Learning A 3D Gaze Estimator with Improved Itracker Combined with Bidirectional LSTM

Conference Paper · July 2019

DOI: 10.1109/ICME.2019.00151

Xiaolong Zhou
Zhejiang University of Technology
75 PUBLICATIONS **419** CITATIONS
SEE PROFILE

Jiang Jiaqi
Zhejiang University of Technology
3 PUBLICATIONS **7** CITATIONS
SEE PROFILE

LEARNING A 3D GAZE ESTIMATOR WITH IMPROVED ITRACKER COMBINED WITH BIDIRECTIONAL LSTM

Xiaolong Zhou, Jianing Lin, Jiaqi Jiang, Shengyong Chen

College of Computer Science and Technology
Zhejiang University of Technology
zxl@zjut.edu.cn;sy@ieee.org

ABSTRACT

Free-head 3D gaze estimation which outputs gaze vector in 3D space has wide application in human-computer interaction. In this paper, we propose a novel 3D gaze estimator by improving the Itracker and employing a many-to-one bidirectional LSTM (bi-LSTM). First, we improve the conventional Itracker by removing the face-grid and reducing one network branch via concatenating the two-eye region images to predict the subject's gaze of a single frame. Then, we employ the bi-LSTM to fit the temporal information between frames to estimate gaze vector for video sequence. Experimental results show that our improved Itracker obtains 11.6% significant improvement over the state-of-the-art methods on MPIIGaze dataset (single image frame) and has robust estimation accuracy for different image resolutions. Moreover, experimental results on EyeDiap dataset (video sequence) further bring 3% accuracy improvement by employing the bi-LSTM.

***Index Terms*—** Gaze estimation, Itracker, RNN, LSTM

1. INTRODUCTION

Gaze direction has been viewed as an important clue to speculate on the target's attention, which has wide application in human-computer interaction fields [1]. Although extensive gaze estimation methods have been explored, redundant calibration process, complex system settings, limitations of lighting conditions and the non-universal calibration for different subjects as well as low tolerance to head movement remain challenges for the exsiting gaze estimation systems.

Traditional manual-feature-driven gaze estimation methods normally achieve the position of gaze point by mapping the eigenvector formed by the local features such as corneal reflections or eye corners as well as iris contours to the final target. Feng et al. [2] preprocessed the pixel values of the whole eye region to form the feature, and applied the eigenvectors generated in the calibration process to linearly fit the eigenvectors in real-time prediction. Kacete et al. [3] used random forest regression to perform gaze estimation and combined with depth information to improve the result under large head pose. Wang et al. [4] proposed a KNN method based on the head pose and iris center position. They used head pose and iris center position as the criterion of classification and trained the class-independent regression model to fit the mapping relationship on the corresponding data. Although some appearance-based methods can achieve high accuracy, they tend to be poorly tolerant to variable head poses, illumination changes and need person-specified calibration.

Recent data-driven gaze estimation methods such as CNN-based methods have great potential to handle many traditional challenges, since they use a data-driven off-line training instead of cumbersome personal calibration and only use a web-cam to avoid tedious system setup. Zhang et al. [5] used CNN to map the eye images and head poses to gaze vector, which showed that the CNN-based method has higher accuracy than classic methods for various illumination and appearance differences. Afterwards, they improved the basic network from Alexnet to vgg16 [6], and put the head pose information into the penultimate layer. It got a better estimation result but increased the scale of the network. Deng et al. [7] first trained the separate model for head pose and eyeball movement, then aggregated them to estimate gaze vector from coarse to fine. This method had less potential for head-correlation over-fitting, but lacked evaluation on public datasets. Krafka et al. [8] separately input the whole face image and eye images to the corresponding branch and used face grid branch to locate face position to supply the location information for prediction of gaze point on the screen. However, this method was only limited to the 2D gaze estimation, the performance on 3D was not evaluated. Zhang et al. [9] directly input the normalized face to the Alex-based network and proposed a spatial weights CNN to reduce redundant information in face region. This method has demonstrated more robust result under significant variation in illumination conditions but appears to be more complexity, which is not friendly

This work was supported by National Natural Science Foundation of China (61876168, 61403342, 61325019), Zhejiang Provincial Natural Science Foundation of China (LY18F030020), and Hubei Key Laboratory of Intelligent Vision Based Monitoring for Hydroelectric Engineering (2017SDSJ03).

978-1-5386-9552-4/19/$31.00 ©2019 IEEE
DOI 10.1109/ICME.2019.00151

enough to different hardware.

To improve the 3D gaze estimation accuracy while keeping hardware friendly, we propose an improved Itracker as a static model to predict the final result for a single image frame gaze estimation and introduce the bidirectional LSTM (bi-LSTM) to simulate the temporal sequence information for consecutive frames gaze estimation. To the best of our knowledge, this is the first time that a bidirectional recurrent neural network has been introduced into gaze estimation to simulate temporal information.

2. PROPOSED METHOD

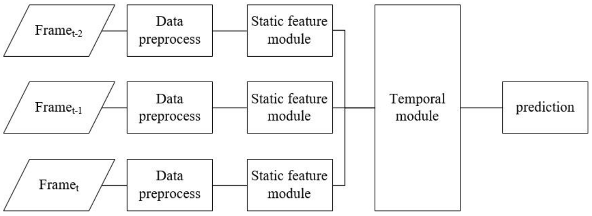


Fig. 1. The overall architecture of proposed 3D gaze estimation method.

The purpose of 3D gaze estimation is to learn a function f to map the image I to a 3D gaze vector g, where $g = f(I)$. This vector is originated from eyeball center or a reference point of the face. The 2D case can be obtained by intersecting the 3D gaze vector g with the specific 2D target plane. The overall architecture is shown in Fig.1.

2.1. Data Preprocessing

To weaken the effects of different head poses and various camera parameters on the final gaze estimation result, we make a certain perspective transformation on the original images so that we only need to train the model under the specific virtual space. This process greatly reduces the complexity of the fitting problem as well as the potential model size. The camera is first rotated by a conversion matrix so that the face reference point will be at the image center from a fixed distance, which could reduce the appearance variance. The face is then transformed into an image plane of a specific camera space through a transform matrix to reduce the negative effects of different camera configurations.

Assuming that a is the coordinate of the face reference point under camera space, we make the virtual camera face to the reference point by letting the z-axis of the virtual space be $v_z = \frac{a}{\|a\|_2}$, and then assuming that $H[h_x, h_y, h_z]$ is the rotation matrix of head pose. In order to make the x-axis parallel to the horizontal direction of head, we make $v_x = v_y \times v_z$, where $v_y = v_z \times h_x$. The rotation matrix R is computed as $R = [r_x, r_y, r_z]^T$. We assume that the distance between the virtual camera and the reference point is d. The conversion matrix $M = SR$ is finally used for the first step, where $S = diag\left(1, 1, \frac{d}{\|a\|_2}\right)$.

The second step is implemented by the warp matrix $W = C_o M C_n^{-1}$, where C_o is the intrinsic matrix of original camera and C_n is the intrinsic matrix of virtual camera that is determined by the size of output image.

Similar to the transformation of the image, we also need to convert the corresponding gaze label during training procedure. Let $g_n = Rg_o$, where g_n denotes the normalized gaze label and g_o denotes the original gaze label. Then, we represent it by Euler angle to release the constraint relationship of unit vector. In test phase, for each prediction result, we need to convert them from virtual space back to the original camera space, so the result is obtained from $g_o = R^{-1}g_n$.

2.2. Network Architecture

We propose a network architecture combined with bi-LSTM to incorporate temporal information for 3D gaze estimation. We notice that a face grid module has been added to the network structure in [8], which had a greater impact on the final estimation results. Since the method in [8] needs to obtain the exact gaze point on the device, in addition to predicting the gaze direction, it is necessary to know exactly the head position in camera space. This information is primarily provided by the face grid module. However, since we mainly discuss how to efficiently and accurately get the 3D gaze direction in this paper, we ignore the face grid module in our topology.

To reduce the network size as much as possible while ensuring accuracy, we concatenate the left and right eye images to form a single 6 channels input. In this way, we can successfully reduce nearly half the number of the network parameters (from approximately 3.6 million to 1.8 million) without any reduction in the final estimation performance.

It can also be seen from [6] and [10] that the additional face landmarks or head pose information has little impact on the final result. Because of the rich combination of hyper parameters, we cannot conclude that this weak improvement is due to the introduction of these additional structured information. So, in this paper, for simplicity we don't include these extra shape cues or head poses information in our network. All non-linear relationships are directly encoded by the network. The final static feature module is depicted in Fig.2.

To the best of our knowledge, existing gaze estimation models rarely use the temporal information. In this paper, we use the bi-LSTM to simulate the temporal relationship to increase the accuracy and robustness of the network. The overall architecture is shown in Fig.1, which is divided into two modules: static module and temporal module. The static module learns features from the separate face and eyes appearance. It consists of a two-branch CNN and unified FC layers. One branch CNN extracts the features from normal-

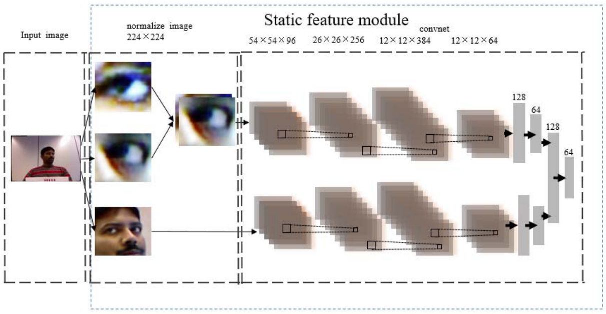


Fig. 2. The static feature extraction module.

ized face and the other from concatenated eyes image. The FC layers combine these two parts of features and learn a joint representation for the fused features. The learned features are then input to the many-to-one bi-LSTM. A linear regression is finally used to get the predicted result in normalized space from the hidden units in last time step.

2.3. Temporal Module Description

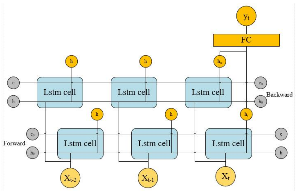


Fig. 3. The temporal module.

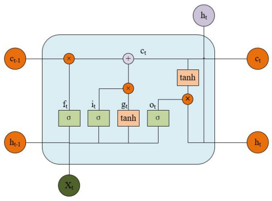


Fig. 4. The structure of single LSTM cell.

The overall structure of temporal module is shown in Fig.3. The LSTM structure contains a series of repeated L-STM cells, and the structure of the LSTM cell is shown in Fig.4. Each LSTM cell contains three multiplicative units that represent the forget gate, the input gate, and the output gate. These multiplicative units allow LSTM memory cells to store and transfer information over a long period of time. In Fig.3, (x_t, c_{t-1}, h_{t-1}) indicates the input layers and (h_t, c_t) indicates the output layers. Next, we briefly introduce how a standard LSTM cell generates outputs from inputs.

At time step t, the forget gate, the input gate and the output gate are represented as f_t, i_t, o_t respectively. The LSTM cell first uses the forget gate to filter out the information that needs to be retained.

$$f_t = \sigma \left(W_{if} x_t + b_{if} + W_{hf} h_{t-1} + b_{hf} \right) \tag{1}$$

where (W_{if}, b_{if}) and (W_{hf}, b_{hf}) respectively represent the weight matrix and bias term mapping input layer and hidden layer to the forget gate. The σ is the gate activation function, which is selected as the sigmoid function in this paper.

Then, the LSTM cell uses the input gate to incorporate valid information.

$$g_t = tanh \left(W_{ig} x_t + b_{ig} + W_{hg} h_{t-1} + b_{hg} \right) \tag{2}$$

$$i_t = \sigma \left(W_{ii} x_t + b_{ii} + W_{hi} h_{t-1} + b_{hi} \right) \tag{3}$$

$$c_t = f_t c_{t-1} + i_t g_t \tag{4}$$

where (W_{ig}, b_{ig}) and (W_{hg}, b_{hg}) respectively represent the weight matrix and bias term mapping the input layer and hidden layer to the cell gate. (W_{ii}, b_{ii}) and (W_{hi}, b_{hi}) respectively represent the weight matrix and bias term mapping the input layer and hidden layer to the input gate.

Finally, the LSTM cell gets the output hidden layer from the output gate.

$$o_t = \sigma \left(W_{io} x_t + b_{io} + W_{ho} h_{t-1} + b_{ho} \right) \tag{5}$$

$$h_t = o_t tanh \left(c_t \right) \tag{6}$$

where (W_{io}, b_{io}) and (W_{ho}, b_{ho}) respectively represent the weight matrix and bias term mapping the input layer and hidden layer to the output gate.

As shown in Fig.3, bi-LSTM contains a forward LSTM layer and a backward LSTM layer. In this paper, a sequence is composed by three image frames. The final gaze prediction is obtained by a fully connected layer. This layer maps the hidden layers got from forward and backward units of the last frame in time t to the final two-dimensional gaze vector g.

$$g = fc \left(h_t, h_{tr} \right) \tag{7}$$

2.4. Implement Details

We use a reduced version of the convolution layers of Alexnet as the basic network for each branch. Each basic network has 4 convolution layers. The face branch is connected to a 128-dimensional FC layer followed by a 64-dimensional FC layer while the eyes branch is connected to a 64-dimensional FC layer. Then, these two parts of the features are combined through a 64-dimensional FC layer and regularized by a Dropout layer to prevent over-fitting problem. If it is a static model, the final prediction results could be obtained directly through a 2-dimensional FC layer. Else, the 64-dimensional features would be used as input to the temporal model. In this paper, we use bi-LSTM as the temporal model which has 1 LSTM layer and 32 hidden units.

In the temporal model, we use a stage-wise training approach. First, we train the static model from scratch and do not use any data augmentation processing to ensure the high reproducibility of the experimental results. Next, we treat the static model as a deep feature extractor whose parameters are frozen and no longer adjusted during the second training stage. We re-arrange the training data by a sliding window fashion. Every successive three frames form a sequence and the last frame of each sequence is treated as the ground truth.

We train the model using Euclidean loss with the Adam optimizer. The basic learning rate is set to 0.0001 and the probability of dropout is set to 0.3. The batch size is 100 while the epoch is 20 for both static and temporal models.

3. EXPERIMENTS

3.1. Datasets

We evaluate the proposed method on two publicly available datasets: MPIIGaze [5] and EyeDiap [11]. For the MPIIGaze dataset, we take the center of the six provided face landmarks as the start point of gaze vector as well as the facing point of the virtual camera. In data preprocessing step, to reduce the illumination variance, we apply adaptive histogram equalization on each input image. MPIIGaze dataset has a total of 15 participants. For the EyeDiap dataset, we take the midpoint of the provided two iris centers as the origin of gaze vector as well as the facing point of the virtual camera. Similar with MPIIGaze, we apply adaptive histogram equalization to reduce illumination variance. The gaze targets on this dataset fall into two categories: screen targets and floating targets. To facilitate comparison, we only use the screen targets for evaluation and sample one image per 15 frames from 4 VGA videos of each participant. We filter out frames that meet the following conditions: (1) The participant is not looking at the screen; (2) The annotation is not provided properly; (3) The gaze angle is violating the physical constraints $(|theta| \leq 40^o, |phi| \leq 30^o)$.

3.2. Cross Person/ Group Evaluation and Within Person Evaluation

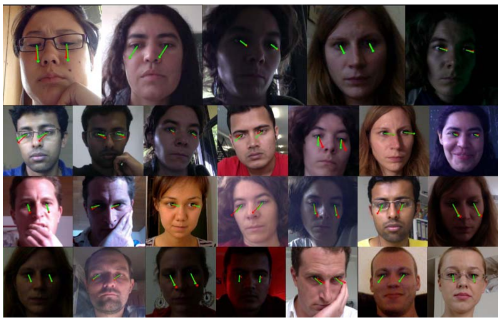


Fig. 5. Some prediction results on MPIIGaze, green lines indicate the predictions and red lines indicate the ground truth.

We conduct cross person/group evaluation to show the basic performance of our method. Since it is a 3D gaze estimation, we use the angle error between the prediction value and ground truth to indicate the prediction accuracy. Table 1 and Table 2 show the comparison between our method and other state-of-the-art methods on MPIIGaze and EyeDiap datasets, respectively. Table 1 shows that our method has achieved excellent result on the MPIIGaze evaluation. MPIIGaze dataset covers significant variation in illumination. Fig.5 shows that our method can guarantee high accuracy in a very different background which indicates that our method is robust to various illumination challenge. Table 2 shows that our method ranks the 2nd on the EyeDiap evaluation, but it still has a significant advantage in network complexity over the ranked 1^{st} method. Network parameters comparison: ranked 1^{st} (about 130 million) vs ournet (about 1.8 million). It indicates that our method needs to be improved to better balance the complexity and accuracy under large head pose environment. We have selected two state-of-the-art face-based gaze estimation methods for further comparison: (1) AlexSW, a state-of-the-art full-face-based method proposed in [9];(2) Itracker, a fundamental method used in this paper proposed in [8]. For a fair comparison, the image normalization process is the same as used in this paper, and the final output is resized to a resolution of 224 × 224.

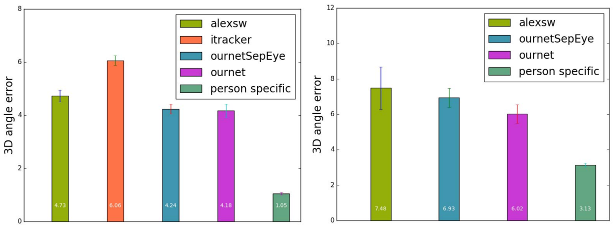
Table 1. Comparison result with the state-of-the-art methods on MPIIGaze.

Methods	3D degrees error
Baltrusaitis T et al. 2016 [10]	9.96
Wood E et al.2016 [12]	9.58
Shrivastava A et al. 2016 [13]	7.8
Nie S et al. 2018 [14]	7.1
Zhang X et al. 2015 [5]	6
Krafka K et al. 2016 [8]	5.6
Zhang X et al. 2017 [6]	5.4
Zhang X et al. 2017 [9]	4.8
our static model	4.18

Table 2. Comparison result with the state-of-the-art methods on EyeDiap.

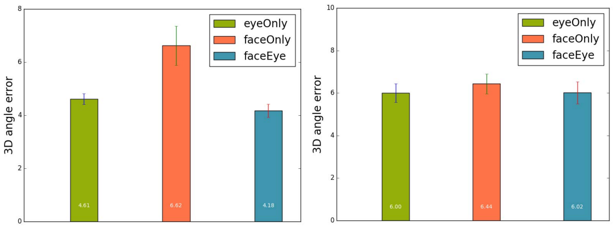
Methods	3D degrees error
Krafka K et al. 2016 [8]	8.3
Park S et al.2018 [15]	7.4
Zhang X et al. 2017 [9]	6.0
Palmero C et al. (temporal) 2018 [16]	3.4
Our static model	6.02
Our static + bi-LSTM model	5.84

Fig.6(a) shows the comparison result between our method and other state-of-the-art methods on MPIIGaze dataset. Our method has a significant 11.6% improvement over the state-of-the-art methods. Meanwhile, the overall complexity of our network is much smaller than [9], which is the first bar on the chart showed in Fig.6(a). The third bar shows the result of



(a) MPIIGaze (b) EyeDiap

Fig. 6. Cross person and person specific evaluation results on MPIIGaze and EyeDiap.



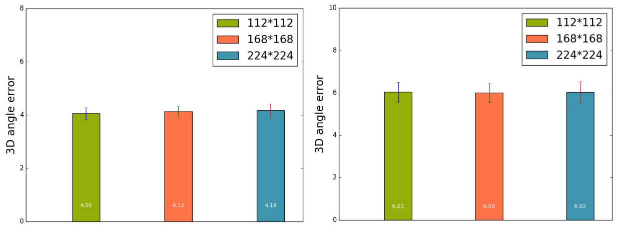
(a) MPIIGaze (b) EyeDiap

Fig. 7. Ablation study on MPIIGaze and EyeDiap.

two separate eyes part network structure. The combination of the two eyes does not reduce any accuracy but even give a slight improvement. The last bar is the result of within-person evaluation. We use all the data to train the model, and then we have a separate evaluation for each single person. The evaluation of this part is mainly for the purpose of demonstrating the potential (upper bound) of our network. Fig.6(b) shows the comparison result on EyeDiap dataset. Similarly, our method has a small improvement in accuracy compared to the baselines. However, it should be noted that for the method in [9], the result given in our chart are not as good as that mentioned in the original paper. The 3D degree error on EyeDiap in [9] is 6.0, that is to say, our method has similar accuracy compared with [9]. But we still have great advantages in terms of network size. At the same time, it can be seen from the second and third bars that even better results are obtained after the eye parts are concatenated. The last bar demonstrates that our method performs poor even in within person evaluation on EyeDiap dataset compared to the result on MPIIGaze dataset, which indicates that our method degrades when encountering large head pose but has robust result respond to various illumination conditions.

3.3. Network Parts Evaluation

We split the network into two separate branches (eye module and face module) to verify the role of each network module.



(a) MPIIGaze (b) EyeDiap

Fig. 8. Resolution study on MPIIGaze and EyeDiap.

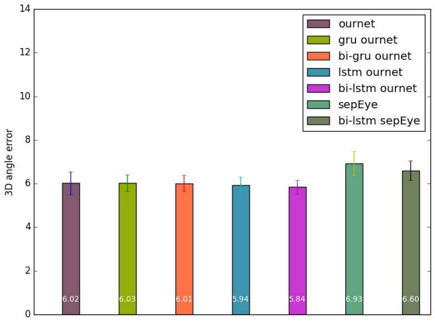


Fig. 9. RNN evaluation on EyeDiap.

Fig.7(a) shows the results of our evaluation on MPIIGaze. It shows that the final prediction accuracy is mainly depending on the eyes branch network and the face part also contributes a bit. Similarly, Fig.7(b) shows the results of our evaluation on EyeDiap. It demonstrates that the face branch is less important on the EyeDiap evaluation, which means that we may design a more efficient way to get the prediction that only requires the eyes part as input.

3.4. Resolution Evaluation

Gaze estimation system is normally required to maintain accuracy over a wide range of distances. Although the normalization process can greatly reduce the input variance caused by different distances by rescaling the image to the proper resolution, it still cannot avoid the loss of useful information which would result in a decline in the final estimation. In order to simulate this information loss due to various distances, down-sampling is performed on the input images as follows: (1) Input image 224×224 is downscaled to 168×168 and upscaled to 224×224; (2) Input image 224×224 is downscaled to 112×112 and upscaled to 224×224.

Fig.8 show the results of resolution experiments performed by our method on MPIIGaze and EyeDiap, respectively. The results illustrate that our method is very robust to different distances. Even if the image distance is twice as far as the origin, the result of our method still not degrade.

3.5. Temporal Model Evaluation

In this session, we evaluate the contribution of adding the temporal model to the static model. Since the MPIIGaze dataset is a discontinuous single image format, we only do evaluation on EyeDiap dataset. Fig.9 shows our evaluation results. The first bar is the result of our static model and the second to fifth bars are the results of the four different RNN models combined with the static model, while the sixth and seventh bars are the results of the network model where the left and right eyes are input as separate branches. The results illustrate that no matter which basic network used, the evaluation result always be improved when combining with the temporal model. In all the RNN models used in this paper, the bi-LSTM model contributes the most by improving the accuracy of the static model by about 3%. What's more, all bidirectional RNN models have better results than common RNN models. It shows that we can better improve the final estimation accuracy when adding the backpropagation information to the RNN temporal model.

4. CONCLUSION

In this paper, we have presented a novel 3D gaze estimation method by combining an improved Itracker and bi-LSTM. we have effectively modified the Itracker model to improve the estimate accuracy as well as reduce the network parameters. The resolution experimental results have demonstrated that our network is robust to images with different resolutions. In contrast to the state-of-the-art methods, our static method not only significantly improves the accuracy, but also greatly reduces the network size. For gaze estimation in video sequence, we have introduced a bi-LSTM to fit the temporal information, which brings improvement on the final estimation result.

5. REFERENCES

[1] Xiaolong Zhou, Haibin Cai, Youfu Li, and Honghai Liu, "Two-eye model-based gaze estimation from a kinect sensor," in *ICRA*, 2017, pp. 1646–1653.

[2] Feng Lu, Yusuke Sugano, Takahiro Okabe, and Yoichi Sato, "Inferring human gaze from appearance via adaptive linear regression," in *ICCV*, 2011, pp. 153–160.

[3] Amine Kacete, Renaud Séguier, Michel Collobert, and Jérôme Royan, "Unconstrained gaze estimation using random forest regression voting," in *ACCV*, 2016, pp. 419–432.

[4] Yafei Wang, Tongtong Zhao, Xueyan Ding, Jinjia Peng, Jiming Bian, and Xianping Fu, "Learning a gaze estimator with neighbor selection from large-scale synthetic eye images," *Knowledge-Based Systems*, vol. 139, pp. 41–49, 2018.

[5] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling, "Appearance-based gaze estimation in the wild," in *CVPR*, 2015, pp. 4511–4520.

[6] X. Zhang, Y Sugano, M Fritz, and A Bulling, "Mpiigaze: Real-world dataset and deep appearance-based gaze estimation," *IEEE Trans Pattern Anal Mach Intell*, vol. PP, no. 99, pp. 1–1, 2017.

[7] Haoping Deng and Wangjiang Zhu, "Monocular freehead 3d gaze tracking with deep learning and geometry constraints," in *ICCV*, 2017, pp. 3162–3171.

[8] Kyle Krafka, Aditya Khosla, Petr Kellnhofer, Harini Kannan, Suchendra Bhandarkar, Wojciech Matusik, and Antonio Torralba, "Eye tracking for everyone," in *CVPR*, 2016, pp. 2176–2184.

[9] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling, "It's written all over your face: Full-face appearance-based gaze estimation," in *CVPRW*, 2017, pp. 2299–2308.

[10] Tadas Baltrušaitis, Peter Robinson, and Louis-Philippe Morency, "Openface: an open source facial behavior analysis toolkit," in *WACV*, 2016, pp. 1–10.

[11] Kenneth Alberto Funes Mora, Florent Monay, and Jean-Marc Odobez, "Eyediap: A database for the development and evaluation of gaze estimation algorithms from rgb and rgb-d cameras," in *ETRA*, 2014, pp. 255–258.

[12] Erroll Wood, Tadas Baltrušaitis, Louis-Philippe Morency, Peter Robinson, and Andreas Bulling, "Learning an appearance-based gaze estimator from one million synthesised images," in *ETRA*, 2016, pp. 131–138.

[13] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb, "Learning from simulated and unsupervised images through adversarial training.," in *CVPR*, 2017, vol. 2, p. 5.

[14] Siqi Nie, Meng Zheng, and Qiang Ji, "The deep regression bayesian network and its applications: Probabilistic deep learning for computer vision," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 101–111, 2018.

[15] Seonwook Park, Xucong Zhang, Andreas Bulling, and Otmar Hilliges, "Learning to find eye region landmarks for remote gaze estimation in unconstrained settings," *arXiv preprint arXiv:1805.04771*, 2018.

[16] Cristina Palmero, Javier Selva, Mohammad Ali Bagheri, and Sergio Escalera, "Recurrent cnn for 3d gaze estimation using appearance and shape cues," *arXiv preprint arXiv:1805.03064*, 2018.