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

Neural networks for optical vector and eye ball parameter 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

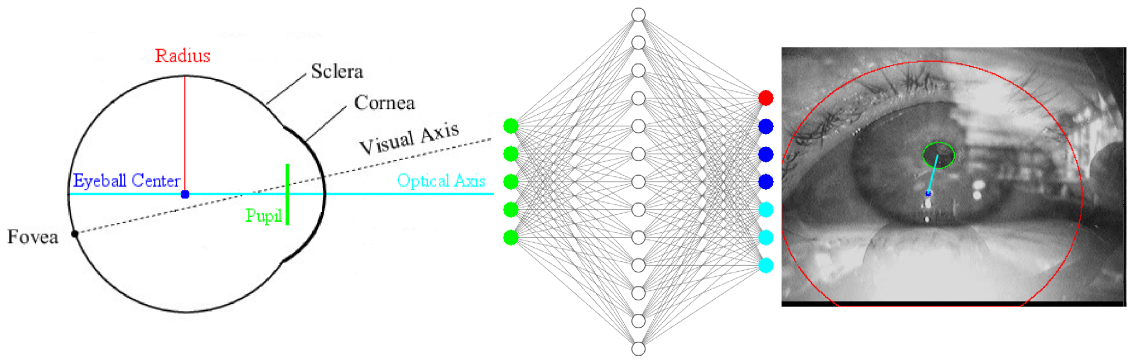

Figure 1: Estimation of the eyeball parameters (red and blue) and the optical axis (turquoise) using neuronal networks on the pupil ellipse (green).

ABSTRACT

In this work we evaluate neural networks, support vector machines and decision trees for the regression of the center of the eyeball and the optical vector based on the pupil ellipse. In the evaluation we analyze single ellipses as well as window-based approaches as input. Comparisons are made regarding accuracy and runtime. The evaluation gives an overview of the general expected accuracy with different models and amounts of input ellipses. A simulator was implemented for the generation of the training and evaluation data. For a visual evaluation and to push the state of the art in optical vector estimation, the best model was applied to real data. This real data came from public data sets in which the ellipse is already annotated by an algorithm. The optical vectors on real data and the generator are made publicly available. Link to the generator and models.

CCS CONCEPTS

• **Computing methodologies** → **Supervised learning**; **Machine learning approaches**; **Simulation tools**; • **Human-centered computing** → *Pointing*;

KEYWORDS

Gaze vector estimation, machine learning, eye tracking, runtime comparison, data set, pupil ellipse generator

ACM Reference Format:

Wolfgang Fuhl, Hong Gao, and Enkelejda Kasneci. 2020. Neural networks for optical vector and eye ball parameter estimation. In *Symposium on Eye Tracking Research and Applications (ETRA '20 Short Papers), June 2–5, 2020, Stuttgart, Germany*. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3379156.3391346

1 INTRODUCTION

3D gaze estimation is an essential part of eye tracking research. This 3D gaze information provides more information about the cognitive processes of a person [Bulling and Zander 2014] and allows to correct errors on a 2D plane via the additional information of depth [Wibirama et al. 2017]. The estimation of a person's 3D gaze position is a constantly growing application field and important in many research areas [Zhang et al. 2019]. The gaze position has a variety of applications like virtual reality where it is needed for foveated rendering [Patney et al. 2016] as well as input signal [Ktena et al. 2015]. Another field of application is gaze-based control as used in surgical microscopes [Eivazi et al. 2015; Fuhl et al. 2017b, 2016b], public displays [Zhang et al. 2014] or for controlling industrial robots [Roncone et al. 2016]. Also, the eye signal of a human being can be used for identification [Fookes et al. 2010] as well as for the teaching of novices [Castner et al. 2018]. This multitude of applications and research areas, however, requires a gaze signal that is as accurate and reliable as possible under the most diverse challenges and as minimally invasive as possible.

Figure 2 shows some of these image based challenges such as reflections, poor lighting conditions but also partially obscured

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

ACM ISBN 978-1-4503-7134-6/20/06…$15.00
https://doi.org/10.1145/3379156.3391346

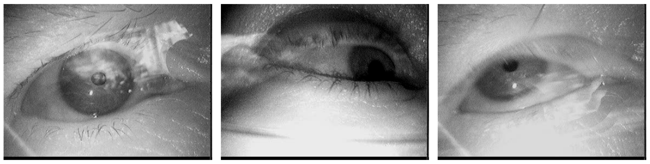

Figure 2: Different challenges for image-based extraction of the pupil ellipse.

pupils [Fuhl et al. 2015, 2017a]. Further challenges come from the use itself. If eye trackers are worn, there may be shifts which require compensation [Sugano and Bulling 2015]. In the case of remote eye tracking, the low resolution of the eye area is a challenge on the one hand [Klingner et al. 2008], and on the other hand to ensure an accurate gaze estimation without fixing the test person [Villanueva et al. 2009]. While the purely image-based problems as well as the low resolution of the eye area are not considered in this work, it deals with the determination of the optical axis (Figure 1) using the pupil ellipse since the optical axis is independent of an eye tracker shift or freely moving subjects thus making it robust against those challenges. In general, linear equation systems are solved to get from several pupil ellipses, which deform on the eyeball, to the parameters of the eyeball. This is possible because, depending on the viewing position and the stationary assumed camera, the pupil deforms according to the rotation on the eyeball. If one has several such deformations, the eyeball can be determined and the eye vector can be computed.

To get from the optical axis to the gaze vector only a personal calibration is necessary and has to be performed only once for a person [Kohlbecher et al. 2008]. The pure use of the pupil ellipse itself has the additional advantage that no further features like glints have to be projected onto the retina, which restrict the user in his movement [Villanueva et al. 2009]. Those two advantages make the pure use of the pupil ellipse highly interesting for research and industry. Since the unique calibration is also possible for the mass market and a reliable procedure can be used in remote as well as head mounted eye tracking even global corporations do research in this area.

This work deals with the determination of the eyeball parameters and the optical axis based on the pupil ellipse. Therefore, we wrote a simple pupil ellipse generator which ignores cornea refraction and used it for training data generation. Different neural network architectures are evaluated. In the evaluation window-based approaches, which process several pupil ellipses, as well as accuracy using a single ellipse are considered. This work can be seen as a kind of guideline for the selection of a neural network for the online determination of the optical axis using pupil ellipses. The trained models were also applied to real data from public data sets. The results on these more than 900.000 images together with a generator for pupil ellipses as well as eyeball parameters are made available to the public.

The summed contribution of this work are:

1. A generator for pupil ellipses based on a sphere (our eye ball approximation) and optical axis.
2. Evaluation of different neural network models for online application to determine the eyeball and the optical axis.
3. Evaluations for single pupil ellipses as well as window based approaches in which several pupil ellipses are processed in parallel.
4. Application to real data from public data sets.
5. Provision of the optical axis as well as the eyeball parameters for more than 900.000 images from public data sets.

2 RELATED WORK

In video based eye tracking there are a variety of approaches for gaze estimation. There exist mainly three categories. The first and still most commonly used technique for head mounted eye trackers is regression. This involves determining a function that projects the center of the pupil or other eye features onto the scene. Different methods have been used like least squares fitting of polynomials [Wang et al. 2005], Gaussian processes [Sesma-Sanchez et al. 2016] and also machine learning methods like support vector machine [Zhu et al. 2006] and neural networks [Jian-nan et al. 2009].

The second main category are the appearance based approaches. Here machine learning approaches are used to learn a direct mapping between eye images and scene. Since these approaches require a large amount of training data, a generator together with a Nearest Neighbor Regression has already been presented [Wood et al. 2016]. Also combinatorial methods were presented, which extract features in a first step and then determine the eye position with decision trees [Wang et al. 2016] or support vector machines [Xu et al. 2015]. More modern methods use convolutional neural networks which are able to learn the feature extraction as well as the mapping function [Fischer et al. 2018; Krafka et al. 2016; Zhang et al. 2015, 2017]. Since most approaches rely on pre-processing steps such as face and eye detection [Krafka et al. 2016; Zhang et al. 2015], a new data set as well as a model based on long short term memory cells has already been presented [Kellnhofer et al. 2019]. The main disadvantage of appearance based approaches are the high computing costs and the associated energy consumption.

The third category of gaze estimation approaches are model based. Here it is about estimating the eye ball and the optical axis [Hansen and Ji 2009]. Many approaches use near infrared or other light sources, which are projected onto the eye and produce reflections (glints) [Chen et al. 2008; Guestrin and Eizenman 2006; Hennessey et al. 2006; Shih and Liu 2004]. The pattern of the light sources is known in advance and the displacement of them on the eye ball is used for the parameter approximation. Together with the pupil center, the optical axis can be calculated. Afterwards, only a transformation of this vector is necessary to get the gaze vector (see Figure 1). An alternative to using glints uses the pupil ellipse and other eye features to compute the eye ball [Chen and Ji 2008; Hansen and Ji 2009]. Approaches based purely on the ellipses of the pupil [Li et al. 2018; Swirski and Dodgson 2013] or the iris [Li and Li 2016] are generally better suited, since, for example, the corners of the eye may not always be present in the image. However, approaches based on the ellipses have other disadvantages which result from incorrect extraction of the ellipses. Iterative methods with different cost functions were presented to compensate for this. An extension of the model-based approach is the consideration of the refraction on the cornea. In [Lai et al. 2014] two cameras were used to fit a model by the ellipses considering the refraction. In the case of glint based systems algorithms have already been presented for remote [Barsingerhorn et al. 2017; Villanueva et al. 2008] and head mounted [Dierkes et al. 2018] eye trackers. However, they are still based on iterative optimization. In [Dierkes et al. 2019] a direct method was introduced which works with only one camera.

Our work belongs to the model-based approaches, as we apply different machine learning algorithms to the pupil ellipses to calculate the eye model parameters. All methods used are applied directly to a single pupil ellipse or to several pupil ellipses in a time window and are therefore not iterative.

3 METHOD

The general approach of our method, can be seen in Figure 1. We use the parameters of the ellipse (green) to determine the eye ball radius (red), the eye ball center (dark blue), and the optical vector (turquoise). Since our method consists ellipse generation and the evaluation of different machine learning methods, this section was divided into two sub-sections. The first sub-section describes the generator and the second sub-section describes the training parameters used for the different machine learning methods.

3.1 Generator

Our generator works on the simple principle of an intersection between a cone (Pupil) and a sphere (Eye ball) with the sphere at the origin. This means we ignore the refraction and the cornea in our model. A grid is laid over the entire model and for each point in this grid it is checked whether it lies in the intersection surface or not. All points at the edge of the intersection surface (at least one neighbour which is not in the surface) are selected and a least squares ellipse fit is calculated. Afterwards, all points are shifted in relation to the sphere coordinates (away from the origin). This simple calculation method allows to calculate 5.000 ellipses on a single i5 CPU core per second in Matlab. In order to calculate different ellipses, iteration is performed over given ranges of values for the parameters sphere coordinates (X Y Z), aperture angle (pupil opening angle) and the two angles used to calculate the optical vector.

3.2 Machine Learning Algorithms and Training

For a simple reproduce ability we did not use any data preprocessing like mean subtraction and division by the standard deviation. The only change made to the input ellipse (Center c_x, center c_y, rotation phi, Axis a, and axis b) is that c_x and c_y are percentages of the image resolution in the respective direction and the axis a and axis b are divided by the pupil area.

In the following the parameters of all used machine learning algorithms are given to ensure reproduce ability. The training parameters for all neuronal network models using the Levenberg Marquart backpropagation [Hagan and Menhaj 1994] are maximum number of epochs 10.00, minimum of gradient 10^{-7}, initial scalar $mu = 0.001$, scalar reduction factor 0.1, scalar increase factor 10, and maximum scalar of 10^{10}. For the useage of Baysian Regularization [Foresee and Hagan 1997] only the initial scalar is changed to $mu = 0.005$ since it also uses the Leveberg Marquart Backpropagation internally. For the decision tree ensembles we changed only the number of tree depth 1, 5, and 10. As split criterion we used Gini's diversity index with a minimum observations per leaf of 5 and a minimum observations per parent of 10. For the creation of the ensemble we used bagging and least squares gradient boosting separately. For the support vector machine we used two different kernels, the linear and Gaussian. As parameters for training using the Gaussian kernel we set the box constraint (C) to one and Lambda (λ) to $1/(Cn)$ where n is the number of training samples.

4 EVALUATION

Table 1: Naming convention for the evaluation.

Name	Parameters
NN10LM	10 Neurons, Levenberg Marguart for training
NN20LM	20 Neurons, Levenberg Marguart for training
NN30LM	30 Neurons, Levenberg Marguart for training
NN40LM	40 Neurons, Levenberg Marguart for training
NN50LM	50 Neurons, Levenberg Marguart for training
NN10BR	10 Neurons, Baysian Regularization for training
NN20BR	20 Neurons, Baysian Regularization for training
NN30BR	30 Neurons, Baysian Regularization for training
NN40BR	40 Neurons, Baysian Regularization for training
NN50BR	50 Neurons, Baysian Regularization for training
T1LS	Decision tree, Least Squares gradient boosting
T5LS	5 decision trees, Least Squares gradient boosting
T10LS	10 decision trees, Least Squares gradient boosting
T1Bag	Decision tree, Bagging
T5Bag	5 decision trees, Bagging
T10Bag	10 decision trees, Bagging
SVMLIN	Support vector machine with Linear kernel
SVMGAU	Support vector machine with Gaussian kernel

Since our data for the evaluation as well as for the training comes from our simulator, we have decided on a 10% to 90% split. Here 10% is for the training and 90% for the evaluation. The training data was sampled uniformly over the entire data set which contains 9.000.000 samples. So every tenth sample was added to the training data. For the neural networks, we divided the 10% training data additionally into 5% training and 5% validation, whereby this division was carried out uniformly again. For the decision trees, the Support Vector Machines and the Gaussian Process Regression, the further division is not necessary. Table 1 describes all naming conventions for the evaluated algorithms. In the following, the evaluations are now discussed in terms of accuracy as well as runtime.

Table 2(T1), Table 2(T2), and Table 2(T3), show the errors made in 3D position, eyeball radius, and optical vector estimation. As can be seen, the neural networks outperform all other methods for a window size of 20 ellipses. In addition, the Baysian Regularization nearly always gives better results compared to the Levenberg Marquart backpropagation. The decision trees with least squares gradient boosting can be seen as the second best method. Up to a window size of 5 they even outperform the neural networks. If we also look at the runtimes in Table 2(T4), we can clearly see that the neural networks are much more efficient to compute than all other methods except the linear SVM. Since all runtimes refer to 1.000 input data and are specified in milliseconds, all methods can be considered real-time capable.

Table 2: Mean absolute error in pixel of the 3D eye ball position computed using the euclidean distance, mean absolute error in pixel of the eye ball radius, mean absolute error in degree between the 3D optical vector and the ground truth, and average runtime on one CPU core for 1.000 examples in milliseconds. W is the data window size (Amount of input ellipses). Bold values are the best result per widow size and sub table.

	T1:3D eye ball position (px)				T2:Eye ball radius (px)				T3:Mean optical vector error (°)				T4:Runtime for 1.000 ex. (ms)			
Method	W1	W5	W10	W20	W1	W5	W10	W20	W1	W5	W10	W20	W1	W5	W10	W20
NN10LM	10.84	8.72	6.44	5.69	10.84	8.72	6.44	5.69	13.11	10.56	4.18	3.40	5.00	2.56	1.03	1.18
NN20LM	8.80	5.69	4.62	3.10	8.80	5.69	4.62	3.10	12.76	6.70	2.81	2.24	5.00	2.35	0.73	0.60
NN30LM	7.78	4.52	3.18	2.35	7.78	4.52	3.18	2.35	12.57	6.00	2.58	1.82	5.00	2.26	0.50	0.47
NN40LM	7.57	4.09	2.25	1.99	7.57	4.09	2.25	1.99	12.47	5.90	2.03	1.53	5.00	2.16	0.45	0.38
NN50LM	7.30	3.84	**1.62**	1.39	7.30	3.84	**1.62**	1.39	12.33	5.62	**1.46**	1.21	5.00	2.16	0.39	0.36
NN10BR	10.73	7.75	7.19	5.65	10.73	7.75	7.19	5.65	13.11	12.51	4.02	3.46	5.00	2.48	1.15	1.40
NN20BR	8.81	5.77	3.41	3.37	8.81	5.77	3.41	3.37	12.73	7.32	3.19	2.40	5.00	2.27	0.55	0.62
NN30BR	7.89	4.19	3.28	2.06	7.89	4.19	3.28	2.06	12.52	5.93	2.53	1.70	5.00	2.17	0.48	0.33
NN40BR	7.41	3.44	2.22	1.69	7.41	3.44	2.22	1.69	12.49	5.28	1.93	1.32	5.00	2.16	0.33	0.37
NN50BR	**7.20**	3.10	1.97	**1.32**	**7.20**	3.10	1.97	**1.32**	12.33	4.84	1.73	**1.02**	5.00	2.14	**0.34**	**0.32**
T1LS	15.31	5.19	5.29	5.80	15.31	5.19	5.29	5.80	12.41	5.32	5.24	5.60	5.00	1.24	1.25	1.34
T5LS	8.17	3.63	2.89	3.37	8.17	3.63	2.89	3.37	11.37	3.90	2.81	3.16	5.00	0.68	0.69	0.74
T10LS	7.47	**3.03**	2.32	2.07	7.47	**3.03**	2.32	2.07	**10.72**	**3.05**	2.48	2.12	**4.99**	**0.61**	0.59	0.63
T1Bag	18.82	20.11	22.80	18.91	18.82	20.11	22.80	18.91	13.45	20.11	22.81	18.89	5.00	2.26	2.08	3.13
T5Bag	17.71	10.89	11.21	7.94	17.71	10.89	11.21	7.94	13.15	11.06	11.00	7.92	5.00	1.52	1.48	2.15
T10Bag	16.38	7.72	7.64	5.67	16.38	7.72	7.64	5.67	12.92	7.36	7.74	5.64	5.00	1.35	1.38	1.59
SVMLIN	15.72	5.03	5.13	5.17	15.72	5.03	5.13	5.17	13.67	4.88	5.02	5.08	5.02	1.03	1.03	1.03
SVMGAU	9.33	7.06	10.97	18.47	9.33	7.06	10.97	18.47	12.40	7.34	11.45	19.64	5.03	0.89	1.08	1.49

4.1 Real Data

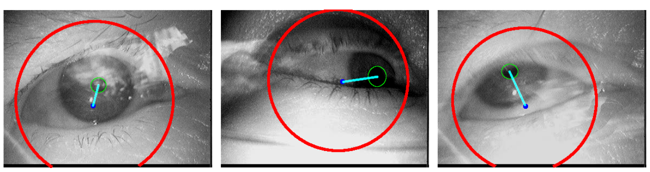

Figure 3: Exemplary results on real data of our best model.

For the evaluation on real data we used the semitic segmented data of [Fuhl et al. 2019b] and [Fuhl et al. 2019a], because the pupil ellipse can be extracted from the pupil segment. [Fuhl et al. 2019b] has more than 800.000 segmented images from the same studies [Kasneci et al. 2014] as used for the data sets in [Fuhl et al. 2015, 2017c, 2016a]. [Fuhl et al. 2019a] contains the segmentations for more than 100.000 images from the known data set [Tonsen et al. 2016]. In total, our approach has been applied to more than one million single images with each image having a size of 192×144. Figure 3 shows some exemplary results of our best model NN50W20. These results are made available to the public together with the generator code for training.

5 CONCLUSION

In this work we presented a generator for the fast generation (5.000 samples per second in Matlab on a i5 single core) of training data together with an evaluation of different machine learning methods. Among the evaluated procedures are Support Vector Machines with different kernel functions, decision trees, Boosted decision trees, and different neural network models. Each of these methods was evaluated with respect to accuracy and run-time. As an alternative to single pupil ellipses as input, a window-based approach with different sizes was evaluated. The best model was also applied to real data from a public source. The results on the public data as well as the code for the generator are made available to the public.

Future research in this area should explore the applicability of machine learning approaches to determine parameters for more complex models than a simple sphere as an eye ball. This would provide further improvements in the field of gaze vector determination and allow to provide even more accurate data sets.

This data and the generator should make it easier to train and evaluate own models in the future. It is freely accessible for both science and industry and shall further advance the progress in the field of eyeball parameter estimation as well as the calculation of the optical axis. The results of this work will serve as reference values and the models can be used as well. Link to the generator and models.

ACKNOWLEDGMENTS

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

REFERENCES

AD Barsingerhorn, FN Boonstra, and HHLM Goossens. 2017. Optics of the human cornea influence the accuracy of stereo eye-tracking methods: a simulation study. *Biomedical optics express* 8, 2 (2017), 712–725.

Andreas Bulling and Thorsten O Zander. 2014. Cognition-aware computing. *IEEE Pervasive Computing* 13, 3 (2014), 80–83.

Nora Castner, Enkelejda Kasneci, Thomas Kübler, Katharina Scheiter, Juliane Richter, Thérése Eder, Fabian Hüttig, and Constanze Keutel. 2018. Scanpath comparison in medical image reading skills of dental students: distinguishing stages of expertise development. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*. 1–9.

Jixu Chen and Qiang Ji. 2008. 3d gaze estimation with a single camera without ir illumination. In *2008 19th International Conference on Pattern Recognition*. IEEE, 1–4.

Jixu Chen, Yan Tong, Wayne Gray, and Qiang Ji. 2008. A robust 3D eye gaze tracking system using noise reduction. In *Proceedings of the 2008 symposium on Eye tracking* *research & applications*. ACM, 189–196.

Kai Dierkes, Moritz Kassner, and Andreas Bulling. 2018. A novel approach to single camera, glint-free 3D eye model fitting including corneal refraction. In *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*. ACM, 9.

Kai Dierkes, Moritz Kassner, and Andreas Bulling. 2019. A fast approach to refraction-aware 3D eye-model fitting and gaze prediction. In *Proc. International Symposium on Eye Tracking Research and Applications (ETRA)*. https://doi.org/10.1145/3314111.3319819

Shahram Eivazi, Roman Bednarik, Ville Leinonen, Mikael von und zu Fraunberg, and Juha E Jääskeläinen. 2015. Embedding an eye tracker into a surgical microscope: Requirements, design, and implementation. *IEEE Sensors Journal* 16, 7 (2015), 2070–2078.

Tobias Fischer, Hyung Jin Chang, and Yiannis Demiris. 2018. Rt-gene: Real-time eye gaze estimation in natural environments. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 334–352.

Clinton Fookes, Anthony Maeder, Sridha Sridharan, and George Mamic. 2010. Gaze based personal identification. In *Behavioral Biometrics for Human Identification: Intelligent Applications*. IGI Global, 237–263.

F Dan Foresee and Martin T Hagan. 1997. Gauss-Newton approximation to Bayesian learning. In *Proceedings of International Conference on Neural Networks (ICNN'97)*, Vol. 3. IEEE, 1930–1935.

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, 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, 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, and Enkelejda Kasneci. 2017b. Fast camera focus estimation for gaze-based focus control. *arXiv preprint arXiv:1711.03306* (2017).

Wolfgang Fuhl, Thiago Santini, Gjergji Kasneci, Wolfgang Rosenstiel, and Enkelejda Kasneci. 2017c. 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. 2016b. 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. 2016a. 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.

Elias Daniel Guestrin and Moshe Eizenman. 2006. General theory of remote gaze estimation using the pupil center and corneal reflections. *IEEE Transactions on biomedical engineering* 53, 6 (2006), 1124–1133.

Martin T Hagan and Mohammad B Menhaj. 1994. Training feedforward networks with the Marquardt algorithm. *IEEE transactions on Neural Networks* 5, 6 (1994), 989–993.

Dan Witzner Hansen and Qiang Ji. 2009. In the eye of the beholder: A survey of models for eyes and gaze. *IEEE transactions on pattern analysis and machine intelligence* 32, 3 (2009), 478–500.

Craig Hennessey, Borna Noureddin, and Peter Lawrence. 2006. A single camera eye-gaze tracking system with free head motion. In *Proceedings of the 2006 symposium on Eye tracking research & applications*. ACM, 87–94.

Chi Jian-nan, Zhang Chuang, Yan Yan-tao, Liu Yang, and Zhang Han. 2009. Eye gaze calculation based on nonlinear polynomial and generalized regression neural network. In *2009 Fifth International Conference on Natural Computation*, Vol. 3. IEEE, 617–623.

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), e87470.

Petr Kellnhofer, Adria Recasens, Simon Stent, Wojciech Matusik, and Antonio Torralba. 2019. Gaze360: Physically unconstrained gaze estimation in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*. 6912–6921.

Jeff Klingner, Rakshit Kumar, and Pat Hanrahan. 2008. Measuring the task-evoked pupillary response with a remote eye tracker. In *Proceedings of the 2008 symposium on Eye tracking research & applications*. ACM, 69–72.

Stefan Kohlbecher, Stanislavs Bardinst, Klaus Bartl, Erich Schneider, Tony Poitschke, and Markus Ablassmeier. 2008. Calibration-free eye tracking by reconstruction of the pupil ellipse in 3D space. In *Proceedings of the 2008 symposium on Eye tracking research & applications*. ACM, 135–138.

Kyle Krafka, Aditya Khosla, Petr Kellnhofer, Harini Kannan, Suchendra Bhandarkar, Wojciech Matusik, and Antonio Torralba. 2016. Eye tracking for everyone. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2176–2184.

Sofia Ira Ktena, William Abbott, and A Aldo Faisal. 2015. A virtual reality platform for safe evaluation and training of natural gaze-based wheelchair driving. In *2015 7th International IEEE/EMBS Conference on Neural Engineering (NER)*. IEEE, 236–239.

Chih-Chuan Lai, Sheng-Wen Shih, and Yi-Ping Hung. 2014. Hybrid method for 3-D gaze tracking using glint and contour features. *IEEE Transactions on Circuits and Systems for Video Technology* 25, 1 (2014), 24–37.

Jianfeng Li and Shigang Li. 2016. Two-phase approachâĂŤCalibration and iris contour estimationâĂŤFor gaze tracking of head-mounted eye camera. In *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 3136–3140.

Jianfeng Li, Shigang Li, Tong Chen, and Yiguang Liu. 2018. A Geometry-Appearance-Based Pupil Detection Method for Near-Infrared Head-Mounted Cameras. *IEEE Access* 6 (2018), 23242–23252.

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.

Alessandro Roncone, Ugo Pattacini, Giorgio Metta, and Lorenzo Natale. 2016. A Cartesian 6-DoF Gaze Controller for Humanoid Robots.. In *Robotics: science and systems*, Vol. 2016.

Laura Sesma-Sanchez, Yanxia Zhang, Andreas Bulling, and Hans Gellersen. 2016. Gaussian processes as an alternative to polynomial gaze estimation functions. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*. ACM, 229–232.

Sheng-Wen Shih and Jin Liu. 2004. A novel approach to 3-D gaze tracking using stereo cameras. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 34, 1 (2004), 234–245.

Yusuke Sugano and Andreas Bulling. 2015. Self-calibrating head-mounted eye trackers using egocentric visual saliency. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. ACM, 363–372.

Lech Swirski and Neil Dodgson. 2013. A fully-automatic, temporal approach to single camera, glint-free 3d eye model fitting. *Proc. PETMEI* (2013).

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.

Arantxa Villanueva, Rafael Cabeza, et al. 2008. Evaluation of corneal refraction in a model of a gaze tracking system. *IEEE Transactions on Biomedical Engineering* 55, 12 (2008), 2812–2822.

Arantxa Villanueva, Gintautas Daunys, Dan Witzner Hansen, Martin Böhme, Rafael Cabeza, André Meyer, and Erhardt Barth. 2009. A geometric approach to remote eye tracking. *Universal Access in the Information Society* 8, 4 (2009), 241.

Jian-Gang Wang, Eric Sung, and Ronda Venkateswarlu. 2005. Estimating the eye gaze from one eye. *Computer Vision and Image Understanding* 98, 1 (2005), 83–103.

Yafei Wang, Tianyi Shen, Guoliang Yuan, Jiming Bian, and Xianping Fu. 2016. Appearance-based gaze estimation using deep features and random forest regression. *Knowledge-Based Systems* 110 (2016), 293–301.

Sunu Wibirama, Hanung A Nugroho, and Kazuhiko Hamamoto. 2017. Evaluating 3D gaze tracking in virtual space: A computer graphics approach. *Entertainment computing* 21 (2017), 11–17.

Erroll Wood, Tadas Baltrušaitis, Louis-Philippe Morency, Peter Robinson, and Andreas Bulling. 2016. Learning an appearance-based gaze estimator from one million synthesised images. In *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research and Applications*. ACM, 131–138.

Pingmei Xu, Krista A Ehinger, Yinda Zhang, Adam Finkelstein, Sanjeev R Kulkarni, and Jianxiong Xiao. 2015. Turkergaze: Crowdsourcing saliency with webcam based eye tracking. *arXiv preprint arXiv:1504.06755* (2015).

Xucong Zhang, Yusuke Sugano, and Andreas Bulling. 2019. Evaluation of Appearance-Based Methods and Implications for Gaze-Based Applications. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, 416.

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.

Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. 2017. It's written all over your face: Full-face appearance-based gaze estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 51–60.

Yanxia Zhang, Jörg Müller, Ming Ki Chong, Andreas Bulling, and Hans Gellersen. 2014. GazeHorizon: enabling passers-by to interact with public displays by gaze. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 559–563.

Zhiwei Zhu, Qiang Ji, and Kristin P Bennett. 2006. Nonlinear eye gaze mapping function estimation via support vector regression. In *18th International Conference on Pattern Recognition (ICPR'06)*, Vol. 1. IEEE, 1132–1135.