Link to data:
https://atreus.informatik.uni-tuebingen.de/seafile/d/8e2ab8c3fdd444e1a135/?p=%2F&mode=list

CBF: Circular binary features for robust and real-time pupil center detection

Wolfgang Fuhl
University Tuebingen, Perception Engineering
Tuebingen, Baden-Wuerttemberg
wolfgang.fuhl@uni-tuebingen.de

David Geisler
University Tuebingen, Perception Engineering
Tuebingen, Baden-Wuerttemberg
david.geisler@uni-tuebingen.de

Thiago Santini
University Tuebingen, Perception Engineering
Tuebingen, Baden-Wuerttemberg
thiago.santini@uni-tuebingen.de

Tobias Appel
University Tuebingen, LEAD Graduate School
Tuebingen, Baden-Wuerttemberg
tobias.appel@uni-tuebingen.de

Wolfgang Rosenstiel
University Tuebingen, Technical Computer Science
Tuebingen, Baden-Wuerttemberg
wolfgang.rosenstiel@uni-tuebingen.de

Enkelejda Kasneci
University Tuebingen, Perception Engineering
Tuebingen, Baden-Wuerttemberg
enkelejda.kasneci@uni-tuebingen.de

ABSTRACT

Modern eye tracking systems rely on fast and robust pupil detection, and several algorithms have been proposed for eye tracking under real world conditions. In this work, we propose a novel binary feature selection approach that is trained by computing conditional distributions. These features are scalable and rotatable, allowing for distinct image resolutions, and consist of simple intensity comparisons, making the approach robust to different illumination conditions as well as rapid illumination changes. The proposed method was evaluated on multiple publicly available data sets, considerably outperforming state-of-the-art methods, and being real-time capable for very high frame rates. Moreover, our method is designed to be able to sustain pupil center estimation even when typical edge-detection-based approaches fail – e.g., when the pupil outline is not visible due to occlusions from reflections or eye lids / lashes. As a consequece, it does not attempt to provide an estimate for the pupil outline. Nevertheless, the pupil center suffices for gaze estimation – e.g., by regressing the relationship between pupil center and gaze point during calibration.

CCS CONCEPTS

• **Mathematics of computing** → **Probabilistic representations**; • **Computing methodologies** → *Machine learning approaches*; *Feature selection*; *Cross-validation*; *Image processing*;

KEYWORDS

Pupil detection, conditional distrubution, random ferns, machine learning, statistics, scaleable features, rotatable features

ETRA '18, June 14–17, 2018, Warsaw, Poland

ACM ISBN 978-1-4503-5706-7/18/06…$15.00
https://doi.org/10.1145/3204493.3204559

ACM Reference Format:
Wolfgang Fuhl, David Geisler, Thiago Santini, Tobias Appel, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2018. CBF: Circular binary features for robust and real-time pupil center detection. In *ETRA '18: ETRA '18: 2018 Symposium on Eye Tracking Research and Applications, June 14–17, 2018, Warsaw, Poland*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3204493.3204559

1 INTRODUCTION

Visual perception is the focus of many research areas like medicine, psychology, advertisement, autonomous driving, or application control. The main focus of these areas lies in the extraction and understanding of human visual perception and attention. Various manufacturers have provided such eye-tracking devices to measure and investigate the human viewing behavior in real-world and dynamic tasks [Instruments 2015; Kassner et al. 2014; Lange et al. 2006; Tobii 2015]. In general there are two main categories of eye-tracking systems: 1) Remote systems, which consist of at least one external camera recording the subject, and 2) head mounted eye-trackers, which are worn by the subject and record the eyes and field of view. For remote eye tracking, additional challenges have to be solved like face and eye detection as well as the determination of the head orientation ([Jian and Lam 2015; Jian et al. 2013]). Due to the large recording area of the camera, the extracted eye images have only low resolutions. Many approaches have been proposed for pupil detection in such images, including image hallucination [Jian et al. 2014], which is used to increase the resolution and, therefore, the gaze estimation accuracy. In contrast, the measurement of eye movements in head-mounted eye-trackers is based on recordings from the eye of the subject directly. Therefore, the camera is placed close to the eye, delivering high resolution images, and the eye region does not have to be detected. Since the the eye-tracker is head-worn, the camera recording the field of view moves with the subjects head, which also eliminates the head orientation determination.

In both systems, the gaze position of a person is typically determined based on the pupil center and a mapping function obtained during calibration. While there are other approaches that determine the gaze location directly from the eye image [Tonsen et al.

2017; Wood and Bulling 2014; Zhang et al. 2015], the pupil center based approach is currently the most developed and widely used. In order for this approach to be successful, the center of the pupil has to be detected accurately under various challenging conditions like reflections, changing illumination, make up, contact lenses, etc. [Schnipke and Todd 2000] summarized already in the year 2000 many difficulties occurring during eye tracker usage. Nowadays, in natural environments, many researchers report difficulties during their studies (e.g., driving [Braunagel et al. 2015; Kasneci 2013; Kasneci et al. 2015, 2014a; Liu et al. 2002; Trösterer et al. 2014], museum visit [Santini et al. 2018a], shopping [Kasneci et al. 2014b; Sippel et al. 2014], and walking around [Sugano and Bulling 2015]) regarding the gaze estimation quality caused by a erroneous pupil detection. Therefore, the data in such studies has to be manually corrected. Although this is a laborious and error-prone job, it is still feasible for offline analyssis; in contrast, online applications can not use such post processing approaches. Examples of online applications are driver assistance systems [Tafaj et al. 2013], gaze-based interaction [Majaranta and Bulling 2014; Stellmach and Dachselt 2012; Turner et al. 2014] and virtual reality interaction [Duchowski et al. 2002, 2001; Hillaire et al. 2008; Krapichler et al. 1998] (more applications can be found in [Duchowski 2002]).

2 RELATED WORK

A large body of algorithms for pupil detection has been proposed in the last years. Most approaches are developed for usage under laboratory conditions [Goni et al. 2004; Keil et al. 2010] where thresholds are applied to images recorded under near infrared illumination [Lin et al. 2010; Long et al. 2007; Peréz et al. 2003] to extract the pupil. For center estimation the center of mass is used. In [Zhu et al. 1999], another threshold based approach was used with the difference that the pupil is detected based on the shape of the extracted region. A similar approach is also used by the recently published algorithm SET [Javadi et al. 2015], which first extracts pupil pixels based on a luminance threshold. Afterwards, the shape of the thresholded area is extracted and compared against a sine curve. Another curvature based approach using isophotes was proposed in [Valenti and Gevers 2012] where the center with maximal isophote edge votes is used as pupil center. Similar to the isophotes approch an iterative circle detection method using isopohte edge pixel selection was proposed in [Marco et al. 2015]. The most famouse algorithm is Starburst [Li et al. 2005]. This algorithm sends out rays in multiple directions and collects all positions where the difference of consecutive pixels is higher than a threshold. The mean position is then calculated, and this step is repeated until convergence. Swirski et. al [Świrski et al. 2012] starts with a coarse positioning using Haar-like features and then refines the pupil center position. The intensity histogram of the coarse position is clustered using k-means followed by a modified RANSAC ellipse fit. In the open source software from Pupil Labs [Kassner et al. 2014], there is also an purely image based pupil detection algorithm. The algorithm also uses the Haar features from [Świrski et al. 2012] to estimate a coarse position. Afterwards edges are computed, filtered based on their surrounding intensity values, and collected as connected components [Suzuki et al. 1985]. Ellipses are fitted to subsets of component contours and rated based on their supporting edges and ellipse circumference. ExCuSe [Fuhl et al. 2015], ElSe [Fuhl et al. 2016b] and PuRe [Santini et al. 2018b] are edge based approaches that rely on edge filtering. The extracted edges are morphologically filtered to break up incorrect connections and the best edge is selected for ellipse fitting. ExCuSe [Fuhl et al. 2015] and ElSe [Fuhl et al. 2016b] provide alternative approaches for cases when edge detection is not applicable. PuRe in comparison is capable of selecting multiple edges for the final fitting and edge selection. The first convolutional-neural-network-based approach was proposed in [Fuhl et al. 2016a] and uses one CNN for coarse positioning and an additional one to refine the center estimate. Additionally, other approaches differing from the aforementioned ones are appearance-based methods, which directly compute the gaze location from the eye image [Tonsen et al. 2017; Wood and Bulling 2014; Zhang et al. 2015].

3 METHOD

Our method is based on random ferns, similar to [Ozuysal et al. 2010; Villamizar et al. 2010]. A fern consists of multiple binary decisions, which evaluate to one or zero. These results construct an index that is used to access a probability distribution. Figure 1

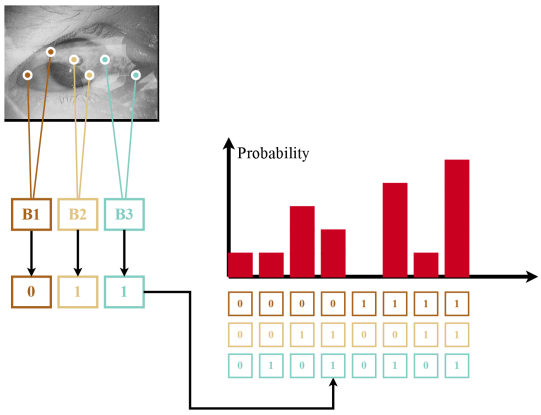


Figure 1: Three binary decisions and the indexing schema. The red bars represent the conditional distribution.

illustrates a fern, where B1, B2, and B3 are binary decisions. In our implementation, they are simply *larger- and smaller-than* pixel intensity comparison operations to make our features robust to illumination changes. The distribution, illustrated in red, holds the probability of an image position to be the pupil center under the conditions B1, B2 and B3. These ferns represent weak classifiers, which, combined, form a strong classifier. Therefore, the probability of multiple ferns is multiplied. In our implementation, we summed the results to avoid a multiplication with zero probability. Another difference of our implementation is the usage of two conditional distributions similar to [Villamizar et al. 2018]: One for the valid and one for the invalid examples. The final result for an index is, therefore, $\frac{P(Positive|B)}{P(Negative|B)+\epsilon}$, where ϵ is as small constant to avoid division by zero. These two distributions allow us to use gradient decent, which is the last part in training our approach. In the sequence, we will describe our circular binary feature selection procedure, the training, and our detector.

3.1 Features

In order to have a scalable and transformable set of binary features, we construct them circularly around one position. This means that, instead of selecting random image locations, we randomly select a) an angle (Figure 2 a), and b) two random distances (Figure 2 d1 and d2) along a vector along the selected angle. This feature construction is motivated by the shape of the pupil itself, whereas the binary comparisons are inspired by edge detection. Feature constructions is

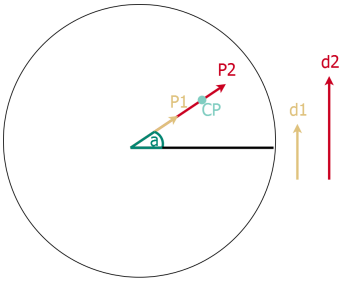


Figure 2: The feature construction. a is the angle, d1 and d2 the distances, and CP the center point between these distances.

illustrated in Figure 2. This procedure allows to scale and transform the features around the circle center for different image scales or distortions. In order for the scaling to be applicable, we have to evaluate each pixel position, where each pixel represents the circle center. This scaling, for example, can be performed either per distance or using the center between both distances (CP). We used the per distance scaling in our implementation. Another benefit from this scalable feature are the computational cost reduction by a pyramid-like detection. In other words, we start with a downscaled image and up scale in subsequent iterations.

3.2 Training

The main problem with training a conditional distribution is that overrepresented samples have a strong influence. In eye tracking data, this is a common issue because the eye remains at the same position while the user perceives the gazed area – i.e., during a fixation. To work around this problem, only a very small subset of all possible training data was selected to train a fern ($\approx$ 2000 images). This was done multiple times randomly. These training images originate from different datasets with different resolutions. Therefore, we selected the validation image size downscaled by a factor of two as image size for all training images. The downscaling was done to reduce computational costs.

Our training consisted of two main steps. The first step was the fern feature selection and preliminary training. In the second step, we combined these features into a strong classifier and refined the distributions using gradient decent.

In the first step, we randomly selected a subset of training images from the entire training data set and randomly evaluated the generated binary features on it. The ranking of these features was done by counting the outcome for valid and invalid image positions. The final rank was computed as $|(True_{valid}+False_{Invalid})-(False_{valid}+True_{Invalid})|$. Here, $True_{invalid}$ means that the binary decision evaluated to true, and the image position was invalid. The absolute value was used so that a high score in $(True_{valid} + False_{Invalid})$ or $(False_{valid} + True_{Invalid})$ is judged as a good feature as long as the opposite is low (instead swapping P1 and P2). The best ten percent features are then used as feature pool. For the ferns, we selected randomly $N = 20$ features. This is a memory consuming choice because each fern needs $4 * 2 * (2^N) \approx 8,4$ mega bytes. We did this to reduce the necessary amount of ferns in our final detector, which reduces the binary evaluation and, thus, runtime. The training of the distribution was done by randomly selecting one hundred invalid positions per image and forty nine valid samples (7×7 region surrounding the hand-labeled center). Each valid sample was weighted by their Euclidean distance to the labeled center ($\frac{distance}{maximaldistance} * 10$). The factor ten is used to compensate for the different amounts of samples between the invalid and valid distributions. For the second step of our detector, we used only the region surrounding the pupil (31×31) to collect invalid examples. Afterwards, all ferns were evaluated on the validation set, and the one with the best performance was selected. This was done multiple times with different training sets. This procedure generates a fern pool for the second step of our training.

The second step of our training consisted of selecting fern tuples from the fern pool. For each fern tuple, we applied batch based gradient decent. This means that we selected a batch of 100 images from the two training sets of the two ferns and evaluated those. For each failure, the positive distribution was increased with a specified learning rate $0.01-0.0001$ in the area surrounding the labeled center (7×7) and decreased at the selected wrong position (7×7). For the positive increase, we selected learning rates between $0.01 - 0.0001$, and, for the negative, this learning rate was doubled. A failure is if the coarse step has an Euclidean distance larger than fifteen and for the accuracy step if the Euclidean distance is larger than five. After each batch, the detector was evaluated on the validation data set. If the performance increased, the detector was saved, if not the training continued. After five consecutive performance decreasing batches the current distribution was dropped and the last one stored loaded.

3.3 Detector

For our detector, we employed a two step approach: one for the coarse positioning, and one for the accurate center estimation. The coarse positioning detector is applied to each image position. The maximum in the resulting probability map is afterwards used as starting position for the accurate center estimation. The accurate detector searches outgoing from this starting position in an $(31) \times (31)$ region. The resulting maximum is used as the pupil center.

4 LIMITATIONS

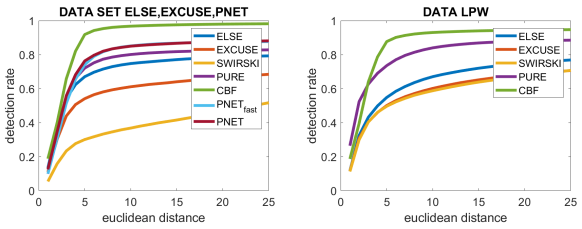
The training of ferns with a conditional distribution of size 2^{20} is a memory consuming process. Moreover, the evaluation of hundreds of thousands of features and the gradient decent of multiple detectors is also time consuming. Another issue with the scoring and training is that statistics have an high impact on the learning and selection. The same is true for the selection of the ferns based on their performance on the validation data set. If this validation data

only represents one challenge, the ferns will be selected accordingly. In direct comparison with the edge selection and ellipse fitting, the proposed method, similar to PupilNet, is generally less accurate because of the pixel voting.

5 EVALUATION

Figure 3 shows the detection rates for the data sets ExCuSe [Fuhl et al. 2015], ELSe [Fuhl et al. 2016b], PupilNet [Fuhl et al. 2016a] and labeled pupils in the wild [Tonsen et al. 2016]. The results for Pure and PupilNet where provided by their respective authors. As can be seen, the proposed approach outperforms the state-of-the-art algorithms. However, these results have to be taken with care. As previously mentioned, we only used a tiny subset of the entire training data to train our ferns, and the detectors consist only of two ferns per stage (coarse and accuracy). These decisions where made to have a method that is quick and resource-saving, but restrict the generalization over the entire data. For a generalized detector, more ferns have to be selected, and the validation, as well as the training data, has to contain a large variety of distinct challenges. Table 2 lists the runtime of all evaluated algorithms. For the other approaches, we used the runtime provided in [Fuhl et al. 2016a; Santini et al. 2018b]. As can be seen, the pyramid based approach is capable of providing a runtime below one millisecond. Table 1 shows data sets, in which we see the main application area of our approach. In particular, in cases when edge detection is not possible, such as nearly closed eyes, blurred images, or bad image quality. In the data set ExCuSe II, for examples, there are few examples with low contrast and nearly closed eyes. ElSe and ExCuSe can only cope with it because of their secondary approach. LPW 3 file 1 contains reflections that blur the image causing the edge detection to fail. In LPW 4 file 1, nearly closed eyes from an highly off axial angle are shown. Together with LPW 5 file 3, which contains the pupil mainly behind lenses borders, these shows how underrepresented many challenges are. This is also the reason that our method could not find good features for LPW 5 file 3.

6 RESULTS



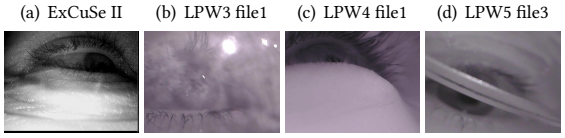
(a) (b)

Figure 3: The detection result as average per data set (each recording is weighted equally) for the labeled pupils in the wild, ExCuSe, ElSe, and PupilNet data sets.

7 CONCLUSION

We proposed a binary feature selection approach and a training procedure for random ferns. The main purpose of this paper is the demonstration of the applicability of random ferns for real-time pupil detection because the evaluated detectors are not generalized. The proposed approach outperforms the state-of-the-art but is rather limited to scenarios where edge detection is not possible due to its reduced accuracy. The low computational costs in combination with its robustness make it a valuable coarse positioning or backup method for scenarios like driving against the sun light.

Future research will go into the simplification of the training and a framework to combine detectors trained on different data sets. This would allow to refine the detector later on and reduce training time. The library for training and boosting as well as two general detectors (trained on the LPW and validated on the ExCuSe, ElSe and PupilNet data sets) can be downloaded at http://www.ti.uni-tuebingen.de/Pupil-detection.1827.0.html. The difference between both is the fern size (15 for the first and 20 for the second one). The runtime for both detectors is 0.5ms and they have detection rates of 78.9% and 91% for all images ($\leq$ 5 pixels). The limitations for both detectors is the RAM consumption due to the precomputed indexes and the conditional distribution size (3 GB and 9.5 GB).



Data	ElSe	ExCuSe	PupilNet	PuRe	CBF
(a)	65	40	80	29	***99***
(b)	8	1	–	30	***92***
(c)	7	3	–	9	***52***
(d)	1	2	–	1	***20***

Table 1: Detection rate for some of the data sets where edge detection is not applicable.

Downscaled by 2 (384 × 288)	2.4ms
Pyramid approach (384 × 288)	0.44ms
Downscaled by 2 (640 × 480)	6.8ms
Pyramid approach (640 × 480)	0.46ms
ExCuSe	2ms
ElSe	6.5ms
PuRe	5.5ms
PupilNet	7ms
Swirski	4ms

Table 2: The runtime on one CPU core of the proposed approach for the resolutions 640 × 480 **(LPW) and** 384 × 288 **(ExCuSe,ElSe,PNet) with precomputed indexes in comparison to the competitor algorithms.**

REFERENCES

C. Braunagel, E. Kasneci, W. Stolzmann, and W. Rosenstiel. 2015. Driver-Activity Recognition in the Context of Conditionally Autonomous Driving. In *International Conference on Intelligent Transportation Systems*. IEEE, 1652–1657. https://doi.org/10.1109/ITSC.2015.268

Andrew T Duchowski. 2002. A breadth-first survey of eye-tracking applications. *Behavior Research Methods, Instruments, & Computers* 34, 4 (2002), 455–470.

Andrew T Duchowski, Eric Medlin, Nathan Cournia, Anand Gramopadhye, Brian Melloy, and Santosh Nair. 2002. 3D eye movement analysis for VR visual inspection training. In *Proceedings of the 2002 symposium on Eye tracking research & applications*. ACM, 103–110.

Andrew T Duchowski, Eric Medlin, Anand Gramopadhye, Brian Melloy, and Santosh Nair. 2001. Binocular eye tracking in VR for visual inspection training. In *Proceedings of the ACM symposium on Virtual reality software and technology*. ACM, 1–8.

Wolfgang Fuhl, Thomas Kübler, Katrin Sippel, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2015. ExCuSe: Robust Pupil Detection in Real-World Scenarios. In *Computer Analysis of Images and Patterns*, George Azzopardi and Nicolai Petkov (Eds.). Springer International Publishing, Cham, 39–51.

Wolfgang Fuhl, Thiago Santini, Gjergji Kasneci, and Enkelejda Kasneci. 2016a. PupilNet: Convolutional Neural Networks for Robust Pupil Detection. *CoRR* abs/1601.04902 (2016). arXiv:1601.04902 http://arxiv.org/abs/1601.04902

Wolfgang Fuhl, Thiago C. Santini, Thomas Kübler, and Enkelejda Kasneci. 2016b. ElSe: Ellipse Selection for Robust Pupil Detection in Real-world Environments. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications (ETRA '16)*. ACM, New York, NY, USA, 123–130.

Sonia Goni, Javier Echeto, Arantxa Villanueva, and Rafael Cabeza. 2004. Robust algorithm for pupil-glint vector detection in a video-oculography eyetracking system. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*. IEEE.

Sébastien Hillaire, Anatole Lécuyer, Rémi Cozot, and Géry Casiez. 2008. Using an eye-tracking system to improve camera motions and depth-of-field blur effects in virtual environments. In *Virtual Reality Conference, 2008. VR'08. IEEE*. IEEE, 47–50.

SensoMotoric Instruments. 2015. SMI Eye Tracking Glasses 2 Wireless. *Mobile eye tracking made easy, robust, efficient and versatile. URL: http://www. eyetracking-glasses. com/fileadmin/user_upload/documents/smi _ etg2w _ flyer _ naturalgaze. pdf (visited on 12/22/2015)* (2015).

Amir-Homayoun Javadi, Zahra Hakimi, Morteza Barati, Vincent Walsh, and Lili Tcheang. 2015. SET: a pupil detection method using sinusoidal approximation. *Frontiers in neuroengineering* 8 (2015).

Muwei Jian and Kin-Man Lam. 2015. Simultaneous Hallucination and Recognition of Low-Resolution Faces Based on Singular Value Decomposition. *Circuits and Systems for Video Technology, IEEE Transactions on* 25, 11 (2015), 1761–1772.

Muwei Jian, Kin-Man Lam, and Junyu Dong. 2013. A novel face-hallucination scheme based on singular value decomposition. *Pattern Recognition* 46, 11 (2013), 3091–3102.

Muwei Jian, Kin-Man Lam, and Junyu Dong. 2014. Facial-feature detection and localization based on a hierarchical scheme. *Information Sciences* 262 (2014), 1–14.

E. Kasneci. 2013. *Towards the Automated Recognition of Assistance Need for Drivers with Impaired Visual Field.* Ph.D. Dissertation. University of Tübingen, Wilhelmstr. 32, 72074 Tübingen. http://tobias-lib.uni-tuebingen.de/volltexte/2013/7033

Enkelejda Kasneci, Gjergji Kasneci, Thomas C. Kübler, and Wolfgang Rosenstiel. 2015. *Artificial Neural Networks: Methods and Applications in Bio-/Neuroinformatics*. Springer International Publishing, Chapter Online Recognition of Fixations, Saccades, and Smooth Pursuits for Automated Analysis of Traffic Hazard Perception, 411–434. https://doi.org/10.1007/978-3-319-09903-3_20

E. Kasneci, K. Sippel, K. Aehling, M. Heister, W. Rosenstiel, U. Schiefer, and E. Papageorgiou. 2014a. Driving with Binocular Visual Field Loss? A Study on a Supervised On-road Parcours with Simultaneous Eye and Head Tracking. *Plos One* 9, 2 (2014), e87470.

Enkelejda Kasneci, Katrin Sippel, Martin Heister, Katrin Aehling, Wolfgang Rosenstiel, Ulrich Schiefer, and Elena Papageorgiou. 2014b. Homonymous Visual Field Loss and Its Impact on Visual Exploration: A Supermarket Study. *TVST* 3, 6 (2014).

Moritz Kassner, William Patera, and Andreas Bulling. 2014. Pupil: an open source platform for pervasive eye tracking and mobile gaze-based interaction. In *Adj. Proc. of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*. 1151–1160. https://doi.org/10.1145/2638728.2641695

Andrea Keil, Georgia Albuquerque, Kai Berger, and Marcus Andreas Magnor. 2010. Real-Time Gaze Tracking with a Consumer-Grade Video Camera. (2010).

Christian Krapichler, Michael Haubner, Rolf Engelbrecht, and Karl-Hans Englmeier. 1998. VR interaction techniques for medical imaging applications. *Computer methods and programs in biomedicine* 56, 1 (1998), 65–74.

Christian Lange, Jae-Woo Yoo, M Wohlfarter, and H Bubb. 2006. Dikablis (digital wireless gaze tracking system)–operation mode and evaluation of the human machine interaction. (2006), 121–129.

Dongheng Li, David Winfield, and Derrick J Parkhurst. 2005. Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches. In *Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on*. IEEE, 79–79.

Lin Lin, Lin Pan, LiFang Wei, and Lun Yu. 2010. A robust and accurate detection of pupil images. In *Biomedical Engineering and Informatics (BMEI), 2010 3rd International Conference on*, Vol. 1. IEEE, 70–74.

Xia Liu, Fengliang Xu, and Kikuo Fujimura. 2002. Real-time eye detection and tracking for driver observation under various light conditions. In *Intelligent Vehicle Symposium, 2002. IEEE*, Vol. 2. IEEE, 344–351.

Xindian Long, Ozan K Tonguz, and Alex Kiderman. 2007. A high speed eye tracking system with robust pupil center estimation algorithm. In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*. IEEE.

Päivi Majaranta and Andreas Bulling. 2014. *Eye Tracking and Eye-Based Human-Computer Interaction*. Springer Publishing London, 39–65. https://doi.org/10.1007/978-1-4471-6392-3_3

Tommaso De Marco, Dario Cazzato, Marco Leo, and Cosimo Distante. 2015. Randomized circle detection with isophotes curvature analysis. *Pattern Recognition* 48, 2 (2015), 411 – 421. https://doi.org/10.1016/j.patcog.2014.08.007

Mustafa Ozuysal, Michael Calonder, Vincent Lepetit, and Pascal Fua. 2010. Fast keypoint recognition using random ferns. *IEEE transactions on pattern analysis and machine intelligence* 32, 3 (2010), 448–461.

Antonio Peréz, ML Cordoba, A Garcia, Rafael Méndez, ML Munoz, José Luis Pedraza, and F Sanchez. 2003. A precise eye-gaze detection and tracking system. (2003).

Thiago Santini, Hanna Brinkmann, Luise Reitstätter, Helmut Leder, Raphael Rosenberg, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2018a. The Art of Pervasive Eye Tracking. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications (ETRA) – Adjunct: PETMEI*. ACM.

Thiago Santini, Wolfgang Fuhl, and Enkelejda Kasneci. 2018b. PuRe: Robust pupil detection for real-time pervasive eye tracking. *Computer Vision and Image Understanding* (Feb 2018). https://doi.org/10.1016/j.cviu.2018.02.002

Susan K Schnipke and Marc W Todd. 2000. Trials and tribulations of using an eye-tracking system. In *CHI'00 extended abstracts on Human factors in computing systems*. ACM.

Katrin Sippel, Enkelejda Kasneci, Kathrin Aehling, Martin Heister, Wolfgang Rosenstiel, Ulrich Schiefer, and Elena Papageorgiou. 2014. Binocular Glaucomatous Visual Field Loss and Its Impact on Visual Exploration - A Supermarket Study. *PLoS ONE* 9, 8 (2014), e106089. https://doi.org/10.1371/journal.pone.0106089

Sophie Stellmach and Raimund Dachselt. 2012. Look & touch: gaze-supported target acquisition. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2981–2990.

Yusuke Sugano and Andreas Bulling. 2015. Self-Calibrating Head-Mounted Eye Trackers Using Egocentric Visual Saliency. In *Proc. of the 28th ACM Symposium on User Interface Software and Technology (UIST)* (2015-11-05). 363–372. https://doi.org/10.1145/2807442.2807445

Satoshi Suzuki et al. 1985. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing* 30, 1 (1985), 32–46.

Lech Świrski, Andreas Bulling, and Neil Dodgson. 2012. Robust real-time pupil tracking in highly off-axis images. In *Proceedings of the Symposium on Eye Tracking Research & Applications (ETRA)*. ACM, 173–176. https://doi.org/10.1145/2168556.2168585

E. Tafaj, T. Kübler, G. Kasneci, W. Rosenstiel, and M. Bogdan. 2013. Online Classification of Eye Tracking Data for Automated Analysis of Traffic Hazard Perception. In *Artificial Neural Networks and Machine Learning ICANN 2013*, Vol. 8131. Springer Berlin Heidelberg, 442–450.

Tobii. 2.(2015). *Tobii glasses* 2 (2015).

Marc Tonsen, Julian Steil, Yusuke Sugano, and Andreas Bulling. 2017. InvisibleEye: Mobile Eye Tracking Using Multiple Low-Resolution Cameras and Learning-Based Gaze Estimation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 106.

Marc Tonsen, Xucong Zhang, Yusuke Sugano, and Andreas Bulling. 2016. Labelled Pupils in the Wild: A Dataset for Studying Pupil Detection in Unconstrained Environments. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications (ETRA '16)*. ACM, New York, NY, USA, 139–142. https://doi.org/10.1145/2857491.2857520

Sandra Trösterer, Alexander Meschtscherjakov, David Wilfinger, and Manfred Tscheligi. 2014. Eye Tracking in the Car: Challenges in a Dual-Task Scenario on a Test Track. In *Proceedings of the 6th AutomotiveUI*. ACM.

Jayson Turner, Andreas Bulling, Jason Alexander, and Hans Gellersen. 2014. Cross-device Gaze-supported Point-to-point Content Transfer. In *Proceedings of the ACM International Symposium on Eye Tracking Research & Applications (ETRA)*. 19–26. https://doi.org/10.1145/2578153.2578155

Roberto Valenti and Theo Gevers. 2012. Accurate Eye Center Location through Invariant Isocentric Patterns. *Transactions on pattern analysis and machine intelligence* 34, 9 (2012), 1785–1798.

Michael Villamizar, Juan Andrade-Cetto, Alberto Sanfeliu, and Francesc Moreno-Noguer. 2018. Boosted Random Ferns for Object Detection. *IEEE transactions on pattern analysis and machine intelligence* 40, 2 (2018), 272–288.

Michael Villamizar, Francesc Moreno-Noguer, Juan Andrade-Cetto, and Alberto Sanfeliu. 2010. Efficient rotation invariant object detection using boosted random ferns. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 1038–1045.

Erroll Wood and Andreas Bulling. 2014. EyeTab: Model-based gaze estimation on unmodified tablet computers. In *Proc. of the 8th Symposium on Eye Tracking Research & Applications (ETRA)*. 207–210. https://doi.org/10.1145/2578153.2578185

Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. 2015. Appearance-based gaze estimation in the wild. In *Proceedings of the IEEE Conference on Computer*

Vision and Pattern Recognition. 4511–4520.
Danjie Zhu, Steven T Moore, and Theodore Raphan. 1999. Robust pupil center detection using a curvature algorithm. *Computer methods and programs in biomedicine* 59, 3 (1999), 145–157.