Planning Report for MEng Individual Project

Deep Learning for Calibration-free Near-eye Gaze-Tracking

**Author name(s): Joseph Warren**

Supervisor(s): Dr Aldo Faisal, Dr Pavel Orlov

Submitted in partial fulfilment of the requirements for the award of BEng/MEng in Biomedical Engineering from Imperial College London

January 2019 Word count: 3953

# Project Specification

Significant progress has recently been made in remote gaze-tracking through the use of Deep Learning. Both remote and headset-based gaze-tracking suffer from similar variations in data: geometric translations of the eye due to head/headset movement, variable illumination, and variation in user eye appearance. Conventional headset-based methods cope with these each time of use through a calibration process(1-6). Remote gaze-tracking techniques however must deal with head-movement, and as such calibration-free techniques capable of coping with this have been developed(7-14).

An inherent problem exists in the calibration process employed by current headset methods: headset movement relative to the head is not tolerated. Headset movement is inevitable in the real world due to adjustment, slippage and change of user. This is not practical beyond the lab or desktop setting as a complex calibration set-up is required to rectify this. Furthermore, any calibration process relies on the compliance of the user. This may not be possible if the user is a child(15) , an adult with learning disabilities or impaired language comprehension, or patients with a brain injury. As a result, the calibration step restricts the scalability of the technology to the outside world. Ultimately, the ideal technology would be a headset that is highly accurate, requires no calibration and is person-independent.

This project aims to investigate how methods could be adopted from Deep Learning remote gaze-tracking and applied to headset-based gaze-tracking to achieve both invariance to headset movement and generalisation to new users. There are several reasons for optimism in this area. 1) The impressive results achieved by the aforementioned remote methods (table 1). 2) The high quality data obtained through the use of specialised near-eye equipment (SMI) in comparison to the webcams frequently used in remote methods. 3) The extremely limited range of headset movement in comparison to the free head movement dealt with by remote methods. 4) The diverse range of ideas and methods for data collection, smart augmentation and processing that have not yet been implemented in conjunction. 5) The fact that calibration-free remote methods focus entirely on locating gaze-position in the 2D domain and are untested in 3D. 6) The dearth of scientific literature on Deep Learning methods in headset-based gaze-tracking.

Since this topic is relatively unexplored, a range of objectives of increasing difficulty have been identified. The aim of this project will be to progress as far as possible through them sequentially. The objectives are listed below and expanded upon in section 4.

In the 2D domain:

1. Verify accuracy and feasibility of Deep Learning methods to predict gaze-position from near eye data, in absence of headset movement or user generalisation
2. Investigate ability to generalise to new users, in absence of headset movement
3. Investigate ability to cope with headset movement, in absence of user generalisation

In the 3D domain:

1. Investigate ability to cope with headset movement, in absence of user generalisation
2. Provide basis for future work on system capable of coping with headset movement *and* generalisation to new users

In implementation:

Efficiently implement the algorithm so that gaze-position predictions are made in real-time. Running at 30fps is a minimum requirement, however 60-120fps would be preferred.

Note that objectives 3-5 will require the collection of new data and as such are far more time consuming.

# Ethical Analysis

As discussed later in section 3.1, eye movement is strongly correlated with intention. This data has immense power for good. For example, gaze-tracking provides an outlet for those incapable of physical interaction with the world to regain this ability via robotics and prosthetics. However, this data is also extremely sensitive and can be misused. After the recent data fallout involving companies such as Google, Facebook and Cambridge Analytica, it is clear that data revealing behaviours, vices and opinions is extremely valuable. Indeed, companies already harvest data on smartphone ‘swiping’ frequency to deduce attention and target ads. This is evidently a moral grey area, and gaze-tracking would align with this vision. Worryingly, it has even been shown that manipulation of gaze-direction can bias intention and moral decisions(16). For these reasons, data collected from users should be kept anonymous, both in this study and beyond.

# Literature Review

The primary aim for any gaze-tracking system is to map input information of the eye to a gaze-position, or ‘Point-of-Regard’ (PoR). The form of input varies between methods: some require hand-crafted features and shapes such as pupil centre position or pupil contour, whereas others operate on raw pixel data from the eye image. The former is commonly referred to as ‘Shape-based’, while the latter is known as ‘Appearance-based’(8). Shape-based approaches invariably require a calibration step in order to map from the features to a PoR, whereas appearance-based methods frequently do not(17) . Each approach will be discussed in further detail in sections 3.2 & 3.3.

## Motivations for 3D Gaze-Tracking

Eyes are perhaps the most important sensory data stream to the brain and play a key role in how we interact with the world. The fovea is the most sensitive region of the retina, and so eye movement is essential in order to focus regions of interest, such as objects we would like to grasp, on this point(5). For this reason, eye movement is strongly correlated with intention. For patients that suffer from serious motor disabilities, eye movement may be the primary method of interaction with the world. This is because the oculo-motor system frequently remains functional after serious injury such as spinal trauma, paralysis and stroke, and degenerative diseases such as Parkinson’s and Multiple Sclerosis(18). In these cases, gaze-tracking provides an outlet to physically interact with the world, by controlling technologies such as motorised wheelchairs and robotic arms.

## Overview of Calibration-based Methods

1. Performance in Headset-based Gaze-Tracking

Accuracies achieved by calibration-based methods are relatively high, often reaching 0.5°. A Review and Analysis of Eye-Gaze Estimation, Algorithms and Performance Evaluation Methods in Consumer Platforms(19) is an excellent review of the current landscape, and concisely summarises the accuracies and methods from a number of studies on page 9 and 13.

1. Calibration Process

The calibration process requires the user to look at a set of points, which are then used to learn a mapping. Importantly, more calibration points used results in improved accuracy, since less interpolation is required. This number often varies from 3-13, while there must be some points spread along the edges of the display(19). Training is performed using a variety of algorithms, the most common being 2D regression. In this method, the required input is a vector connecting the pupil centre (PC) and corneal reflections (CR), which reveals information about the three-dimensional form of the eye and the optical axis. To detect PC and CR, image processing techniques are required to first isolate the pupil and then locate its centre. Then, a regression algorithm, frequently polynomial regression or SVR, learns a mapping from the vector to a 2D gaze position. The number of terms used in the polynomial regression technique is often debated: for a large number of a calibration points a high order polynomial maybe used. This may however result in overfitting. Katarzyna et al.(5) use the second polynomial seen in equations 1 & 2.

Equation 1

Equation 2

Where (X, Y) are gaze-estimates in 2D space and (x, y) are the coordinates of the PCCR vector. Parameters **a, b** are found using least squares regression. This study found that polynomial regression is more sensitive to poor examples than SVR and Artificial Neural Networks (ANN), however less sensitive to the distribution of calibration points in 2D space. Another study achieved an accuracy as high as 0.39° using an ANN with 16 calibration points(2) .

Accurate pupil detection is of course an extremely important step in the methods mentioned previously. Any error at this step will be amplified during regression. Santini, Fuhl and Kasneci propose an edge-based state-of-the-art pupil detection algorithm called PuRe(20). Down-sampling and normalisation are followed by Canny edge-detection and Morphological operations to find edges. These edges are then condensed into D key-points using a cosine chain and identified as belonging to the pupil using a number of heuristics. An ellipse is then fit to the edge segment using a least-squares method, and again kept or discarded based on different heuristics. Pupil centre location was located to within 5 pixels of the true value 72.02% of the time. Run-time was approximately 5ms. Occlusions are also a key obstacle to shape-based approaches. Ma et al. propose a novel technique for coping with missing CR via geometric transforms(4).

1. 3D Gaze-Tracking Methods

There are two frequently used methods for 3D gaze-estimation: vergence and RGB-D. The first involves constructing 3D gaze vectors of each eye and computing the intersection of these(6,18). However, this assumes that gaze vectors do intersect, and accuracy in the z-direction degrades over larger distances. Alternative methods involve the use of a head-mounted RGB-D camera. The camera provides depth information of the field of view. A 2D gaze-estimate is mapped to a pixel in the camera image, and a depth deduced. However, this is not robust for edges or surfaces with large gradient in the z-direction.

## Overview of Calibration-free Methods

Calibration-free methods have recently gained ground through the use of Neural Networks and Deep Learning. In 1994 Baluja and Pomerleau first proposed the use of an ANN with a single hidden layer to deduce a mapping directly from pixels to PoR, achieving an accuracy of 1.5deg while tolerating free head-movement(12). Since then, and the incredible success of algorithms such as AlexNet(21) in the domain of image recognition, recent efforts have focused on the use of Convolutional Neural Networks (CNN) to deduce a mapping(7-11,22). SVR has also been shown to cope with head-movement(23).

1. Performance in Remote Gaze-Tracking

CNNs and 3-layer feedforward ANNs have shown impressive performance on highly variable eye data (table 2). Indeed, studies that ignored user-generalisation and trained/tested on the same subjects achieved accuracies as high as 1.5° despite free head-movement(12,13). Studies that also tested for user generalisation were able to achieve error of approximately 4-8° (7-10). Processing speed is in the range of 50ms per frame, with the notable exception of (7) which achieves a speed of 3-15ms.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Citation** | **Error (deg)** | **Speed (ms)** | **Free head-movement** | **User Generalisation** | **Dataset used** |
| (7) | 3.64 | 3-15 | Yes | Yes | MPIIGaze |
| (8) | NA\*\*\* | 50 | Yes | Yes | GazeCapture (Own), TabletGaze |
| (9) | 5.9 (UT Multiview)  6.3 (MPIIGaze) | - | Yes | Yes | MPIIGaze, UTMultiview |
| (10) | 5-8 | 38 | Yes | Yes | Own |
| (11) | 6.7 | - | No | No | Own, CAVE |
| (12) | 1.5 | 67 | Yes | No | Own |
| (13) | 1.5 | 50 | Yes | No\* | Own (User-specific pre-training) |
| (14) | 3.7 | - | No | No\*\* | Own (User-specific calibration step) |
| Table 3 – Summary of studies utilising Neural Networks for remote gaze estimation | | | | | |

\*Requires user-specific training process.

\*\*Requires 60s calibration.

\*\*\*Error in cm

1. Architectures and Implementation

ANN architectures used are composed of an input layer, connected to image pixels of a single eye, a hidden layer, and an output layer corresponding to gaze position. In (12,13) a divided hidden layer is utilised as shown in fig. 2a. The two halves were trained to recognise PoR in the x and y directions respectively. This assumes that features relevant to predict the x-coordinate are not required for predicting the y-coordinate(12). The output layer localises PoR onto a 50x40 grid in 2D space. Gaussian shaped coding, as opposed to 1-out-of-N, was used to represent the target gaze-positions. While in 1-out-of-N a single output is set to 1, Gaussian shaped coding sets soft target values as given by equation 3. This is appropriate as gaze-estimation requires a mapping function, as opposed to strict classification(13). The hyperbolic tangent function was used for activation. The cost function used in (13) was summed squared error, while deviation of the predicted and actual PoR was used as a stopping criterion. Both (12,13) required the use of an offset table to correct PoRs. See (13) for in-depth description of training procedure and parameters. In contrast, study (14) utilised an ANN with two output units, corresponding to x and y coordinates. This again assumed linear separability. The Fast Artificial Neural Network (FANN) library was used to implement this. Histogram normalisation is frequently used to enhance eye features(13,14) fig. 2b.

Equation 3

Where n0 is the true gaze coordinate in either x or y and n the output unit.

|  |  |
| --- | --- |
| (a) | (b) |
| Figure 2: (a) ANN with divided hidden layer as used in (13)*.* (b) Eye image before (upper) and after (lower) normalisation in (13). Note increase in contrast and enhancement of features. | |

CNNs follow a similar input-output procedure, with the additional input requirement of head angle. Successive convolutional layers are followed by pooling, ReLU and fully connected layers, which eventually perform regression to a 2D PoR estimate or classification onto a predefined grid. Studies (9,11) utilise the LeNet CNN architecture, as shown in fig. 3d & e respectively. (7) makes significant efficiency improvements, achieving a processing time of 3-15ms, by adapting the architecture to reduce the number of parameters (fig. 3b). (8) applies the ideas of Hinton et al.(24) to reduce model complexity, memory requirements and processing time. This involves distilling the knowledge acquired from a number of ensemble models into a single smaller model. (10) uses a CNN to obtain ‘deep features’, which are then used to estimate gaze position using Random Forest regression.

Data augmentation is essential to increase the number of training samples, reducing the risk of overfitting. In gaze-tracking this frequently involves geometric transformations or adding noise. Down-sampling of training images in (7) is done to simulate the effect of different distances from the camera. Crucially, a different resampling formula was used on the training and test data to ensure the specific interpolation method was not learnt. Furthermore, dropout is performed to reduce risk of overfitting. Another study(8) translates the eyes, resulting in a 25-fold increase in training data. Interestingly, augmentation was also performed on test data, allowing for the resulting predictions to be averaged to yield a more accurate single prediction. Augmentation of both the test and training sets was found to improve prediction accuracy. In (11) data is augmented by adding a variety of noise types to the images, while in (22) rotations, blurring and scaling were performed. Interestingly, the binning of data to a lower precision may improve learning(11).

Indeed, in all studies data segmentation was found to be extremely beneficial. However, a more powerful technique for augmentation is proposed by Lemley, Bazrafkan and Corcoran(25). This involves the use of a Generative Adversarial Network (GAN) to generate entirely new images that are a mixture of training images with the same label, and thus has the potential to significantly expand a dataset. This technique may be used to significantly reduce the size of the required network, with one study finding a 207-fold decrease in the required computational resources(26).

Use of the GPU over CPU results in a significant decrease in processing time, allowing for real-time operation. One study found a 29-fold decrease when implementing the CNN on a GPU(10). Another utilised the GPU to achieve processing times as low as 3ms (7). Caffe (Convolutional Architecture for Fast Feature Embedding), a library for fast and efficient implementation of CNNs in Python/C++ (27), was used in the study.

|  |  |
| --- | --- |
|  |  |
| (a) | (b) |
|  | |
| (c) | |
|  | |
| (d) | |
|  | |
| (e) | |
| Figure 3: (a) CNN used by Krafka et al.(8). CONV and FC refer to convolutional and fully connected layers respectively. (b) Efficient CNN utilised by Lemley et al(7). (c) Architecture for 7-way gaze-classification(22). (d) Adapted LeNet architecture for 2D gaze-estimation(9). (e) Standard LeNet CNN architecture(11). | |

1. Datasets

Models must be pre-trained to learn a mapping function, requiring a large and varied training dataset. The dataset contains images of eyes corresponding to known PoRs. Crucially, to achieve invariance to head-movement and illumination, and generalisation to new users, the dataset must contain a diverse range of subjects, lighting conditions and head angles. Indeed, (8) found that the number of training subjects significantly impacts generalisation ability.

Datasets used in previous studies are summarised in table 2. Notably, studies (11-13) were not interested in user generalisation and as such include only 1-5 subjects. Of the datasets tailored towards user generalisation, the number of subjects varies between 15 and 1474, and images per subject varies between approximately 1000 and 20000 for continuous PoR. The CAVE dataset (28) is for training an algorithm to classify discrete gaze positions and therefore requires far less data per subject (105 images).

Many of the datasets in previous studies only have data for a narrow range of viewing angles – below roughly 25° (fig.4a & b). This is because data collection is often performed using points displayed on a small screen. This is not an issue if the primary aim is to localise PoR on a 2D screen, however for applications in 3D this narrow range does not suffice. Eyediap (29) is the only existing database that contains 3D PoR data and a horizontal viewing angles of upto 45°.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Dataset** | **Number of Images** | **Number of Subjects** | **Average Images per Subject** | **Gaze Position** | **Free Head-Movement** | **Lighting Conditions** | **3D PoR** |
| MPIIGaze(9) | ~200,000 | 15 | ~13000 | cont. | cont. | NL | No |
| GazeCapture(8) | ~2,500,000 | 1474 | ~1700 | cont. | cont. | NL | No |
| UT Multiview(30) | 64,000 | 50 | 1280 | disc. | disc. | NL | No |
| TabletGaze Subset(31) | ~100,000 | 41 | 2439 | disc. | cont. | NL | No |
| CAVE(28) | 5,880 | 56 | 105 | disc. | disc. | NL | No |
| Eyediap(29) | ~360,000\* | 16 | 22500 | cont. & disc. | cont. & disc. | NL | Yes |
| (10) | ~100,000 | 6 | ~17000 | disc. | cont. & disc. | NL | No |
| (11) | ~36,000\*\* | 5 | ~7200 | cont. | none | NL | No |
| (12) | 8000 | 1 | 8000 | cont. | disc. | NL | No |
| (13) | ~4000 | 1 | ~4000 | disc. | assumed none | NL | No |
| Table 2 – Summary of datasets used for training in remote gaze estimation studies | | | | | | | |

\*Based on lower bound of 4hrs footage recorded at 25fps.

\*\*Based on 5x 4mins of footage recorded at approximately 30fps.

|  |  |
| --- | --- |
| (a) | (b) |
| Figure 4: (a) Head angle (**h)** and PoR (**g)** for the TabletGaze, MPIIGaze and GazeCapture datasets(8). (b) **h** and **g** for the UT Multiview and Eyediap datasets(9). Note that the axis scale is -1.0 to 1.0, corresponding to viewing angles of ±90°. Red denotes high frequency of images whereas blue denotes low. | |

1. Data Collection

Standard methods for collecting data involve asking the user to fixate on a selection of known points for a prolonged period of time. However, this procedure only yields data for relatively few PoRs, meaning that far more data must be collected in order to have an even distribution over visual space. This is important to reduce risk of overfitting.

Studies (6,11,12) propose more elegant solutions by utilising our ability to track a target with smooth pursuit. Tostado, Abbott and Faisal(6) utilise the trajectory of a robotic arm to generate a rich set of unique 3D PoR calibration points. Likewise, (11,12) utilise a cursor sweeping across a screen to populate the training set with a complete set of finely incremented 2D PoR training points. These methods provide a far more efficient way of acquiring data, with (11) reporting that when recording at 30fps and moving the cursor at 7.5°/s, the maximal speed for smooth pursuit, over 7000 data points corresponding to unique PoRs were collected in a 4-minute recording session.

In order to obtain the ground truth PoR, head position must be measured relative to the targets. This is currently done in the lab via motion tracking.

# Implementation Plan

As mentioned in section 1 there are five key milestones in the project. These are summarised below and in table 3.

## 4.1 Testing in the 2D domain

1. *Verify CNN ability to accurately predict 2D PoR in absence of headset movement and user generalisation, using existing lab database*

The lab currently has an existing database of near-eye images corresponding to known 2D PoRs, taken from different subjects. For each subject, images will be split into training and testing. CNN accuracy will be tested on each subject, while implementing different architectures, data augmentation techniques and down-sampled image resolutions. Performance will be compared to the standard polynomial regression calibration technique and the Artificial Neural Network implementation proposed in (12).

Approximate duration: 4 weeks

Approximate timeline: 07/01/18 – 04/02/18

1. *Investigate ability to generalise to new users in absence of headset movement, using existing lab database*

The same experiment will be performed as above, however the training set will consist of a selection of subjects while the test set will contain the remaining ‘unseen’ subjects.

Approximate duration: 4 weeks

Approximate timeline: 04/02/18 – 04/03/18

1. *Investigate invariance to headset movement in absence of user generalisation, using own 2D PoR data*

The dataset mentioned previously was collected using a different system to the SMI hardware currently used in the lab. Therefore, to test the method on SMI, new 2D PoR data will be collected from a small number of subjects using the current headset. Furthermore, the previous dataset did not account for different headset positions on the same user and therefore cannot be used to test for invariance.

Data will be collected using the smooth pursuit method described in (11) to acquire an even distribution of continuous gaze-positions, thereby minimising the risk of overfitting. In each test, the subject will repeatedly be asked to perturb the headset (e.g. by removal or adjustment) before following a cursor as it sweeps across the screen. This will be repeated multiple times to ensure a wide range of headset positions. Assuming, as in (11), around 10000 images are required for a CNN to localise gaze in 2D for a single user and headpose, and at least 4 different head poses are required for invariance as in (12), then approximately 40,000 images are required. This corresponds to roughly 7 minutes of video at 100fps.

Approximate duration: 8 weeks

Approximate timeline: 04/02/18 – 15/04/18

## 4.2 Expanding to the 3D domain

1. *Investigate invariance to headset-movement when predicting 3D PoR in absence of user generalisation, using own 3D PoR data*

Data will be collected as above, however each subject will be asked to follow a target in 3D visual space. This may be performed as in (6), using a robotic arm end-point as a visual target. It is important to note that performing gaze-tracking in 3D requires far more data than the 2D case. For example, if one wishes to localise depth in the range of 1m to a precision of 1cm, a 100-fold increase in data is expected.

Approximate duration: 8+ weeks

Approximate timeline: 15/04/18 – 10/06/18

1. *Provide basis for future work on system capable of coping with headset movement and generalisation to new users*

Work must be scalable to larger future projects.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Milestone | Headset Movement | User Generalisation | 3D | Dataset required | Approximate required dataset size | Approximate footage time |
| 1 | No | No | No | Existing lab | - | - |
| 2 | No | Yes | No | Existing lab | - | - |
| 3 | Yes | No | No | Own | ~40000 | ~10min |
| 4 | Yes | No | Yes | Own | ~106 | ~3hrs |
| *5\** | *Yes* | *Yes* | *Yes* | *Own* | *~107* | *~45hrs* |
| Table 3 – Summary of project objectives | | | | | | |

\*This is unattainable in the scope of this project, however the groundwork could be set for such a dataset/experiment

# Evaluation

Final accuracies achieved will be found using the test sets described above. These will be compared to those in scientific literature for both remote CNN-based methods and calibration-based near-eye methods. Furthermore, artificial eyes may be used to assess the accuracy of the proposed methods against the ground truth, in the absence of eye-movements. Processing speed will also be assessed and compared, and the maximal fps calculated. For experimental set-ups refer to section 4.

# Preliminary Results

Pupil detection techniques have been experimented with on Python. OpenCV was used for morphological operations, while Otsu thresholding was used for segmentation. Connected components has been implemented to isolate the pupil.

I have begun educating myself on SMI data acquisition open-source C++ software in preparation for data collection. The next key task is aligning image timestamps with cursor/robotic arm position timestamps.

# References

(1) E.D. Guestrin, M. Eizenman. General theory of remote gaze estimation using the pupil center and corneal reflections. *IEEE Transactions on Biomedical Engineering.* 2006; 53 (6): 1124. Available from: doi: 10.1109/TBME.2005.863952 .

(2) Wang J, Zhang G, Shi J. 2D Gaze Estimation Based on Pupil Glint Vector. *Applied Sciences.* 2016; 6 (6): 174.

(3) Chi Jian-nan, Zhang Chuang, Yan Yan-tao, Liu Yang and Zhang Han. Eye Gaze Calculation Based on Nonlinear Polynomial and Generalized Regression Neural Network. *2009 Fifth International Conference on Natural Computation* : IEEE; Aug 2009. pp. 617-623. Available from: <https://ieeexplore.ieee.org/document/5363190>. Available from: 10.1109/ICNC.2009.599.

(4) Ma C, Choi K, Choi B and Ko S. Robust remote gaze estimation method based on multiple geometric transforms. *Optical Engineering.* 2015; 54 (8): 083103. Available from: doi: 10.1117/1.OE.54.8.083103 Available from: <http://www.dx.doi.org/10.1117/1.OE.54.8.083103> .

(5) Harezlak K, Kasprowski P and Stasch M. Towards Accurate Eye Tracker Calibration – Methods and Procedures. *Procedia Computer Science.* 2014; 35 1073-1081. Available from: doi: 10.1016/j.procs.2014.08.194 Available from: <https://www.sciencedirect.com/science/article/pii/S1877050914011594> .

(6) Tostado PM, Abbott WW and Faisal AA. 3D gaze cursor: Continuous calibration and end-point grasp control of robotic actuators. *2016 IEEE International Conference on Robotics and Automation (ICRA)* : IEEE; May 2016. pp. 3295-3300. Available from: <https://ieeexplore.ieee.org/document/7487502>. Available from: 10.1109/ICRA.2016.7487502.

(7) Lemley J, Kar A, Drimbarean A and Corcoran P. Efficient CNN Implementation for Eye-Gaze Estimation on Low-Power/Low-Quality Consumer Imaging Systems*.* 2018. Available from: <https://www.openaire.eu/search/publication?articleId=od________18::a13013d48ace35309f87cbac6f7af951> .

(8) Krafka K, Khosla A, Kellnhofer P, Kannan H, Bhandarkar S, Matusik W, Torralba A*.* Eye Tracking for Everyone. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR*) 2016; 2176-2184. Available from: 10.1109/CVPR.2016.239.

(9) Xucong Zhang, Sugano Y, Fritz M and Bulling A. Appearance-based gaze estimation in the wild. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* : IEEE; Jun 2015. pp. 4511-4520. Available from: <https://ieeexplore.ieee.org/document/7299081>. Available from: 10.1109/CVPR.2015.7299081.

(10) Wang Y, Shen T, Yuan B, Bian J, Fu X. Appearance-based Gaze Estimation using Deep Features and Random Forest Regression. *Knowledge-Based Systems.* 2016; Available from: doi:110. 10.1016/j.knosys.2016.07.038.

(11) Konrad R, Shrestha S, Varma P. Near-Eye Display Gaze Tracking via Convolutional Neural Networks. 2016;

(12) Baluja S, Pomerleau D. *Non-Intrusive Gaze Tracking Using Artificial Neural Networks.* 1994. Available from: <http://www.dtic.mil/docs/citations/ADA275186>.

(13) Xu L, Machin D, Sheppard P. *A Novel Approach to Real-time Non-intrusive Gaze Finding.* 1998. Available from: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.16.665

(14) Sewell W, Komogortsev O. Real-time eye gaze tracking with an unmodified commodity webcam employing a neural network. *CHI '10 Extended Abstracts on human factors in computing systems* : ACM; Apr 10, 2010. pp. 3739-3744. Available from: <http://dl.acm.org/citation.cfm?id=1754048>. Available from: 10.1145/1753846.1754048.

(15) Noris B, Keller J and Billard A. A wearable gaze tracking system for children in unconstrained environments. *Computer Vision and Image Understanding.* 2011; 115 (4): 476-486. Available from: doi: 10.1016/j.cviu.2010.11.013 Available from: <https://www.sciencedirect.com/science/article/pii/S1077314210002493> .

(16) Philip Pärnamets, Petter Johansson, Lars Hall, Christian Balkenius, Michael J. Spivey and Daniel C. Richardson. Biasing moral decisions by exploiting the dynamics of eye gaze. *Proceedings of the National Academy of Sciences of the United States of America.* 2015; 112 (13): 4170-4175. Available from: doi: 10.1073/pnas.1415250112 Available from: <https://www.jstor.org/stable/26462427> .

(17) Hansen, Dan & Ji, Qiang. (2010). In the Eye of the Beholder: A Survey of Models for Eyes and Gaze. *IEEE transactions on pattern analysis and machine intelligence.* 2010; 32. 478-500. Avaliable from: doi: 10.1109/TPAMI.2009.30.

(18) Abbott WW, Faisal AA. Ultra-low-cost 3D gaze estimation: an intuitive high information throughput compliment to direct brain-machine interfaces. *Journal of neural engineering.* 2012; 9 (4): 046016. Available from: doi: 10.1088/1741-2560/9/4/046016 Available from: <https://www.ncbi.nlm.nih.gov/pubmed/22791699> .

(19) Kar A, Corcoran P. A Review and Analysis of Eye-Gaze Estimation Systems, Algorithms and Performance Evaluation Methods in Consumer Platforms. *IEEE Access.* 2017; 5 16495-16519. Available from: doi: 10.1109/ACCESS.2017.2735633 Available from: <https://ieeexplore.ieee.org/document/8003267> .

(20) Santini T, Fuhl W and Kasneci E. PuRe: Robust pupil detection for real-time pervasive eye tracking. *Computer Vision and Image Understanding.* 2018; 170 40-50. Available from: 10.1016/j.cviu.2018.02.002 Available from: <https://www.sciencedirect.com/science/article/pii/S1077314218300146> .

(21) Krizhevsky A, Sutskever I and Hinton G. *ImageNet classification with deep convolutional neural networks.* New York: ACM; 2017. Available from: <http://dl.acm.org/citation.cfm?id=3065386> .

(22) George A, Routray A. Real-time eye gaze direction classification using convolutional neural network. *2016 International Conference on Signal Processing and Communications (SPCOM)* : IEEE; Jun 2016. pp. 1-5. Available from: <https://ieeexplore.ieee.org/document/7746701>. Available from: 10.1109/SPCOM.2016.7746701.

(23) Zhiwei Zhu, Qiang Ji and Bennett KP. Nonlinear Eye Gaze Mapping Function Estimation via Support Vector Regression. *18th International Conference on Pattern Recognition (ICPR'06)* : IEEE; 2006. pp. 1132-1135. Available from: <https://ieeexplore.ieee.org/document/1699089>. Available from: 10.1109/ICPR.2006.864.

(24) Hinton G, Vinyals O and Dean J. Distilling the Knowledge in a Neural Network*.* 2015. Available from: <https://www.openaire.eu/search/publication?articleId=od________18::35301935752bea561452be93076b5116> .

(25) Lemley J, Bazrafkan S and Corcoran P. Smart Augmentation Learning an Optimal Data Augmentation Strategy. *IEEE Access.* 2017; 5 5858-5869. Available from: doi: 10.1109/ACCESS.2017.2696121 Available from: <https://ieeexplore.ieee.org/document/7906545> .

(26) Lemley J, Bazrafkan S, Corcoran P. Learning data Augmentation for Consumer Devices and Services.[*2018 IEEE International Conference on Consumer Electronics (ICCE)*](https://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=8322492)*.* 2018; Available from: <https://ieeexplore.ieee.org/document/8326321>. Available from: [10.1109/ICCE.2018.8326321](https://doi.org/10.1109/ICCE.2018.8326321)

(27) Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R.B., Guadarrama, S., & Darrell, T. Caffe: Convolutional Architecture for Fast Feature Embedding. 2014; ACM Multimedia.*.*

(28) *Smith B, Yin Q, Feiner SK, Nayar SK.* Gaze locking: Passive eye contact detection for human-object interaction. *UIST 2013 - Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology.* 2013; *271-280.* Available from:10.1145/2501988.2501994.

(29) Funes Mora KA, Monay F and Odobez J. EYEDIAP. *Proceedings of the Symposium on eye tracking research and applications* : ACM; Mar 26, 2014. pp. 255-258. Available from: <http://dl.acm.org/citation.cfm?id=2578190>. Available from: 10.1145/2578153.2578190.

(30) Sugano Y, Matsushita Y and Sato Y. Learning-by-Synthesis for Appearance-Based 3D Gaze Estimation. *2014 IEEE Conference on Computer Vision and Pattern Recognition* : IEEE; Jun 2014. pp. 1821-1828. Available from: <https://ieeexplore.ieee.org/document/6909631>. Available from: 10.1109/CVPR.2014.235.

(31) Huang Q. *Tablet Gaze: Dataset and Algorithm for Unconstrained Appearance-based Gaze Estimation in Mobile Tablets.* ProQuest Dissertations Publishing; 2015.