Tiny convolution, decision tree, and binary neuronal networks for robust and real time pupil outline estimation

Wolfgang Fuhl
Human-Computer Interaction
University of Tübingen
wolfgang.fuhl@uni-tuebingen.de

Hong Gao
Human-Computer Interaction
University of Tübingen
hong.gao@uni-tuebingen.de

Enkelejda Kasneci
Human-Computer Interaction
University of Tübingen
enkelejda.kasneci@uni-tuebingen.de

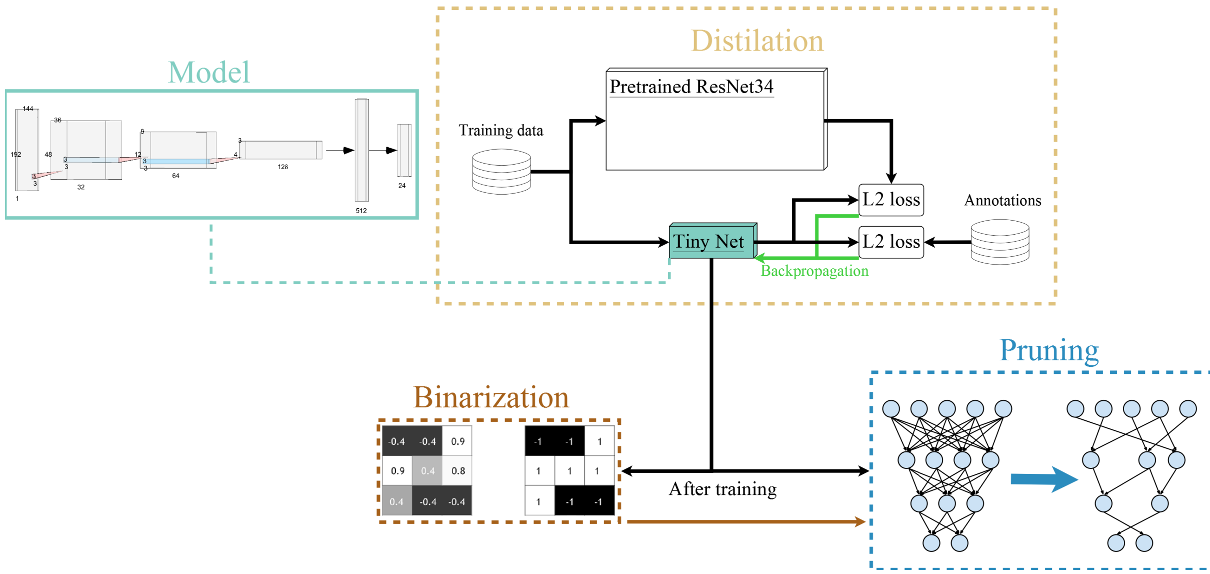

Figure 1: The used process to train tiny models and improve their runtime as well as the used model.

ABSTRACT

In this work, we compare the use of convolution, binary, and decision tree layers in neural networks for the estimation of pupil landmarks. These landmarks are used for the computation of the pupil ellipse and have proven to be effective in previous research. The evaluated structure of the neural networks is the same for all layers and as small as possible to ensure a real-time application. The evaluations include the accuracy of the ellipse determination based on the Jaccard Index and the pupil center. Furthermore, the CPU runtime is considered to make statements about the real-time usability. The trained models are also optimized using pruning to improve the runtime. These optimized nets are also evaluated with respect to the Jaccard index and the accuracy of the pupil center estimation. Link to the framework and models.

ETRA '20 Short Papers, June 2–5, 2020, Stuttgart, Germany

ACM ISBN 978-1-4503-7134-6/20/06…$15.00
https://doi.org/10.1145/3379156.3391347

CCS CONCEPTS

• **Computing methodologies → Machine learning**; **Image processing**; *Shape analysis*;

KEYWORDS

Eye tracking, pupil ellipse, pupil center, neuronal network, binarization, pruning, quantization, decision tree

ACM Reference Format:
Wolfgang Fuhl, Hong Gao, and Enkelejda Kasneci. 2020. Tiny convolution, decision tree, and binary neuronal networks for robust and real time pupil outline estimation. In *Symposium on Eye Tracking Research and Applications (ETRA '20 Short Papers), June 2–5, 2020, Stuttgart, Germany.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3379156.3391347

1 INTRODUCTION

Eye tracking is finding more and more new areas of application such as driver monitoring [Braunagel et al. 2017; Liu et al. 2002], virtual reality [Duchowski et al. 2000; Guenter et al. 2012; Patney et al. 2016], augmented reality [Ishimaru et al. 2014; Pfeiffer and Renner 2014], surgery [Eivazi et al. 2016; Fuhl et al. 2016e; Oltean et al. 2001], market research [Hervet et al. 2011; Wedel and Pieters 2008], self-diagnostic systems [Anderson and Colombo 2009; Karahan et al. 2017; Tennant 1988], human computer interaction [Bulling 2016;

Bulling and Gellersen 2010], sharing expert knowledge [Kübler et al. 2015; Reingold and Sheridan 2011] and many more.

One challenge that this diverse field of application brings with it is the need for a non-invasive solution that exists through image-based eye tracking [Duchowski 2002]. However, different and new challenges arise due to lighting conditions, imaging techniques like RGB and NIR, but also due to the diversity of people [Fuhl et al. 2016f; Tonsen et al. 2016]. Another challenge that faces us today is the use of eye tracking on mobile devices. This is compounded by the challenges of the limited computing resources available on mobile devices and the limited runtime of the devices under full load. This resource restriction leads to the fact that still classical algorithms are used on mobile devices [Fuhl et al. 2016b, 2017a, 2016d; Santini et al. 2018]. These have the disadvantage that a constant runtime cannot be guaranteed and that they perform significantly worse than machine learning approaches under everyday conditions [Fuhl et al. 2019a, 2018c, 2019b]. In addition, classical algorithms have the disadvantage that they cannot be adapted to new challenges through training. Usually parts or the whole algorithm has to be revised or redesigned. The use of modern machine learning methods has already delivered significant improvements in pupil recognition [Fuhl et al. 2019a, 2018c, 2019b] but there has been little development in the area of real-time capable algorithms on limited resources.

In this work, we deal with resource-saving approaches of neural networks for pupil recognition. This is due to the fact that neural networks have a constant runtime, the training is simple compared to other machine learning approaches, and is not as susceptible to unbalanced data. On the one hand, we focus on the real time capability on only one CPU core. Furthermore, our evaluation evaluates individual challenges separately to better assess the applicability of the algorithms. For all evaluations, the runtime as well as the pupil center and pupil area are used. In the following, a summarizing list of key points is given for a a quick overview.

1. Trained models for real time pupil ellipse detection.
2. Runtime optimized Framework to train and use the models.
3. Evaluation for different challenges separately.
4. Evaluation regarding runtime, pupil center, and pupil area.

2 RELATED WORK

Work in the field of pupil detection is mainly concerned with the issue of robust and reliable detections. For this purpose, there are already a number of rule-based approaches that have been summarized for head-mounted eye tracking [Fuhl et al. 2016f; Tonsen et al. 2016]. Images for head-mounted eye tracking differ strongly from images for remote eye tracking, which is why these areas were considered separately over a long period of time [Fuhl et al. 2016a]. Their main differences are the lower resolution of the eye area for remote eye tracking as well as a changing perspective due to the movement of the head which leads to partial occlusions of one eye by the nose for example [Fuhl et al. 2018a].

In this work, we deal with pupil detection on images regarding head mounted eye tracking. Therefore, already published approaches in this field are described in detail in the following. The first major breakthrough in this area for pupil detection was the use of edges [Świrski et al. 2012]. Since edge images contain a certain amount of noise, filtering methods were introduced which make it easier to detect the pupil in them [Fuhl et al. 2015, 2016d]. Based on this, methods for edge combination were presented to further improve the detection rate [Fuhl et al. 2016d; Santini et al. 2018]. The disadvantage of the purely edge-based methods is the exact positioning of the edges, which is strongly influenced by motion blur alone. To overcome this disadvantage, there were several approaches like Blob Detection [Fuhl et al. 2016d] and adaptive thresholding [Haro et al. 2000]. Based on the segmented image over a threshold, the detection rate could be further improved by splitting the segment into sub-segments [Javadi et al. 2015]. Each of these segments is then evaluated and the outline of the pupil is determined by several good subsegments [Javadi et al. 2015]. In the field of machine learning, there were also approaches for pupil detection. In the field of neural networks there was a window-based approach [Fuhl et al. 2016c, 2017b] which even fulfilled the real time runtime on a single CPU core. Window-based in this context means that an image is divided into small partial images and each partial image is classified individually. Other approaches were based on transposed convolution layers and generated segmentations [Fuhl et al. 2019a; Vera-Olmos and Malpica 2017; Vera-Olmos et al. 2019; Yiu et al. 2019]. A regression with integrated landmark detection was also presented which was trained in combination with a segmentation [Fuhl et al. 2019b]. Further approaches which are executable on a CPU core in real time compared to neural networks for landmark detection and segmentation are CBF [Fuhl et al. 2018c] and BORE [Fuhl et al. 2018b]. CBF [Fuhl et al. 2018c] is based on decision trees and a circular selection of features. BORE [Fuhl et al. 2018b], on the other hand, is based on an optimization procedure that can learn unsupervised and uses circularly oriented features. This unsupervised optimization is based on the selection of the best edges for a circular or elliptical object.

In this work, we evaluate neuronal network architectures for landmark detections which have a real-time executability on one CPU core as a requirement. With this, we follow the approaches of tiny architectures [Fuhl et al. 2016c] and landmark detection [Fuhl et al. 2019b] from the state of the art. In addition, we include the validity loss [Fuhl and Kasneci 2019] to obtain a quality measure for each landmark and therefore, a validity of the entire pupil.

3 METHOD

Figure 1 shows our initial model where we used the same architecture for the convolution and decision tree [Fuhl et al. 2020] based nets. Between the individual convolution blocks, max pooling is used. In addition, before the first fully connected layer of 1024 neurons, a 50% dropout is used to compress the learned weights. This compression of the weights is important for the pruning operation. The input of our model is a 144×192 gray scale image and the output are eight $x, y, validity$ triplets. To train this small architecture successfully, we used model distillation [Hinton et al. 2015]. Here the small model is additionally trained by a large pre-trained model (See Figure 1). This is done by including the output of the large mesh as an additional loss function. For the large model, we used a ResNet34 and trained it first on the training data. As a regression target for our model, we used eight landmarks, which lie on the pupil ellipse, as was also done in [Fuhl et al. 2019b]. In addition, we used the validity loss [Fuhl and Kasneci 2019] with which a validity value can be assigned to each pupil ellipse, which, in our case, corresponds to the mean value of all eight validity values. To determine the ellipse parameters, we applied the OpenCV ellipse fit to the eight landmarks.

Since this small architecture alone is not enough to get a real real-time model, we applied two additional techniques for runtime reduction. One is binarization [Courbariaux et al. 2014] and the other is clipping. With binarization, all weights greater than zero are set to the fixed value of one and the negative weights to the value minus one. This means that for execution, the sign only has to be reversed in the case of minus one. We only used this binarization for the convolution layers, since the model would otherwise become too imprecise. For pruning, however, we used an iterative approach [Castellano et al. 1997]. This approach deactivates a convolution or decision tree and checks the influence on the accuracy by evaluating the model on the training data. The same applies to the individual neurons in the penultimate fully connected layer. In each iteration, all possibilities were tested and the one with the least influence was selected. This was continued until only 20% of the original model was left.

To further improve the accuracy of the pruned nets (pruning is also applied to the models with binary weights), we have used fine tuning. Here, the learning rate of the convolution layers is set to zero and only the fully connected layers are trained. For this step, we have used a learning rate of 10^{-7}. For the general training, we used a fixed learning rate of 10^{-5}. Additionally, we used the Adam optimizer with the parameters 0.9 for momentum and 0.999 for the second momentum. Weight decay has been disabled. The batch size was set to 200 and the whole training ran for two weeks on a server, which corresponds to an epoch number of $\approx 100,000$. For data augmentation, random noise, random occlusions, image overlays as reflections, image shifts, and bluring are used and applied online to the data.

Table 1: Naming convention and description of the evaluated challenges and algorithms.

Challenge	Description
C0	Original images
C1, C2	Blur with filter size 9×9 and $\sigma = 1.1$, $\sigma = 1.2$
C3, C4, C5	10%, 20%, and 30% random noise
C6, C7, C8	Reflections with 20%, 40%, and 60% intensity
C9	C1, C3, and C6 combined
C10	C2, C4, and C7 combined
Algorithm	**Description**
ElSe	Edge filtering and selection
PuRe	Edge filtering and combination
ConvP	Pruned tiny neural network
BinP	Pruned and binarized tiny neural network
TreeP	Pruned tiny tree neural network

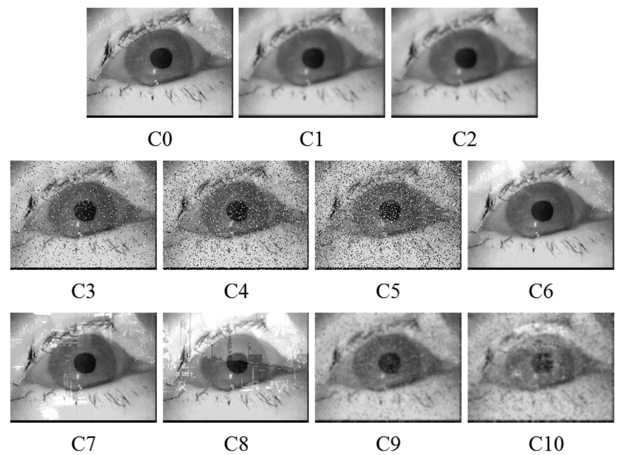

Figure 2: The different challenges applied to an image.

4 EVALUATION

For the training, we used the data from [Fuhl et al. 2019a]. These are segmentations of the known pupil in the wild dataset [Tonsen et al. 2016]. For the evaluation, we used the segmented data from [Fuhl et al. 2019b] and an additional 1,000,000 images from the same studies [Kasneci et al. 2014; Sippel et al. 2014]. Thus, our algorithms were evaluated on over 1.8 million images.

In table 1 the naming convention of each challenge evaluated and its description as well as parameters are shown. Examples for each challenge are shown in Figure 2. As you can see, none of the challenges used, nor the combinations (C9 and C10), pose a problem for a human being. The algorithms, however, behave differently.

Figure 3 shows the cumulative accuracy of the algorithms separately for each challenge. The x-axis is the Euclidean distance between the found and true pupil center. As you can see, the slight blurring is not a problem for any algorithm (C1 and C2), but even a slight addition of noise (C3, C4, and C5) is enough to reduce the performance of the classical algorithms (ElSe and Pure) by almost half. In the case of reflections, however, (C6, C7, and C8) only clearly visible reflections have an enormous influence (C8 see also Figure 2). Compared to the classical algorithms all neural networks are very robust against the challenges regarding the accuracy of the pupil center.

Figure 4 shows the Jaccard index ($\frac{GT \cap DT}{GT \cup DT}$, with GT = Ground Truth Ellipse and DT = Detected Ellipse) or mean intersection over union cumulatively for each challenge separately. For the Jaccard index, a value of 50% or higher is generally considered good. As you can see, the classic edge-based approaches are more accurate as long as they can handle the challenge. However, as with accuracy, it is obvious that only a small amount of noise (C3,C4, and C5) has a huge impact. For clearly visible reflections (C8), the tiny neural networks also have problems extracting a clean pupil ellipse. Overall, however, the neural networks are much more robust compared to the classical algorithms. If one now evaluates the methods with Figure 3 and Figure 4, one can clearly see that the conventional convolution is the most accurate and robust (ConvP). In second place are the decision tree based convolutions (TreeP) and finally the binarized convolutions (BinP).

However, if the runtime is also taken into account (Figure 5), this changes, because the decision-tree-based neural networks only require about one third of the runtime. In addition, one can see that

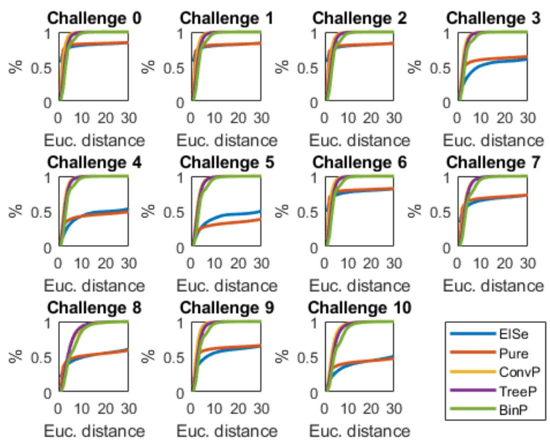

Figure 3: Cumulative accuracy in euclidean distance of the estimated pupil center to the ground truth for all algorithms and separated per challenge.

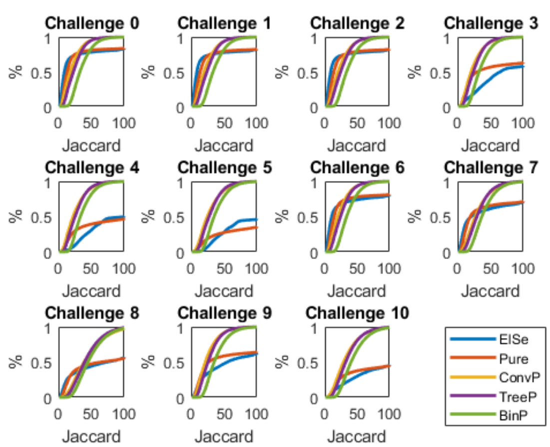

Figure 4: Jaccard index ($\frac{GT \cap DT}{GT \cup DT}$) between the estimated and ground truth pupil for all algorithms and separated per challenge.

the classical algorithms are constant on average but need significantly more computing time for certain images where many curved edges are present (Red crosses in Figure 5).

Table 2 serves to compare the validity signal of the algorithms. Our validity signal correlates with the accuracy of the result, since we have used the validity loss of [Fuhl and Kasneci 2019]. However, this does not apply to the classical algorithms (ElSe and Pure). Therefore, we decided to use an evaluation based on recall and precision. To apply percision and recall, we have used a validity threshold of 5 pixels. This means that if an estimated pupil center is closer than 5 pixels to the annotation, it is considered correct, otherwise it is considered false. For each algorithm, the threshold value for the validity signal was determined iteratively optimal to

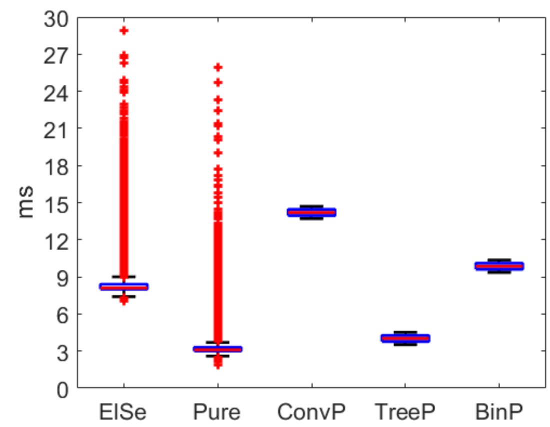

Figure 5: The runtime of ElSe and Pure in comparison to the pruned versions of the convolution, tree, and binary neuronal network using Whisker plots. Evaluated on one single CPU core (i5). The red crosses are outliers of the fitted normal distribution.

Table 2: Precision ($TP/AllPositives$) and Recall $TP/(TP+FN)$ with the optimal selected validity threshold evaluated over all challenges.

Algorithm	Precision	Recall
ElSe	100%	53.37%
Pure	100%	80.21%
ConvP	100%	89.56%
TreeP	100%	86.83%
BinP	100%	75.65%

achive the best recall result. ($TP/(TP+FN)$). This makes it easier for algorithms that generally have a worse accuracy (See Figure 3), but gives a good indication of the reliability of the validity signal. As you can see, all algorithms achieve a precision of 100%, which is because there were more correct than incorrect pupil centers and therefore, it was weighted heavier for the recall calculation. It can be seen that ConvP and TreeP are the best performers. Pure is also good but this is influenced by the much lower detection rate over all challenges.

5 CONCLUSION

In this work we presented different neural networks for real-time use on a single CPU core. For this, we used modern methods like distillation and pruning. As an additional comparison, we have binarized a neural network. All networks were evaluated with respect to their accuracy and reliability under different challenges and also the runtime was considered. The trained models and a runtime optimized framework are made available to the public together with this work.

ACKNOWLEDGMENTS

Work of the authors is supported by the Institutional Strategy of the University of Tübingen (Deutsche Forschungsgemeinschaft, ZUK 63).

REFERENCES

Christa J Anderson and John Colombo. 2009. Larger tonic pupil size in young children with autism spectrum disorder. *Developmental Psychobiology: The Journal of the International Society for Developmental Psychobiology* 51, 2 (2009), 207–211.

Christian Braunagel, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2017. Ready for take-over? A new driver assistance system for an automated classification of driver take-over readiness. *IEEE Intelligent Transportation Systems Magazine* 9, 4 (2017), 10–22.

Andreas Bulling. 2016. Pervasive attentive user interfaces. *Computer* 1 (2016), 94–98.

Andreas Bulling and Hans Gellersen. 2010. Toward mobile eye-based human-computer interaction. *IEEE Pervasive Computing* 9, 4 (2010), 8–12.

Giovanna Castellano, Anna Maria Fanelli, and Marcello Pelillo. 1997. An iterative pruning algorithm for feedforward neural networks. *IEEE transactions on Neural networks* 8, 3 (1997), 519–531.

Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. 2014. Training deep neural networks with low precision multiplications. *arXiv preprint arXiv:1412.7024* (2014).

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, Vinay Shivashankaraiah, Tim Rawls, Anand K Gramopadhye, Brian J Melloy, and Barbara Kanki. 2000. Binocular eye tracking in virtual reality for inspection training. In *Proceedings of the 2000 symposium on Eye tracking research & applications*. ACM, 89–96.

Shahram Eivazi, Roman Bednarik, Ville Leinonen, Mikael von und zu Fraunberg, and Juha E Jääskeläinen. 2016. Embedding an eye tracker into a surgical microscope: requirements, design, and implementation. *IEEE Sensors Journal* 16, 7 (2016), 2070–2078.

Wolfgang Fuhl, Nora Castner, Lin Zhuang, Markus Holzer, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2018a. MAM: Transfer learning for fully automatic video annotation and specialized detector creation. In *Egocentric Perception, Interaction and Computing Workshop (EPIC@ECCV)*.

Wolfgang Fuhl, Shahram Eivazi, Benedikt Hosp, Anna Eivazi, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2018b. BORE: boosted-oriented edge optimization for robust, real time remote pupil center detection. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*. ACM, 48.

Wolfgang Fuhl, David Geisler, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2019a. The applicability of Cycle GANs for pupil and eyelid segmentation, data generation and image refinement. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 0–0.

Wolfgang Fuhl, David Geisler, Thiago Santini, Tobias Appel, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2018c. CBF: circular binary features for robust and real-time pupil center detection. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*. ACM, 8.

Wolfgang Fuhl, David Geisler, Thiago Santini, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2016a. Evaluation of state-of-the-art pupil detection algorithms on remote eye images. In *International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 1716–1725.

Wolfgang Fuhl and Enkelejda Kasneci. 2019. Learning to validate the quality of detected landmarks. In *International Conference of Machine Vision (ICMV)*. SPIE.

Wolfgang Fuhl, G. Kasneci, W. Rosenstiel, and Enkelejda Kasneci. 2020. Training Decision Trees as Replacement for Convolution Layers. In *Conference on Artificial Intelligence (AAAI)*. AAAI.

Wolfgang Fuhl, Thomas Kübler, Katrin Sippel, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2015. Excuse: Robust pupil detection in real-world scenarios. In *International Conference on Computer Analysis of Images and Patterns*. Springer, 39–51.

Wolfgang Fuhl, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2019b. 500,000 images closer to eyelid and pupil segmentation. In *International Conference on Computer Analysis of Images and Patterns*. Springer, 336–347.

Wolfgang Fuhl, Thiago Santini, David Geisler, Thomas Kübler, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2016b. Eyes wide open? eyelid location and eye aperture estimation for pervasive eye tracking in real-world scenarios. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*. 1656–1665.

Wolfgang Fuhl, Thiago Santini, and Enkelejda Kasneci. 2017a. Fast and robust eyelid outline and aperture detection in real-world scenarios. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 1089–1097.

Wolfgang Fuhl, Thiago Santini, Gjergji Kasneci, and Enkelejda Kasneci. 2016c. PupilNet: convolutional neural networks for robust pupil detection. *arXiv preprint arXiv:1601.04902* (2016).

Wolfgang Fuhl, Thiago Santini, Gjergji Kasneci, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2017b. PupilNet v2. 0: Convolutional Neural Networks for CPU based real time Robust Pupil Detection. *arXiv preprint arXiv:1711.00112* (2017).

Wolfgang Fuhl, Thiago Santini, Carsten Reichert, Daniel Claus, Alois Herkommer, Hamed Bahmani, Katharina Rifai, Siegfried Wahl, and Enkelejda Kasneci. 2016e. Non-intrusive practitioner pupil detection for unmodified microscope oculars. *Computers in biology and medicine* 79 (2016), 36–44.

Wolfgang Fuhl, Thiago C Santini, Thomas Kübler, and Enkelejda Kasneci. 2016d. Else: Ellipse selection for robust pupil detection in real-world environments. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*. ACM, 123–130.

Wolfgang Fuhl, Marc Tonsen, Andreas Bulling, and Enkelejda Kasneci. 2016f. Pupil detection for head-mounted eye tracking in the wild: an evaluation of the state of the art. *Machine Vision and Applications* 27, 8 (2016), 1275–1288.

Brian Guenter, Mark Finch, Steven Drucker, Desney Tan, and John Snyder. 2012. Foveated 3D graphics. *ACM Transactions on Graphics (TOG)* 31, 6 (2012), 164.

Antonio Haro, Myron Flickner, and Irfan Essa. 2000. Detecting and tracking eyes by using their physiological properties, dynamics, and appearance. In *Computer Vision and Pattern Recognition*, Vol. 1. IEEE, 163–168.

Guillaume Hervet, Katherine Guérard, Sébastien Tremblay, and Mohamed Saber Chtourou. 2011. Is banner blindness genuine? Eye tracking internet text advertising. *Applied cognitive psychology* 25, 5 (2011), 708–716.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015).

Shoya Ishimaru, Kai Kunze, Koichi Kise, Jens Weppner, Andreas Dengel, Paul Lukowicz, and Andreas Bulling. 2014. In the blink of an eye: combining head motion and eye blink frequency for activity recognition with Google Glass. In *Proceedings of the 5th augmented human international conference*. ACM, 15.

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), 4.

Eyyup Karahan, Omer Karti, Aslı Koskderelioglu, Dilek Top Karti, Murat Uyar, Murat Yildirim Kale, and Muhtesem Gedizlioglu. 2017. Pupil cycle time: as indicator of visual pathway dysfunction in multiple sclerosis. *Acta Neurologica Belgica* 117, 1 (2017), 75–81.

Enkelejda Kasneci, Katrin Sippel, Kathrin Aehling, Martin Heister, Wolfgang Rosenstiel, Ulrich Schiefer, and Elena Papageorgiou. 2014. 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).

Thomas Kübler, Shahram Eivazi, and Enkelejda Kasneci. 2015. Automated visual scanpath analysis reveals the expertise level of micro-neurosurgeons. In *MICCAI workshop on interventional microscopy*. 1–8.

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.

Ioan T Oltean, John K Shimmick, and Terrance N Clapham. 2001. Eye tracking device for laser eye surgery using corneal margin detection. (Oct. 9 2001). US Patent 6,299,307.

Anjul Patney, Marco Salvi, Joohwan Kim, Anton Kaplanyan, Chris Wyman, Nir Benty, David Luebke, and Aaron Lefohn. 2016. Towards foveated rendering for gaze-tracked virtual reality. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 179.

Thies Pfeiffer and Patrick Renner. 2014. EyeSee3D: a low-cost approach for analysing mobile 3D eye tracking data using augmented reality technology. In *Proceedings of the Symposium on Eye Tracking Research and Applications*.

Eyal M Reingold and Heather Sheridan. 2011. Eye movements and visual expertise in chess and medicine. *Oxford handbook on eye movements* 528 (2011).

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

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).

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 and Applications*. ACM, 173–176.

Forest Tennant. 1988. The rapid eye test to detect drug abuse. *Postgraduate medicine* 84, 1 (1988), 108–114.

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*. ACM, 139–142.

FJ Vera-Olmos and Norberto Malpica. 2017. Deconvolutional neural network for pupil detection in real-world environments. In *International Work-Conference on the Interplay Between Natural and Artificial Computation*. Springer, 223–231.

FJ Vera-Olmos, Esteban Pardo, H Melero, and Norberto Malpica. 2019. DeepEye: Deep convolutional network for pupil detection in real environments. *Integrated Computer-Aided Engineering* 26, 1 (2019), 85–95.

Michel Wedel and Rik Pieters. 2008. A review of eye-tracking research in marketing. In *Review of marketing research*. Emerald Group Publishing Limited, 123–147.

Yuk-Hoi Yiu, Moustafa Aboulatta, Theresa Raiser, Leoni Ophey, Virginia L Flanagin, Peter zu Eulenburg, and Seyed-Ahmad Ahmadi. 2019. DeepVOG: Open-source Pupil Segmentation and Gaze Estimation in Neuroscience using Deep Learning. *Journal of neuroscience methods* (2019).