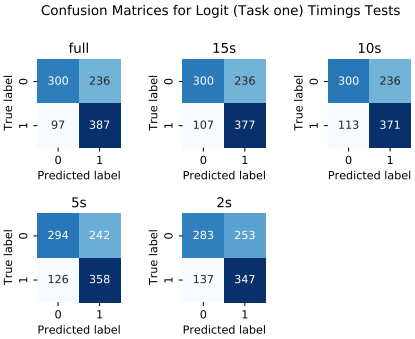


(b) Timings

Figure 29: Confusion Matrices of SVM-model in CT 1 using subsequences via percentages (a) and time points (b)

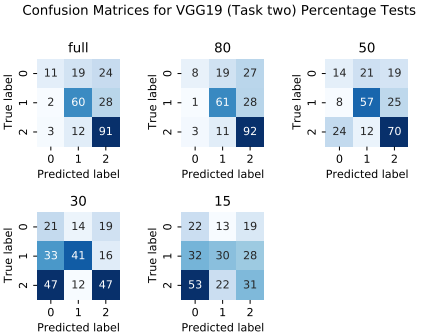


(a) Percentages

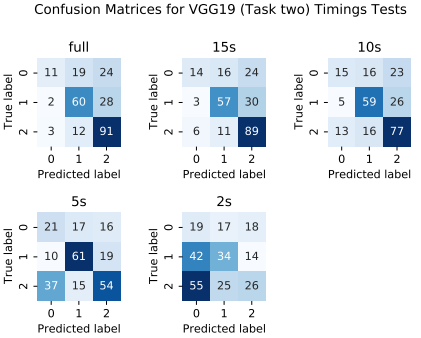


(b) Timings

Figure 30: Confusion Matrices of Logit regression model in CT 1 using subsequences via percentages (a) and time points (b)

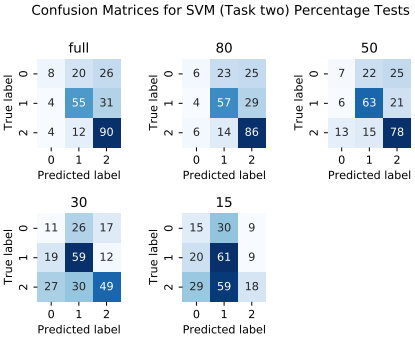


(a) Percentages

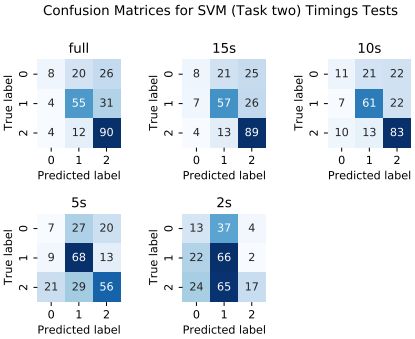


(b) Timings

Figure 31: Confusion Matrices of VGG19-model in CT 2 using subsequences via percentages (a) and time points (b)

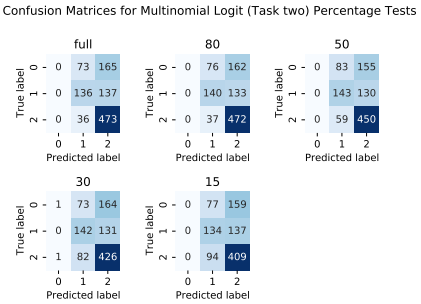


(a) Percentages

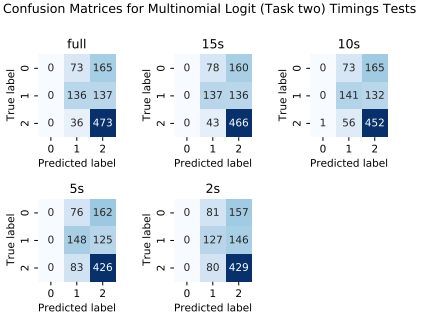


(b) Timings

Figure 32: Confusion Matrices of SVM-model in CT 2 using subsequences via percentages (a) and time points (b)



(a) Percentages



(b) Timings

Figure 33: Confusion Matrices of Multinomial Logit regression model in CT 2 using subsequences via percentages (a) and time points (b)

Percentages	Variable	Estimate	Std. Error	z value	Pr(>|z|)
Full	(Intercept)	0.6289	0.2918	2.16	0.0312
	Own	-4.2517	0.5469	-7.77	0.0000
	Other	2.8274	0.5590	5.06	0.0000
	Intracell	-3.0616	1.4590	-2.10	0.0359
80	(Intercept)	0.6058	0.2737	2.21	0.0269
	Own	-4.1175	0.5168	-7.97	0.0000
	Other	2.1976	0.5102	4.31	0.0000
	Intracell	-2.1822	1.3381	-1.63	0.1029
50	(Intercept)	0.5141	0.2426	2.12	0.0340
	Own	-3.4802	0.4586	-7.59	0.0000
	Other	1.4195	0.4386	3.24	0.0012
	Intracell	-0.9786	1.1078	-0.88	0.3770
30	(Intercept)	0.3457	0.1958	1.77	0.0774
	Own	-2.5292	0.3724	-6.79	0.0000
	Other	1.3784	0.3724	3.70	0.0002
	Intracell	-1.2306	0.7986	-1.54	0.1233
15	(Intercept)	0.1362	0.1475	0.92	0.3557
	Own	-1.7372	0.2930	-5.93	0.0000
	Other	1.4423	0.3140	4.59	0.0000
	Intracell	-0.5133	0.4307	-1.19	0.2334
Timings	Variable	Estimate	Std. Error	z value	Pr(>|z|)
Full	(Intercept)	0.6289	0.2918	2.16	0.0312
	Own	-4.2517	0.5469	-7.77	0.0000
	Other	2.8274	0.5590	5.06	0.0000
	Intracell	-3.0616	1.4590	-2.10	0.0359
15s	(Intercept)	0.5912	0.2785	2.12	0.0338
	Own	-3.9147	0.5151	-7.60	0.0000
	Other	2.3470	0.5233	4.49	0.0000
	Intracell	-2.3422	1.3380	-1.75	0.0800
10s	(Intercept)	0.6270	0.2614	2.40	0.0164
	Own	-3.8105	0.4868	-7.83	0.0000
	Other	1.8515	0.4853	3.81	0.0001
	Intracell	-1.7461	1.2094	-1.44	0.1488
5s	(Intercept)	0.6157	0.2363	2.61	0.0092
	Own	-3.3889	0.4371	-7.75	0.0000
	Other	1.1590	0.4334	2.67	0.0075
	Intracell	-0.7870	0.9360	-0.84	0.4005
2s	(Intercept)	0.4281	0.1781	2.40	0.0162
	Own	-2.5519	0.3409	-7.48	0.0000
	Other	0.6410	0.3451	1.86	0.0633
	Intracell	0.2406	0.5172	0.47	0.6418

Table 7: Logistic *(Logit)* Regression Estimates for Task 1.

Percentages	Variable	Estimate	Std. Error	Pr(>|z|)	z value
Full	Intracell (Naive)	-3.01	1.92	0.12	-1.57
	Other (Naive)	-4.40	0.86	0.00	-5.10
	Own (Naive)	1.28	0.66	0.05	1.94
	Intercept (Naive)	0.68	0.39	0.08	1.72
	Intracell (NE)	-3.07	1.65	0.06	-1.86
	Other (NE)	1.18	0.67	0.08	1.77
	Own (NE)	-2.97	0.64	0.00	-4.64
	Intercept (NE)	1.51	0.35	0.00	4.35
80	Intracell (Naive)	-1.39	1.79	0.44	-0.78
	Other (Naive)	-3.63	0.79	0.00	-4.59
	Own (Naive)	1.47	0.62	0.02	2.37
	Intercept (Naive)	0.45	0.37	0.22	1.22
	Intracell (NE)	-1.75	1.56	0.26	-1.12
	Other (NE)	0.95	0.61	0.12	1.56
	Own (NE)	-2.81	0.61	0.00	-4.60
	Intercept (NE)	1.40	0.33	0.00	4.26
50	Intracell (Naive)	-0.96	1.53	0.53	-0.63
	Other (Naive)	-3.01	0.69	0.00	-4.34
	Own (Naive)	1.29	0.56	0.02	2.29
	Intercept (Naive)	0.41	0.33	0.22	1.23
	Intracell (NE)	-0.96	1.35	0.48	-0.71
	Other (NE)	0.45	0.53	0.39	0.86
	Own (NE)	-2.43	0.55	0.00	-4.39
	Intercept (NE)	1.37	0.30	0.00	4.61
30	Intracell (Naive)	-1.60	1.08	0.14	-1.48
	Other (Naive)	-3.09	0.59	0.00	-5.21
	Own (Naive)	0.48	0.46	0.29	1.05
	Intercept (Naive)	0.73	0.27	0.01	2.70
	Intracell (NE)	-1.83	0.97	0.06	-1.89
	Other (NE)	0.09	0.45	0.85	0.19
	Own (NE)	-2.09	0.46	0.00	-4.55
	Intercept (NE)	1.48	0.25	0.00	5.96
15	Intracell (Naive)	-0.57	0.61	0.35	-0.94
	Other (Naive)	-2.67	0.52	0.00	-5.12
	Own (Naive)	0.42	0.37	0.27	1.11
	Intercept (Naive)	0.63	0.21	0.00	2.95
	Intracell (NE)	-0.56	0.54	0.30	-1.03
	Other (NE)	0.28	0.38	0.47	0.72
	Own (NE)	-1.47	0.37	0.00	-3.95
	Intercept (NE)	1.25	0.19	0.00	6.47

Table 8: Multinomial Regression Estimates for Task 2 (percentages).

Timings	Variable	Estimate	Std. Error	Pr(>|z|)	z value
Full	Intracell (Naive)	-3.01	1.92	0.12	-1.57
	Other (Naive)	-4.40	0.86	0.00	-5.10
	Own (Naive)	1.28	0.66	0.05	1.94
	Intercept (Naive)	0.68	0.39	0.08	1.72
	Intracell (NE)	-3.07	1.65	0.06	-1.86
	Other (NE)	1.18	0.67	0.08	1.77
	Own (NE)	-2.97	0.64	0.00	-4.64
	Intercept (NE)	1.51	0.35	0.00	4.35
15s	Intracell (Naive)	-1.88	1.80	0.30	-1.05
	Other (Naive)	-3.53	0.80	0.00	-4.39
	Own (Naive)	1.36	0.64	0.03	2.13
	Intercept (Naive)	0.47	0.38	0.21	1.24
	Intracell (NE)	-2.38	1.56	0.13	-1.53
	Other (NE)	1.00	0.63	0.11	1.58
	Own (NE)	-2.77	0.61	0.00	-4.52
	Intercept (NE)	1.47	0.34	0.00	4.34
10s	Intracell (Naive)	-1.55	1.68	0.35	-0.93
	Other (Naive)	-3.07	0.76	0.00	-4.06
	Own (Naive)	1.39	0.62	0.02	2.26
	Intercept (Naive)	0.38	0.37	0.30	1.04
	Intracell (NE)	-1.81	1.44	0.21	-1.25
	Other (NE)	0.67	0.59	0.25	1.14
	Own (NE)	-2.70	0.59	0.00	-4.61
	Intercept (NE)	1.49	0.32	0.00	4.64
5s	Intracell (Naive)	-1.79	1.36	0.19	-1.32
	Other (Naive)	-2.95	0.69	0.00	-4.29
	Own (Naive)	1.00	0.57	0.08	1.77
	Intercept (Naive)	0.50	0.34	0.14	1.48
	Intracell (NE)	-1.45	1.13	0.20	-1.29
	Other (NE)	-0.01	0.53	0.98	-0.02
	Own (NE)	-2.58	0.53	0.00	-4.83
	Intercept (NE)	1.60	0.29	0.00	5.44
2s	Intracell (Naive)	-1.73	0.80	0.03	-2.18
	Other (Naive)	-3.03	0.57	0.00	-5.36
	Own (Naive)	0.18	0.44	0.68	0.41
	Intercept (Naive)	0.85	0.26	0.00	3.25
	Intracell (NE)	-0.36	0.66	0.58	-0.55
	Other (NE)	-0.55	0.43	0.20	-1.29
	Own (NE)	-2.30	0.43	0.00	-5.35
	Intercept (NE)	1.60	0.23	0.00	6.86

Table 9: Multinomial Regression Estimates for Task 2 (timings).

Level of consistency	100%	90%	80%	70%	Total
Proportion of Subjects	14%	11%	13%	14%	51%
Number of Subjects	34	26	31	33	124

Table 10: Proportion and number of participants who use the same strategy 100%, 90%, 80% and 70% of the time over the course of the ten games.

Chapter 8

Conclusion

[1]

The dissertation presents methodological contributions for both gaze estimation and gaze analysis, with both contributions highlighting how deep learning methods can significantly improve task performance in comparison with traditional baselines. Starting with gaze estimation, LEyes has the potential to be a game-changer for the many companies and startups attempting to enter the VR and eye-tracking space. LEyes enables these companies to bring their devices to market without the necessity of collecting or purchasing potentially millions of eye images from a third party, alleviating both the costs and hurdles related to data acquisition. This opens up a streamlined path to market, making it an attractive option for emerging companies. In an academic setting, LEyes

[1]The conclusion drawn stem from the following co-authered work: 1.Sean Anthony Byrne, Marcus Nyström, et al. (Dec. 2023). "Precise Localization of Corneal Reflections in Eye Images Using Deep Learning Trained on Synthetic Data". In: *Behavior Research Methods*. ISSN: 1554-3528. DOI: 10.3758/s13428-023-02297-w. URL: https://doi.org/10.3758/s13428-023-02297-w, 2.Sean Anthony Byrne, Virmarie Maquiling, Marcus Nyström, et al. (2023). *LEyes: A Lightweight Framework for Deep Learning-Based Eye Tracking using Synthetic Eye Images*. arXiv: 2309.06129 [cs.CV], 3.Sean Anthony Byrne, Adam Peter Frederick Reynolds, et al. (2023). "Predicting choice behaviour in economic games using gaze data encoded as scanpath images". In: *Scientific Reports* 13.1, p. 4722, 4.Sean Anthony Byrne, Virmarie Maquiling, Adam Peter Frederick Reynolds, et al. (May 2023). "Exploring the Effects of Scanpath Feature Engineering for Supervised Image Classification Models". In: *Proc. ACM Hum.-Comput. Interact.* 7.ETRA. DOI: 10.1145/3591130. URL: https://doi.org/10.1145/3591130

significantly reduces the amount of data required to conduct an eye-tracking study that uses a deep learning model to analyze the data, by eliminating the need to sacrifice recorded data for model training and validation, resulting in both time and cost savings. For example, our model was able to run inference on the entirety of the Chugh et al. 2021 dataset, while the original paper used 88% of the data for both training and validation and were thus left with only 12% for evaluating there model (Chugh et al., 2021; Maquiling, Byrne, Nyström, et al., 2023). Furthermore, LEyes offers an alternative to the challenging task of creating photorealistic synthetic data. Many researchers may not possess the skills, time, or resources to access and use software platforms like Blender or Unity3D. Finally, when combined with the FLEX system which has a hardware cost of about $1000 USD, LEyes offers a low cost and open source alternative to the EyeLink 1000 Plus. Prior to LEyes, the development of gaze estimation algorithms using machine learning was confined to those who possessed the resources to amass large annotated datasets or the technical expertise and large computational resources to generate synthetic data. With LEyes, the training of deep learning models for gaze estimation has become easily accessible to everyone, democratizing the field and opening new avenues for exploration and application.

Moving to gaze estimation and scanpath analysis, our ventures underscores the potency of domain-specific feature engineering in creating superior scanpath images, surpassing baseline benchmarks in accuracy, F1-score, and AUC. Effective strategies like sequential coloring amplify this performance, whereas certain non-optimal tactics can hinder it. As the nexus between machine learning and eye-tracking research strengthens, it becomes imperative to recognize how domain knowledge can synergize with emerging models, rendering them more accessible and efficacious for the entire community. We focused our study on scanpath design showing its usefulness as model input by highlighting the results of both an out-of-the-box CNN and an SVM image classifier. Using scanpaths as model input both of our models outperform traditional methods to predict choice behavior from gaze data in games presented in

normal form. Much like in previous studies we aimed to classify gaze behaviour indicative of players who select the Nash-equilibrium and separately, the exact strategy used by players in our games. We were able to anticipate choices accurately with very little data, with both models performing well over chance with only 30% of the data available in the scanpath subsequences, and with CT 2 containing a moderate class imbalance. Our findings support the well-established hypothesis that it is possible to classify participants by the depth of their strategic abilities using the pattern of their eye-movements (Polonio, Di Guida, and Coricelli, 2015; Devetag, Di Guida, and Polonio, 2016; Li and Camerer, 2021; Li and Camerer, 2020). Not only does our approach deliver a more accurate classification of strategic types, but also contributes to the literature by providing a method that could be developed in future experiments to create an adaptive counterpart that modifies its playing style based on the eye movements of the participant, opening the door to a plethora of new experimental designs in this domain.

The integration of artificial intelligence with gaze data, while innovative, raises significant ethical concerns, especially in terms of user privacy (Steil et al., 2019). Key to navigating these concerns is ensuring that models are trained with an emphasis on equitable access and representation. This requires careful consideration of data sensitivity, securing user consent, and fostering inclusivity in the training process to prevent biases that could exclude or misrepresent diverse user groups. Indeed, the LEyes framework could be seen as a potential step forward in terms of ethical eye-tracking as the model is not trained using real eye images therefore eliminating the need to store large amounts of sensitive data for model training. However, the prediction of choice mentioned in Chapter 6 should serve as a warning as to how eye tracking could be used to create a net bad for the end users especially as these prediction methods are gaining traction in the literature with new models appearing such as RETINA which followed a similar design the study mentioned Chapter 6 but in the realm of consumer choice (Unger, Wedel, and Tuzhilin, 2023).

In the realm of research, the significance of eye-tracking data is on the rise, and scholars are increasingly focusing on refining its measure-

ment and analysis techniques. Much like various other disciplines, the field of eye-tracking has experienced substantial advancements through the incorporation of deep learning methodologies. Addressing the challenge of limited data availability, this dissertation has showcased ingenious strategies for training deep learning models. It is certain that forthcoming investigations will explore the potential of leveraging novel deep learning algorithms, including foundation models (Kirillov et al., 2023; Zhou et al., 2023), for enhancing eye-tracking studies. The impact of these models may be profound due to their multi-modal capabilities meaning eye tracking data may be integrated with other forms of data such as text or speech (Xu, Zhu, and Clifton, 2023). The extent of their influence on this domain remains an intriguing and unanswered question for the next generation of PhD Student entering field.

Bibliography

Ahmed, Zeyad AT and Mukti E Jadhav (2020). "Convolutional Neural Network for Prediction of Autism based on Eye-tracking Scanpaths". In: *International Journal of Psychosocial Rehabilitation* 24.05.

Akinyelu, Andronicus A. and Pieter Blignaut (2020). "Convolutional Neural Network-Based Methods for Eye Gaze Estimation: A Survey". In: *IEEE Access* 8, pp. 142581–142605. DOI: 10.1109/ACCESS.2020.3013540.

Amankwah-Amoah, Joseph et al. (2021). "COVID-19 and digitalization: The great acceleration". In: *Journal of Business Research* 136, pp. 602–611. ISSN: 0148-2963. DOI: https://doi.org/10.1016/j.jbusres.2021.08.011. URL: https://www.sciencedirect.com/science/article/pii/S0148296321005725.

Arulkumaran, Kai et al. (2017). "Deep reinforcement learning: A brief survey". In: *IEEE Signal Processing Magazine* 34.6, pp. 26–38.

Assens, Marc et al. (2018). "PathGAN: Visual scanpath prediction with generative adversarial networks". In: *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pp. 0–0.

Assens Reina, Marc et al. (2017). "Saltinet: Scan-path prediction on 360 degree images using saliency volumes". In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 2331–2338.

Atyabi, Adham et al. (2022). "Stratification of Children with Autism Spectrum Disorder through fusion of temporal information in eye-gaze scan-paths". In: *ACM Transactions on Knowledge Discovery from Data (TKDD)*.

Banerjee, Imon et al. (2019). "Comparative effectiveness of convolutional neural network (CNN) and recurrent neural network (RNN) architectures for radiology text report classification". In: *Artificial intelligence in medicine* 97, pp. 79–88.

Bansal, Ms Aayushi, Dr Rewa Sharma, and Dr Mamta Kathuria (2022). "A systematic review on data scarcity problem in deep learning: solution and applications". In: *ACM Computing Surveys (CSUR)* 54.10s, pp. 1–29.

Bao, Wentao and Zhenzhong Chen (2020). "Human scanpath prediction based on deep convolutional saccadic model". In: *Neurocomputing* 404, pp. 154–164. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2020.03.060. URL: https://www.sciencedirect.com/science/article/pii/S0925231220304331.

Barsingerhorn, A. D., F. N. Boonstra, and J. Goossens (2018). "Development and validation of a high-speed stereoscopic eyetracker". In: *Behavior Research Methods* 50.6, pp. 2480–2497. DOI: 10.3758/s13428-018-1026-7.

Barz, Michael and Daniel Sonntag (2016). "Gaze-guided object classification using deep neural networks for attention-based computing". In: *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, pp. 253–256.

Bhattacharya, Nilavra et al. (2020). "Relevance prediction from eye-movements using semi-interpretable convolutional neural networks". In: *Proceedings of the 2020 conference on human information interaction and retrieval*, pp. 223–233.

Blignaut, Pieter (2010). "Visual span and other parameters for the generation of heatmaps". In: *Proceedings of the 2010 symposium on eye-tracking research & applications*, pp. 125–128.

Borges, Aline FS et al. (2021). "The strategic use of artificial intelligence in the digital era: Systematic literature review and future research directions". In: *International Journal of Information Management* 57, p. 102225.

Braunagel, Christian, David Geisler, et al. (2017). "Online Recognition of Driver-Activity Based on Visual Scanpath Classification". In: *IEEE Intelligent Transportation Systems Magazine* 9.4, pp. 23–36. DOI: 10.1109/MITS.2017.2743171.

Braunagel, Christian, Wolfgang Rosenstiel, and Enkelejda Kasneci (2017). "Ready for take-over? A new driver assistance system for an automated classification of driver take-over readiness". In: *IEEE Intelligent Transportation Systems Magazine* 9.4, pp. 10–22.

Brunyé, Tad T et al. (2019). "A review of eye tracking for understanding and improving diagnostic interpretation". In: *Cognitive research: principles and implications* 4, pp. 1–16.

Buswell, Guy Thomas (1935). "How people look at pictures: a study of the psychology and perception in art." In.

Byrne, Sean Anthony, Nora Castner, Efe Bozkir, et al. (2024). "From Lenses to Living Rooms: A Policy Brief on Eye Tracking in XR Before the Impending Boom". In: *2024 IEEE International Conference on Artificial Intelligence and eXtended and Virtual Reality (AIxVR)*. IEEE, pp. 90–96.

Byrne, Sean Anthony, Nora Castner, Ard Kastrati, et al. (2023). "Leveraging Eye Tracking in Digital Classrooms: A Step Towards Multimodal Model for Learning Assistance". In: *Proceedings of the 2023 Symposium on Eye Tracking Research and Applications*. ETRA '23. Tubingen, Germany: Association for Computing Machinery. ISBN: 9798400701504. DOI: 10.1145/3588015.3589197. URL: https://doi.org/10.1145/3588015.3589197.

Byrne, Sean Anthony, Virmarie Maquiling, Marcus Nyström, et al. (2023). *LEyes: A Lightweight Framework for Deep Learning-Based Eye Tracking using Synthetic Eye Images*. arXiv: 2309.06129 [cs.CV].

Byrne, Sean Anthony, Virmarie Maquiling, Adam Peter Frederick Reynolds, et al. (May 2023). "Exploring the Effects of Scanpath Feature Engineering for Supervised Image Classification Models". In: *Proc. ACM Hum.-Comput. Interact.* 7.ETRA. DOI: 10.1145/3591130. URL: https://doi.org/10.1145/3591130.

Byrne, Sean Anthony, Marcus Nyström, et al. (Dec. 2023). "Precise Localization of Corneal Reflections in Eye Images Using Deep Learning Trained on Synthetic Data". In: *Behavior Research Methods*. ISSN: 1554-3528. DOI: 10.3758/s13428-023-02297-w. URL: https://doi.org/10.3758/s13428-023-02297-w.

Byrne, Sean Anthony, Adam Peter Frederick Reynolds, et al. (2023). "Predicting choice behaviour in economic games using gaze data encoded as scanpath images". In: *Scientific Reports* 13.1, p. 4722.

Camerer, Colin F and Teck-Hua Ho (2015). "Behavioral game theory experiments and modeling". In: *Handbook of game theory with economic applications* 4, pp. 517–573.

Castner, Nora, Enkelejda Kasneci, et al. (2018). "Scanpath comparison in medical image reading skills of dental students: distinguishing stages of expertise development". In: *Proceedings of the ACM Symposium on Eye Tracking Research & Applications*, pp. 1–9.

Castner, Nora, Thomas C Kuebler, Katharina Scheiter, Juliane Richter, Therese Eder, et al. (2020). "Deep Semantic Gaze Embedding and Scanpath Comparison for Expertise Classification during OPT Viewing". In: *ACM Symposium on Eye Tracking Research and Applications*. ETRA '20 Full Papers. Stuttgart, Germany: Association for Comput-

ing Machinery. ISBN: 9781450371339. DOI: 10.1145/3379155.3391320. URL: https://doi.org/10.1145/3379155.3391320.

Castner, Nora, Thomas C Kuebler, Katharina Scheiter, Juliane Richter, Thérése Eder, et al. (2020). "Deep semantic gaze embedding and scanpath comparison for expertise classification during OPT viewing". In: *ACM Symposium on Eye Tracking Research and Applications*, pp. 1–10.

Cerrolaza, Juan J et al. (2012). "Error characterization and compensation in eye tracking systems". In: *Proceedings of the symposium on eye tracking research and applications*, pp. 205–208.

Chaudhary, Aayush K et al. (2022). "Temporal RIT-Eyes: From real infrared eye-images to synthetic sequences of gaze behavior". In: *IEEE Transactions on Visualization and Computer Graphics* 28.11, pp. 3948–3958.

Chen, Shi and Qi Zhao (2019). "Attention-based autism spectrum disorder screening with privileged modality". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1181–1190.

Cheng, Yihua et al. (2021). "Appearance-based gaze estimation with deep learning: A review and benchmark". In: *arXiv preprint arXiv:2104.12668*.

Chollet, François et al. (2015). *Keras*. https://keras.io.

Chugh, Soumil et al. (2021). "Detection and Correspondence Matching of Corneal Reflections for Eye Tracking Using Deep Learning". In: *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, pp. 2210–2217.

Clay, Viviane, Peter König, and Sabine Koenig (2019). "Eye tracking in virtual reality". In: *Journal of eye movement research* 12.1.

Coricelli, Giorgio, Luca Polonio, and Alexander Vostroknutov (2020). "The process of choice in games". In: *Handbook of experimental game theory*. Edward Elgar Publishing.

Costa, Yandre MG, Luiz S Oliveira, and Carlos N Silla Jr (2017). "An evaluation of convolutional neural networks for music classification using spectrograms". In: *Applied soft computing* 52, pp. 28–38.

Costa-Gomes, Miguel, Vincent P. Crawford, and Bruno Broseta (2001). "Cognition and Behavior in Normal-Form Games: An Experimental Study". In: *Econometrica* 69.5, pp. 1193–1235. DOI: https://doi.org/10.1111/1468-0262.00239. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/1468-0262.00239. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/1468-0262.00239.

Costa-Gomes, Miguel A and Georg Weizsäcker (2008). "Stated beliefs and play in normal-form games". In: *The Review of Economic Studies* 75.3, pp. 729–762.

Coutrot, Antoine, Janet H Hsiao, and Antoni B Chan (2018). "Scanpath modeling and classification with hidden Markov models". In: *Behavior Research Methods* 50.1, pp. 362–379.

Crabb, David P, Nicholas D Smith, and Haogang Zhu (2014). "What's on TV? Detecting age-related neurodegenerative eye disease using eye movement scanpaths". In: *Frontiers in Aging Neuroscience* 6, p. 312.

Crawford, Vincent (1997). "Theory and experiment in the analysis of strategic interaction". In: *(1997*.

Dalmaijer, Edwin, Sebastiaan Mathôt, and Stefan Stigchel (Nov. 2013). "PyGaze: An open-source, cross-platform toolbox for minimal-effort programming of eyetracking experiments". In: *Behavior Research Methods* 46. DOI: 10.3758/s13428-013-0422-2.

De Kloe, Yentl JR et al. (2022). "Replacing eye trackers in ongoing studies: A comparison of eye-tracking data quality between the Tobii Pro TX300 and the Tobii Pro Spectrum". In: *Infancy* 27.1, pp. 25–45.

Deng, Jia et al. (2009). "ImageNet: A large-scale hierarchical image database". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

Devetag, Giovanna, Sibilla Di Guida, and Luca Polonio (2016). "An eye-tracking study of feature-based choice in one-shot games". In: *Experimental Economics* 19, pp. 177–201.

Devlin, Jacob et al. (2018). "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *arXiv preprint arXiv:1810.04805*.

Dewhurst, Richard et al. (2018). "How task demands influence scanpath similarity in a sequential number-search task". In: *Vision Research* 149, pp. 9–23.

Dosovitskiy, Alexey et al. (2020). *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. DOI: 10.48550/ARXIV.2010.11929. URL: https://arxiv.org/abs/2010.11929.

Dung, Cao Vu et al. (2019). "A vision-based method for crack detection in gusset plate welded joints of steel bridges using deep convolutional neural networks". In: *Automation in Construction* 102, pp. 217–229.

Einhäuser, Wolfgang, Ueli Rutishauser, Christof Koch, et al. (2008). "Task-demands can immediately reverse the effects of sensory-driven saliency in complex visual stimuli". In: *Journal of Vision* 8.2, pp. 2–2.

Elbattah, Mahmoud et al. (2019). "Learning clusters in autism spectrum disorder: Image-based clustering of eye-tracking scanpaths with deep

autoencoder". In: *2019 41st Annual international conference of the IEEE engineering in medicine and biology society (EMBC)*. IEEE, pp. 1417–1420.

Fudenberg, Drew and David Levine (Feb. 1993). "Steady State Learning and Nash Equilibrium". In: *Econometrica* 61, pp. 547–73. DOI: 10.2307/2951717.

Fudenberg, Drew and David K Levine (2016). "Whither game theory? Towards a theory of learning in games". In: *Journal of Economic Perspectives* 30.4, pp. 151–70.

Fuhl, Wolfgang, Efe Bozkir, Benedikt Hosp, Nora Castner, David Geisler, Thiago C Santini, et al. (2019a). "Encodji: encoding gaze data into emoji space for an amusing scanpath classification approach". In: *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications*, pp. 1–4.

— (2019b). "Encodji: Encoding Gaze Data into Emoji Space for an Amusing Scanpath Classification Approach ;)" in: *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications*. ETRA '19. Denver, Colorado: Association for Computing Machinery. ISBN: 9781450367097. DOI: 10.1145/3314111.3323074. URL: https://doi.org/10.1145/3314111.3323074.

Fuhl, Wolfgang, Thomas Kuebler, et al. (2018). "Automatic generation of saliency-based areas of interest for the visualization and analysis of eye-tracking data". In: *Proceedings of the Conference on Vision, Modeling, and Visualization*, pp. 47–54.

Fuhl, Wolfgang, Thiago Santini, Gjergji Kasneci, and Enkelejda Kasneci (2016). "Pupilnet: Convolutional neural networks for robust pupil detection". In: *arXiv preprint arXiv:1601.04902*.

Fuhl, Wolfgang, Thiago Santini, Gjergji Kasneci, Wolfgang Rosenstiel, et al. (2017). *PupilNet v2.0: Convolutional Neural Networks for CPU based real time Robust Pupil Detection*. arXiv: 1711.00112 [cs.CV].

Fuhl, Wolfgang, Daniel Weber, and Shahram Eivazi (2023). *Pistol: Pupil Invisible Supportive Tool to extract Pupil, Iris, Eye Opening, Eye Movements, Pupil and Iris Gaze Vector, and 2D as well as 3D Gaze*. arXiv: 2201.06799 [cs.CV].

Gao, Cong et al. (2023). "Synthetic data accelerates the development of generalizable learning-based algorithms for X-ray image analysis". In: *Nature Machine Intelligence* 5.3, pp. 294–308.

Garbin, Stephan Joachim et al. (2020). "Dataset for eye tracking on a virtual reality platform". In: *ACM Symposium on Eye Tracking Research and Applications*, pp. 1–10.

Geisler, David et al. (2020). "Exploiting the GBVS for Saliency Aware Gaze Heatmaps". In: *ACM Symposium on Eye Tracking Research and Applications*. ETRA '20 Short Papers. Stuttgart, Germany: Association for Computing Machinery. ISBN: 9781450371346. DOI: 10.1145/3379156.3391367. URL: https://doi.org/10.1145/3379156.3391367.

Goldberg, Joseph H and Jonathan I Helfman (2010). "Visual scanpath representation". In: *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, pp. 203–210.

Hargreaves Heap, Shaun, David Rojo Arjona, and Robert Sugden (2014). "How portable is level-0 behavior? A test of level-k theory in games with non-neutral frames". In: *Econometrica* 82.3, pp. 1133–1151.

He, Kaiming et al. (2015). *Deep Residual Learning for Image Recognition*. DOI: 10.48550/ARXIV.1512.03385. URL: https://arxiv.org/abs/1512.03385.

Helgadottir, Saga, Aykut Argun, and Giovanni Volpe (2019). "Digital video microscopy enhanced by deep learning". In: *Optica* 6.4, pp. 506–513.

Hendrycks, Dan and Kevin Gimpel (2016). "A baseline for detecting misclassified and out-of-distribution examples in neural networks". In: *arXiv preprint arXiv:1610.02136*.

Hessels, Roy S et al. (2018). "Is the eye-movement field confused about fixations and saccades? A survey among 124 researchers". In: *Royal Society open science* 5.8, p. 180502.

Holmqvist, K. et al. (2011). *Eye tracking: a comprehensive guide to methods and measures*. Oxford University Press.

Holmqvist, Kenneth and Pieter Blignaut (2020). "Small eye movements cannot be reliably measured by video-based P-CR eye-trackers". In: *Behavior research methods* 52.5, pp. 2098–2121.

Holmqvist, Kenneth, Marcus Nyström, Richard Andersson, et al. (2011). *Eye tracking: A comprehensive guide to methods and measures*. OUP Oxford.

Holmqvist, Kenneth, Marcus Nyström, and Fiona Mulvey (2012). "Eye tracker data quality: What it is and how to measure it". In: *Proceedings of the Symposium on Eye Tracking Research and Applications*. ACM, pp. 45–52.

Hooge, Ignace, Kenneth Holmqvist, and Marcus Nyström (2016). "The pupil is faster than the corneal reflection (CR): Are video based pupil-CR eye trackers suitable for studying detailed dynamics of eye movements?" In: *Vision research* 128, pp. 6–18.

Hooge, Ignace T C, Diederick C Niehorster, Roy S Hessels, Jeroen S Benjamins, et al. (2022). "How robust are wearable eye trackers to slow

and fast head and body movements?" In: *Behavior Research Methods*, pp. 1–15.

Hooge, Ignace T C, Diederick C Niehorster, Roy S Hessels, Dixon Cleveland, et al. (2021). "The pupil-size artefact (PSA) across time, viewing direction, and different eye trackers". In: *Behavior Research Methods* 53.5, pp. 1986–2006.

Hooge, Ignace T C, Diederick C Niehorster, Marcus Nyström, et al. (2018). "Is human classification by experienced untrained observers a gold standard in fixation detection?" In: *Behavior Research Methods* 50.5, pp. 1864–1881.

Hosp, Benedikt, Shahram Eivazi, et al. (2020). "RemoteEye: An open-source high-speed remote eye tracker". In: *Behavior Research Methods* 52.3, pp. 1387–1401. DOI: 10.3758/s13428-019-01305-2.

Hosp, Benedikt, Florian Schultz, et al. (2021). "Expertise classification of soccer goalkeepers in highly dynamic decision tasks: a deep learning approach for temporal and spatial feature recognition of fixation image patch sequences". In: *Frontiers in Sports and Active Living*, p. 183.

Hosp, Benedikt, Myat Su Yin, et al. (2021). "Differentiating Surgeons' Expertise solely by Eye Movement Features". In: *Companion Publication of the 2021 International Conference on Multimodal Interaction*, pp. 371–375.

Hunter, J. D. (2007). "Matplotlib: A 2D graphics environment". In: *Computing in Science & Engineering* 9.3, pp. 90–95. DOI: 10.1109/MCSE.2007.55.

Iakubovskii, Pavel (2019). *Segmentation Models Pytorch*. https://github.com/qubvel/segmentation_models.pytorch.

Isokoski, Poika, Jari Kangas, and Päivi Majaranta (2018). "Useful approaches to exploratory analysis of gaze data: enhanced heatmaps, cluster maps, and transition maps". In: *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*, pp. 1–9.

Ivanchenko, Daria et al. (May 2021). "A low-cost, high-performance video-based binocular eye tracker for psychophysical research". In: *Journal of Eye Movement Research* 14.3. DOI: 10.16910/jemr.14.3.3.

Jarodzka, Halszka, Kenneth Holmqvist, and Marcus Nyström (2010). "A vector-based, multidimensional scanpath similarity measure". In: *Proceedings of the 2010 symposium on eye-tracking research & applications*, pp. 211–218.

Jha, Dipendra et al. (2018). "Elemnet: Deep learning the chemistry of materials from only elemental composition". In: *Scientific reports* 8.1, pp. 1–13.

Jiang, Ming et al. (June 2015). "SALICON: Saliency in Context". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Johnson, Justin and Taghi Khoshgoftaar (Mar. 2019). "Survey on deep learning with class imbalance". In: *Journal of Big Data* 6, p. 27. DOI: 10.1186/s40537-019-0192-5.

Johnson, Justin M and Taghi M Khoshgoftaar (2019). "Survey on deep learning with class imbalance". In: *Journal of Big Data* 6.1, pp. 1–54.

Jun, Tae Joon et al. (2018). *ECG arrhythmia classification using a 2-D convolutional neural network*. DOI: 10.48550/ARXIV.1804.06812. URL: https://arxiv.org/abs/1804.06812.

Kacur, Juraj et al. (2020). "An analysis of eye-tracking features and modelling methods for free-viewed standard stimulus: application for schizophrenia detection". In: *IEEE Journal of Biomedical and Health Informatics* 24.11, pp. 3055–3065.

Kim, Joohwan et al. (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*, pp. 1–12.

Kim, Soo-Hyung, Guee-Sang Lee, Hyung-Jeong Yang, et al. (2019). "Eye semantic segmentation with a lightweight model". In: *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. IEEE, pp. 3694–3697.

Kingma, Diederik P and Jimmy Ba (2014). "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980*.

— (2017). *Adam: A Method for Stochastic Optimization*. arXiv: 1412.6980 [cs.LG].

Kirillov, Alexander et al. (2023). "Segment anything". In: *arXiv preprint arXiv:2304.02643*.

Knoepfle, Daniel T., Colin F. Camerer, and Joseph Tao-yi Wang (2009). "Studying Learning in Games Using Eye-Tracking". In: *Journal of the European Economic Association* 7.2/3, pp. 388–398. ISSN: 15424766, 15424774. URL: http://www.jstor.org/stable/40282757.

Koonce, Brett and Brett Koonce (2021). "ResNet 34". In: *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*, pp. 51–61.

Kothari, Rakshit et al. (2020). "Gaze-in-wild: A dataset for studying eye and head coordination in everyday activities". In: *Scientific reports* 10.1, p. 2539.

Kothari, Rakshit S et al. (2022a). "EllSeg-Gen, towards Domain Generalization for head-mounted eyetracking". In: *Proceedings of the ACM on Human-Computer Interaction* 6.ETRA, pp. 1–17.

— (May 2022b). "EllSeg-Gen, towards Domain Generalization for Head-Mounted Eyetracking". In: *Proc. ACM Hum.-Comput. Interact.* 6.ETRA. DOI: 10.1145/3530880. URL: https://doi.org/10.1145/3530880.

Krafka, Kyle et al. (2016). "Eye tracking for everyone". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2176–2184.

Krol, Michal and Magdalena Krol (2017). "A novel approach to studying strategic decisions with eye-tracking and machine learning". In: *Judgment and Decision Making* 12.6, p. 596.

Kübler, Thomas C et al. (2017). "SubsMatch 2.0: Scanpath comparison and classification based on subsequence frequencies". In: *Behavior research methods* 49.3, pp. 1048–1064.

Kübler, Thomas C., Enkelejda Kasneci, and Wolfgang Rosenstiel (2014). "SubsMatch: Scanpath Similarity in Dynamic Scenes Based on Subsequence Frequencies". In: *Proceedings of the Symposium on Eye Tracking Research and Applications*. ETRA '14. Safety Harbor, Florida: Association for Computing Machinery, pp. 319–322. ISBN: 9781450327510. DOI: 10.1145/2578153.2578206. URL: https://doi.org/10.1145/2578153.2578206.

Kumar, Ayush et al. (2020). "Challenges in interpretability of neural networks for eye movement data". In: *ACM Symposium on Eye Tracking Research and Applications*, pp. 1–5.

Kümmerer, Matthias and Matthias Bethge (2021). "State-of-the-Art in Human Scanpath Prediction". In: *CoRR* abs/2102.12239. arXiv: 2102.12239. URL: https://arxiv.org/abs/2102.12239.

Kümmerer, Matthias, Lucas Theis, and Matthias Bethge (2015). *Deep Gaze I: Boosting Saliency Prediction with Feature Maps Trained on ImageNet*. arXiv: 1411.1045 [cs.CV].

Lee, JooWon and Jae-Hyeon Ahn (2012). "Attention to banner ads and their effectiveness: An eye-tracking approach". In: *International Journal of Electronic Commerce* 17.1, pp. 119–137.

Li, Dongheng, D. Winfield, and D.J. Parkhurst (2005). "Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches". In: *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, pp. 79–79. DOI: 10.1109/CVPR.2005.531.

Li, Xiaomin and Colin Camerer (2020). "Predictable Effects of Bottom-up Visual Salience in Experimental Decisions and Games". In: *Available at SSRN 3308886*.

Li, Xiaomin and Colin Camerer (2021). "Hidden Markov Modeling of the Cognitive Process in Strategic Thinking". In: *Available at SSRN 3838911*.

Li, Xiaoya et al. (2019). "Dice loss for data-imbalanced NLP tasks". In: *arXiv preprint arXiv:1911.02855*.

Lin, Tsung-Yi et al. (2017). "Focal loss for dense object detection". In: *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988.

— (2018). *Focal Loss for Dense Object Detection*. arXiv: 1708.02002 [cs.CV].

Liszio, Stefan and Maic Masuch (2016). "Designing shared virtual reality gaming experiences in local multi-platform games". In: *Entertainment Computing-ICEC 2016: 15th IFIP TC 14 International Conference, Vienna, Austria, September 28-30, 2016, Proceedings 15*. Springer, pp. 235–240.

Loshchilov, Ilya and Frank Hutter (2019). *Decoupled Weight Decay Regularization*. arXiv: 1711.05101 [cs.LG].

Lotz, Alexander and Sarah Weissenberger (2018). "Predicting take-over times of truck drivers in conditional autonomous driving". In: *International Conference on Applied Human Factors and Ergonomics*. Springer, pp. 329–338.

Maquiling, Virmarie, Sean Anthony Byrne, Diederick C. Niehorster, et al. (2023). *Zero-Shot Segmentation of Eye Features Using the Segment Anything Model (SAM)*. arXiv: 2311.08077 [cs.CV].

Maquiling, Virmarie, Sean Anthony Byrne, Marcus Nyström, et al. (Sept. 2023). "V-ir-Net: A Novel Neural Network for Pupil and Corneal Reflection Detection trained on Simulated Light Distributions". In: *25th International Conference on Mobile Human-Computer Interaction (MobileHCI '23 Companion)*. Athens, Greece: ACM. ISBN: 978-1-4503-9924-1/23/09. DOI: 10.1145/3565066.3608690.

March, Christoph (2021). "Strategic interactions between humans and artificial intelligence: Lessons from experiments with computer players". In: *Journal of Economic Psychology* 87, p. 102426. ISSN: 0167-4870. DOI: https://doi.org/10.1016/j.joep.2021.102426. URL: https://www.sciencedirect.com/science/article/pii/S0167487021000593.

Marchiori, Davide, Sibilla Di Guida, and Luca Polonio (2021). "Plasticity of strategic sophistication in interactive decision-making". In: *Journal of Economic Theory* 196, p. 105291.

Mathew, Amitha, P Amudha, and S Sivakumari (2020). "Deep learning techniques: an overview". In: *International conference on advanced machine learning technologies and applications*. Springer, pp. 599–608.

Merchant, John, Richard Morrissette, and James L Porterfield (1974). "Remote measurement of eye direction allowing subject motion over one cubic foot of space". In: *IEEE transactions on biomedical engineering* 4, pp. 309–317.

Midtvedt, Benjamin, Saga Helgadottir, et al. (2021). "Quantitative digital microscopy with deep learning". In: *Applied Physics Reviews* 8.1, p. 011310.

Midtvedt, Benjamin, Jesús Pineda, et al. (2022). "Single-shot self-supervised object detection in microscopy". In: *Nature Communications* 13.1, pp. 1–13.

Mishra, Abhijit and Pushpak Bhattacharyya (Aug. 2018). "Automatic Extraction of Cognitive Features from Gaze Data: An Investigation Based on Eye-tracking". In: Springer, pp. 153–169. ISBN: 978-981-13-1515-2. DOI: 10.1007/978-981-13-1516-9_7.

Moolayil, Jojo, Jojo Moolayil, and Suresh John (2019). *Learn Keras for deep neural networks*. Springer.

Mulligan, J. B. (1997). "Image processing for improved eye-tracking accuracy". In: *Behavior Research Methods, Instruments, & Computers* 29, pp. 54–65. DOI: 10.3758/BF03200567.

Nair, Nitinraj et al. (2020). "RIT-Eyes: Rendering of near-eye images for eye-tracking applications". In: *ACM Symposium on Applied Perception 2020*, pp. 1–9.

Nash Jr, John F (1950). "Equilibrium points in n-person games". In: *Proceedings of the national academy of sciences* 36.1, pp. 48–49.

Naspetti, Simona et al. (2016). "Automatic analysis of eye-tracking data for augmented reality applications: A prospective outlook". In: *Augmented Reality, Virtual Reality, and Computer Graphics: Third International Conference, AVR 2016, Lecce, Italy, June 15-18, 2016. Proceedings, Part II 3*. Springer, pp. 217–230.

Niehorster, Diederick C, Roy S Hessels, and Jeroen S Benjamins (2020). "GlassesViewer: Open-source software for viewing and analyzing data from the Tobii Pro Glasses 2 eye tracker". In: *Behavior Research Methods* 52.3, pp. 1244–1253.

Niehorster, Diederick C and Marcus Nyström (2018). "Microsaccade Detection Using Pupil and Corneal Reflection Signals". In: *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*. ETRA '18. Warsaw, Poland: Association for Computing Machinery. ISBN: 9781450357067. DOI: 10.1145/3204493.3204573.

Niehorster, Diederick C, Thiago Santini, et al. (2020). "The impact of slippage on the data quality of head-worn eye trackers". In: *Behavior Re-*

search Methods 52.3, pp. 1140–1160. DOI: 10.3758/s13428-019-01307-0.

Niehorster, Diederick C, Raimondas Zemblys, Tanya Beelders, et al. (2020). "Characterizing gaze position signals and synthesizing noise during fixations in eye-tracking data". In: *Behavior Research Methods* 52.6, pp. 2515–2534. DOI: 10.3758/s13428-020-01400-9.

Niehorster, Diederick C, Raimondas Zemblys, and Kenneth Holmqvist (2021a). "Is apparent fixational drift in eye-tracking data due to filters or eyeball rotation?" In: *Behavior Research Methods* 53.1, pp. 311–324. DOI: 10.3758/s13428-020-01414-3.

— (2021b). "Is apparent fixational drift in eye-tracking data due to filters or eyeball rotation?" In: *Behavior Research Methods* 53, pp. 311–324.

Niehorster, Diederick C. et al. (in press). "GlassesValidator: A data quality tool for eye tracking glasses". In: *Behavior research methods*.

Nielsen, Jakob and Kara Pernice (2009). *Eyetracking Web Usability*. 1st. USA: New Riders Publishing. ISBN: 0321498364.

Niu, Lijinliang et al. (2021). "Real-Time Localization and Matching of Corneal Reflections for Eye Gaze Estimation via a Lightweight Network". In: *The Ninth International Symposium of Chinese CHI*, pp. 33–40.

Nyström, Marcus, Diederick C Niehorster, Richard Andersson, Roy S Hessels, and Ignace T C Hooge (2022). "The amplitude of small eye movements can be accurately estimated with video-based eye trackers". In: *Behavior Research Methods*, pp. 1–13.

— (2023). "The amplitude of small eye movements can be accurately estimated with video-based eye trackers". In: *Behavior Research Methods* 55.2, pp. 657–669.

O'Malley, Tom et al. (2019). *KerasTuner*. https://github.com/keras-team/keras-tuner.

Osiński, Błażej et al. (2020). "Simulation-based reinforcement learning for real-world autonomous driving". In: *2020 IEEE international conference on robotics and automation (ICRA)*. IEEE, pp. 6411–6418.

Palmero, Cristina et al. (2021). "Openeds2020 challenge on gaze tracking for vr: Dataset and results". In: *Sensors* 21.14, p. 4769.

Papoutsaki, Alexandra, James Laskey, and Jeff Huang (2017). "SearchGazer: Webcam Eye Tracking for Remote Studies of Web Search". In: *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval*. CHIIR '17. Oslo, Norway: Association for Computing Machinery, pp. 17–26. ISBN: 9781450346771. DOI: 10.1145/

3020165.3020170. URL: https://doi.org/10.1145/3020165.3020170.

Park, Seonwook et al. (2018). "Learning to find eye region landmarks for remote gaze estimation in unconstrained settings". In: *Proceedings of the 2018 ACM symposium on eye tracking research & applications*, pp. 1–10.

Parthasarathy, Raghuveer (2012). "Rapid, accurate particle tracking by calculation of radial symmetry centers". In: *Nature methods* 9.7, pp. 724–726.

Paszke, Adam et al. (2019). *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. DOI: 10.48550/ARXIV.1912.01703. URL: https://arxiv.org/abs/1912.01703.

Pedregosa, F. et al. (2011). "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12, pp. 2825–2830.

Peréz, Antonio et al. (2003). "A precise eye-gaze detection and tracking system". In: *Proceedings of The 11th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*. WSCG'2003. Plzen, Czech Republic, pp. 1–4. URL: http://wscg.zcu.cz/wscg2003/Papers_2003/A83.pdf.

Pfeiffer, Thies and Cem Memili (2016). "Model-based real-time visualization of realistic three-dimensional heat maps for mobile eye tracking and eye tracking in virtual reality". In: *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*, pp. 95–102.

Pierce, Karen et al. (2016). "Eye tracking reveals abnormal visual preference for geometric images as an early biomarker of an autism spectrum disorder subtype associated with increased symptom severity". In: *Biological psychiatry* 79.8, pp. 657–666.

Pineda, Jesús et al. (2022). *Geometric deep learning reveals the spatiotemporal fingerprint of microscopic motion*. DOI: 10.48550/ARXIV.2202.06355. URL: https://arxiv.org/abs/2202.06355.

Polonio, Luca and Giorgio Coricelli (2019). "Testing the level of consistency between choices and beliefs in games using eye-tracking". In: *Games and Economic Behavior* 113, pp. 566–586.

Polonio, Luca, Sibilla Di Guida, and Giorgio Coricelli (2015). "Strategic sophistication and attention in games: An eye-tracking study". In: *Games and Economic Behavior* 94, pp. 80–96.

Popelka, Stanislav and Marketa Beitlova (2022). "Scanpath Comparison using ScanGraph for Education and Learning Purposes: Summary of previous educational studies performed with the use of ScanGraph". In: *2022 Symposium on Eye Tracking Research and Applications*, pp. 1–6.

Praveena, KN and R Mahalakshmi (2022). "Classification of Autism Spectrum Disorder and Typically Developed Children for Eye Gaze Image Dataset using Convolutional Neural Network". In: *International Journal of Advanced Computer Science and Applications* 13.3.

Privitera, Claudio M. and Lawrence W. Stark (2000). "Algorithms for defining visual regions-of-interest: Comparison with eye fixations". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.9, pp. 970–982.

R Core Team (2013). *R: A Language and Environment for Statistical Computing*. ISBN 3-900051-07-0. R Foundation for Statistical Computing. Vienna, Austria. URL: http://www.R-project.org/.

Raghu, Maithra et al. (2019). "Transfusion: Understanding transfer learning for medical imaging". In: *Advances in neural information processing systems* 32.

Rai, Yashas, Patrick Le Callet, and Philippe Guillotel (2017). "Which saliency weighting for omni directional image quality assessment?" In: *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, pp. 1–6.

Rajashekar, Umesh et al. (2008). "GAFFE: A gaze-attentive fixation finding engine". In: *IEEE Transactions on Image Processing* 17.4, pp. 564–573.

Rawat, Tara and Vineeta Khemchandani (2017). "Feature engineering (FE) tools and techniques for better classification performance". In: *International Journal of Innovations in Engineering and Technology* 8.2, pp. 169–179.

Rayner, Keith (1998). "Eye movements in reading and information processing: 20 years of research." In: *Psychological bulletin* 124.3, p. 372.

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (2015). *U-Net: Convolutional Networks for Biomedical Image Segmentation*. arXiv: 1505.04597 [cs.CV].

Roopak, Monika, Gui Yun Tian, and Jonathon Chambers (2019). "Deep learning models for cyber security in IoT networks". In: *2019 IEEE 9th annual computing and communication workshop and conference (CCWC)*. IEEE, pp. 0452–0457.

Ruby, Usha and Vamsidhar Yendapalli (2020). "Binary cross entropy with deep learning technique for image classification". In: *Int. J. Adv. Trends Comput. Sci. Eng* 9.10.

San Agustin, Javier et al. (2010). "Evaluation of a Low-Cost Open-Source Gaze Tracker". In: *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*. ETRA '10. Austin, Texas: Association for Com-

puting Machinery, pp. 77–80. ISBN: 9781605589947. DOI: 10.1145/1743666.1743685.

Santini, Thiago, Wolfgang Fuhl, and Enkelejda Kasneci (2018). "PuRe: Robust pupil detection for real-time pervasive eye tracking". In: *Computer Vision and Image Understanding* 170, pp. 40–50.

Shah, Devarshi, Jin Wang, and Q Peter He (2020). "Feature engineering in big data analytics for IoT-enabled smart manufacturing–Comparison between deep learning and statistical learning". In: *Computers & Chemical Engineering* 141, p. 106970.

Shin, Hoo-Chang et al. (2016). "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning". In: *IEEE transactions on medical imaging* 35.5, pp. 1285–1298.

Shortis, Mark R, Timothy A Clarke, and Tim Short (1994). "Comparison of some techniques for the subpixel location of discrete target images". In: *Videometrics III*. Vol. 2350. SPIE, pp. 239–250.

Shrivastava, Ashish et al. (2016). "Learning from Simulated and Unsupervised Images through Adversarial Training". In: *CoRR* abs/1612.07828. arXiv: 1612.07828. URL: http://arxiv.org/abs/1612.07828.

Simonyan, Karen, Andrea Vedaldi, and Andrew Zisserman (2013). *Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps*. arXiv: 1312.6034 [cs.CV].

Simonyan, Karen and Andrew Zisserman (2015). "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *International Conference on Learning Representations*.

Sims, Shane D and Cristina Conati (2020). "A neural architecture for detecting user confusion in eye-tracking data". In: *Proceedings of the 2020 International Conference on Multimodal Interaction*, pp. 15–23.

Sriram, Harshinee, Cristina Conati, and Thalia Field (Oct. 2023). "Classification of Alzheimer's Disease with Deep Learning on Eye-tracking Data". In: *INTERNATIONAL CONFERENCE ON MULTIMODAL INTERACTION*. ACM. DOI: 10.1145/3577190.3614149. URL: https://doi.org/10.1145%2F3577190.3614149.

Stampe, Dave M (1993). "Heuristic filtering and reliable calibration methods for video-based pupil-tracking systems". In: *Behavior Research Methods, Instruments, & Computers* 25.2, pp. 137–142.

Startsev, Mikhail and Michael Dorr (2019). "Classifying autism spectrum disorder based on scanpaths and saliency". In: *2019 IEEE International*

Conference on Multimedia & Expo Workshops (ICMEW). IEEE, pp. 633–636.

Steil, Julian et al. (2019). "Privacy-aware eye tracking using differential privacy". In: *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications*, pp. 1–9.

Stein, Isabell, Helen Jossberger, and Hans Gruber (2022). "Investigating visual expertise in sculpture: A methodological approach using eye tracking". In: *Journal of Eye Movement Research* 15.2.

Strukelj, Alexander and Diederick C Niehorster (2018). "One page of text: Eye movements during regular and thorough reading, skimming, and spell checking". In: *Journal of Eye Movement Research* 11.1.

Sugano, Yusuke, Yasuyuki Matsushita, and Yoichi Sato (2014). "Learning-by-Synthesis for Appearance-Based 3D Gaze Estimation". In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1821–1828. DOI: 10.1109/CVPR.2014.235.

Sugano, Yusuke, Yasunori Ozaki, et al. (2014). "Image preference estimation with a data-driven approach: A comparative study between gaze and image features". In: *Journal of Eye Movement Research* 7.3.

Tafaj, Enkelejda et al. (2013). "Online classification of eye tracking data for automated analysis of traffic hazard perception". In: *International Conference on Artificial Neural Networks*. Springer, pp. 442–450.

Tao, Y. and M. Shyu (July 2019). "SP-ASDNet: CNN-LSTM Based ASD Classification Model using Observer ScanPaths". In: *2019 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, pp. 641–646. DOI: 10.1109/ICMEW.2019.00124.

Tseng, Po-He et al. (2013). "High-throughput classification of clinical populations from natural viewing eye movements". In: *Journal of Neurology* 260.1, pp. 275–284.

Unger, Moshe, Michel Wedel, and Alexander Tuzhilin (2023). "Predicting consumer choice from raw eye-movement data using the RETINA deep learning architecture". In: *Data Mining and Knowledge Discovery*, pp. 1–32.

Valliappan, Nachiappan et al. (2020). "Accelerating eye movement research via accurate and affordable smartphone eye tracking". In: *Nature Communications* 11.

Van der Gijp, A et al. (2017). "How visual search relates to visual diagnostic performance: a narrative systematic review of eye-tracking research in radiology". In: *Advances in Health Sciences Education* 22.3, pp. 765–787.

Venuprasad, Pranav et al. (2020). "Analyzing gaze behavior using object detection and unsupervised clustering". In: *ACM Symposium on Eye Tracking Research and Applications*, pp. 1–9.

Vortmann, Lisa-Marie et al. (2021). "Imaging Time Series of Eye Tracking Data to Classify Attentional States". In: *Frontiers in Neuroscience* 15, p. 664490.

Wang, Shoujin et al. (2016). "Training deep neural networks on imbalanced data sets". In: *2016 International Joint Conference on Neural Networks (IJCNN)*, pp. 4368–4374. DOI: 10.1109/IJCNN.2016.7727770.

Wang, Yanxia, Jingyi Wang, and Ping Guo (2023). "Eye-UNet: A UNet-based network with attention mechanism for low-quality human eye image segmentation". In: *Signal, Image and Video Processing* 17.4, pp. 1097–1103.

Wedel, Michel, Rik Pieters, and Ralf van der Lans (2023). "Modeling eye movements during decision making: A review". In: *psychometrika* 88.2, pp. 697–729.

Weiss, Karl, Taghi M Khoshgoftaar, and DingDing Wang (2016). "A survey of transfer learning". In: *Journal of Big data* 3.1, pp. 1–40.

Wood, Erroll et al. (Dec. 2015a). "Rendering of Eyes for Eye-Shape Registration and Gaze Estimation". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

— (2015b). "Rendering of Eyes for Eye-Shape Registration and Gaze Estimation". In: *CoRR* abs/1505.05916. arXiv: 1505.05916. URL: http://arxiv.org/abs/1505.05916.

Wu, Ruei-Jr et al. (2022). "High-resolution eye-tracking via digital imaging of Purkinje reflections". In: *bioRxiv*. DOI: 10.1101/2022.08.16.504076. eprint: https://www.biorxiv.org/content/early/2022/08/16/2022.08.16.504076.full.pdf. URL: https://www.biorxiv.org/content/early/2022/08/16/2022.08.16.504076.

Wu, Zhengyang et al. (2019). "Eyenet: A multi-task deep network for off-axis eye gaze estimation". In: *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. IEEE, pp. 3683–3687.

Xiong, Jianghao et al. (2021). "Augmented reality and virtual reality displays: emerging technologies and future perspectives". In: *Light: Science & Applications* 10.1, pp. 1–30.

Xu, Peng, Xiatian Zhu, and David A Clifton (2023). "Multimodal learning with transformers: A survey". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Yarbus, AL (1967). "Eye Movements and Vision (B. Haigh, Trans.) Plenum Press". In: *New York*.

Yin, Yuehan et al. (2021). "Classification of eye tracking data in visual information processing tasks using convolutional neural networks and feature engineering". In: *SN Computer Science* 2.2, pp. 1–26.

Yiu, Yuk-Hoi et al. (2019). "DeepVOG: Open-source pupil segmentation and gaze estimation in neuroscience using deep learning". In: *Journal of neuroscience methods* 324, p. 108307.

Zhang, Xucong et al. (2015). "Appearance-Based Gaze Estimation in the Wild". In: *CoRR* abs/1504.02863. arXiv: 1504.02863. URL: http://arxiv.org/abs/1504.02863.

Zhou, Bolei et al. (2015). *Learning Deep Features for Discriminative Localization*. arXiv: 1512.04150 [cs.CV].

Zhou, Ce et al. (2023). "A comprehensive survey on pretrained foundation models: A history from bert to chatgpt". In: *arXiv preprint arXiv:2302.09419*.

Zhu, Yitan et al. (2021). "Converting tabular data into images for deep learning with convolutional neural networks". In: *Scientific reports* 11.1, pp. 1–11.

Zimmermann, Jan et al. (2016). "Oculomatic: High speed, reliable, and accurate open-source eye tracking for humans and non-human primates". In: *Journal of Neuroscience Methods* 270, pp. 138–146. ISSN: 0165-0270. DOI: 10.1016/j.jneumeth.2016.06.016.

Zonca, Joshua, Giorgio Coricelli, and Luca Polonio (2019). "Does exposure to alternative decision rules change gaze patterns and behavioral strategies in games?" In: *Journal of the Economic Science Association* 5.1, pp. 14–25.

— (2020). "Gaze patterns disclose the link between cognitive reflection and sophistication in strategic interaction". In: *Judgment and Decision Making* 15.2, pp. 230–245.

Zonca, Joshua, Alexander Vostroknutov, et al. (2021). "Timing of social feedback shapes observational learning in strategic interaction". In: *Scientific Reports* 11.1, pp. 1–12.