Gaze360: Physically Unconstrained Gaze Estimation in the Wild (Supplemental)

Petr Kellnhofer*[1], Adrià Recasens*[1], Simon Stent[2], Wojciech Matusik[1], and Antonio Torralba[1]

[1] Massachusetts Institute of Technology, Cambridge MA 02139, USA
[2] Toyota Research Institute, Cambridge, MA, 02139, USA
{pkellnho,recasens,wojciech,torralba}@csail.mit.edu
simon.stent@tri.global

1. Supplementary video

We prepared an explanatory video which includes an example of a recording session during the Gaze360 dataset capture, as well as showing our model working in very diverse in-the-wild situations (from YouTube videos) such as TV shows or a tennis match. We also show a video sequence of the supermarket application presented in the paper, with the corresponding attention prediction. These examples suggest that our system can be applied to a variety of tasks in human interaction and understanding where eye gaze is a useful cue.

2. A note on eye appearance at extreme angles

Our dataset is unique particularly in its coverage of gaze yaw (see Figure 4 of the main submission). It is reasonable to ask why you might want to track the gaze of a subject at high levels of gaze yaw - as this is likely to coincide with the subject's head pointing away from the camera and their eyes being self-occluded. Such scenarios do not exist in existing datasets and break all existing gaze trackers, particularly those which rely on eye detection as a pre-processing step. As shown in Fig. 1, at head yaws of 90° and even 135°, eye appearance can still provide a strong cue for gaze direction. Even under a lower resolution than presented in the figure, the cue is strong enough to produce a better-than-chance estimate of gaze direction. One setting where this may be useful is for a robot or autonomous vehicle to estimate the situational awareness of a human road user. Eyes with the appearance of the left-hand column are far more likely to be aware of the camera than eyes with the appearance of the right hand column. Detecting such a cue may affect the prediction of a road-user's future behavior.

3. Further domain adaptation details

For clarity, we specify the domain adaptation losses explicitly.

Figure 1. **Eye appearance at extreme angles**. Top row: at 90° head yaw, looking right, up, down and left; bottom row: at 135° head yaw, looking right, up, down and left.

3.1. Additional loss functions

Discriminator loss. When performing domain adaptation experiments, we use a discriminator D along with a discriminator loss which penalizes the network for producing discriminative features for the different domains. If y_i is the binary variable describing the domain of sample i, our discriminator loss L_D is:

$$L_D = -\sum_i y_i \log(1 - D(x_i)) + (1 - y_i) \log D(x_i) \quad (1)$$

for i in the mini-batch, where x are the outputs of the backbone network. The loss follows the structure of a cross-entropy loss but with the labels flipped, as the goal of the network is to confuse the discriminator. For the discriminator network, D we use a fully connected layer. It is important to note that the discriminator loss is used to train the gaze network, while the discriminator itself is trained with a cross-entropy loss as presented in Sec. 3.2.

Symmetry loss. The symmetry loss further helps to regularize the network on new domains of data. Let $I_i^{'}$ be the flipped version of I_i, an input crop to the network. Then, we compute $(\theta_i, \phi_i, \sigma_i) = f(I_i)$ and $(\theta_i^{'}, \phi_i^{'}, \sigma_i^{'}) = f(I_i^{'})$.

Table 1. **Cross-dataset ablation study.** The table below reports the mean angular errors for the different ablations of the proposed domain adaptation method.

Train \ Test	Columbia	MPII FaceGaze
Gaze360 + DA	8.3	10.6
OG + DA (No discriminator)	8.6	10.4
OG + DA (No symmetry)	8.9	12.2

To compute the symmetry loss, we use $(-\theta_i^{'}, \phi_i^{'})$ as ground truth for the image I_i, and then apply the Pinball loss to compute the final symmetry loss. Note that training with this loss by itself with no measure of absolute error leads to the model estimating a trivial solution on the line of symmetry.

3.2. Training the discriminator

The discriminator D is trained using a standard cross-entropy loss to classify each sample I_i into its associated domain. For training, we use Adam and the same learning rate schedule as the gaze network. The discriminator is trained in sequence with the gaze network, updating the weights of both of them at every iteration.

3.3. Ablation study

Our final results report the use of both a discriminator loss and a symmetry loss. In Table 1 we report comparative results from using just one of the two additional losses for the MPII FaceGaze dataset and the Columbia dataset. As it has been reported in the paper, the RT-GENE dataset contains significant noise in their images, which makes the evaluation insignificant. We conclude that using both the discriminator and the symmetry loss is overall the best strategy when doing domain adaptation.

4. Further dataset collection details

4.1. Subject positioning

To build the dataset, we use AlphaPose [1] to detect the position of head features and feet of subjects in rectified frames from each camera unit independently. For very close subjects whose feet are beyond the camera field of view, we use the average body proportions of standing subjects to estimate their feet position from their hip position. The Ladybug camera provides a 3D ray in a global Ladybug Cartesian coordinate system $L = [\mathbf{L}_x, \mathbf{L}_y, \mathbf{L}_z]$ (Fig. 2b) for every image pixel. We use it to derive the position of feet and eyes in spherical coordinates (angles α and β in the 2D slice presented in Fig. 2a). The remaining unknown variable is the distance to eyes d. We exploit a measured camera height above the ground plane h and an assumption that both the camera and all subjects stand on the same horizontal plane.

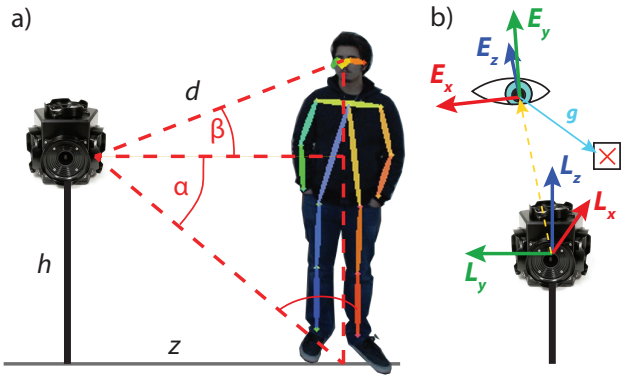

Figure 2. **Setup parameters**: for estimating (a) the subject's eye distance from camera using a ground plane assumption, and (b) the gaze transform between the subject's eye coordinate system (E) and the Ladybug camera's coordinate system (L). Positive E_z is pointing away.

Although this limits our training data collection to flat surfaces, it is not restrictive at test-time. We then use trigonometry to compute the horizontal distance between the camera base and subject's feet z and finally the distance between the camera and the subject's eyes d as

$$z = h \cdot \tan\left(\frac{\pi}{2} - \alpha\right) \quad (2)$$

$$d = \frac{z}{\cos\beta}. \quad (3)$$

This allows us to compute the position of the eyes, $\mathbf{p}_e$, in the Ladybug coordinate system.

4.2. Dataset cleaning

While the generation of ground truth gaze labels in our approach was fully automatic, we took several steps to ensure that the dataset was clean, by removing any false and unreliable subject and target detections from the data. First, we detected the marker for every frame and discarded frames where the detection failed due to occlusion or illumination issues. Next, we detected all the people in the image using AlphaPose and used their body skeletons to estimate head bounding boxes. We used a simple tracker based on head bounding box intersection to assign identities to subjects across as many frames per camera as possible. We then computed the mean distance of each identity from the marker board. The identities of subjects positioned closer to the target than 1.2m on average were removed since they corresponded to the investigator manipulating the board. We further thresholded any persons beyond 3.5m from the camera, who consisted of passers-by or other members of the research team. We confirmed that the detected head bounding box was within the valid part of the rectified image data to remove partially-detected subjects whose heads were outside the rectified image. Finally,

we visually inspected 6 uniformly sampled frames for each detected identity and removed those that did not belong to our subject pool.

5. Notes on other gaze datasets

In Fig. 3 we reprint example images from datasets **Columbia** [3], **MPIIFaceGaze** [4], **RT-GENE** [2] and our own **Gaze360**, used in our Cross-dataset evaluation (Sec. 6.2 of the paper). Note the variation in visual quality, environment, illumination, subject appearance and gaze or head pose. Our dataset is the only dataset considering the full range of possible viewpoints and also the only one covering both indoor and outdoor scenarios. The MPIIFaceGaze dataset captures various levels of mostly artificial illumination, while illumination on RT-GENE and the Columbia dataset is uniform over the dataset. Our dataset additionally includes variation in natural illumination from cloudy to clear sky. The most significant visual difference between the domains is in the presence of graphical artifacts. The Columbia dataset features very high quality close-up pictures but the faces (though not eyes) are partially occluded by the chin-rest. The MPIIFaceGaze images come from a web camera and the quality depends on the external illumination. The RT-GENE dataset was post-processed using a Generative Adversarial Network to remove the appearance of a wearable eye tracker used to annotate the data. The approach leaves visible artifacts in many of the frames. This is likely confusing the models trained outside of this domain and causing the high error reported in Table 3 in the paper. Finally, our dataset's image quality varies with external illumination intensity and with subject distance from the sensor. Close up pictures contain lot of facial details and can be compared to the samples from MPIIFaceGaze while the images taken from a larger distance contain significant blur and/or noise and present a challenging scenario for any model to solve which, however, is essential for practical application of gaze estimation in open crowded environments.

6. Additional test data outputs

In Fig. 4, we show more random samples of our model output on an unseen test split of the Gaze360 dataset. We compare the collected ground truth gaze (yellow) with our prediction (red). Beneath each panel, we also show the actual error bar of our model along with the predicted quantile by our system.

7. Additional supermarket application outputs

In Fig. 5, we show further examples of gaze heat maps acquired from the supermarket application from Section 7 of the main submission. Although the maps contain some variance, they are concentrated significantly close to the object of interest (red bounding box).

References

[1] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. Rmpe: Regional multi-person pose estimation. In *ICCV*, 2017. 2

[2] Tobias Fischer, Hyung Jin Chang, and Yiannis Demiris. RT-GENE: Real-time eye gaze estimation in natural environments. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 334–352, 2018. 3, 4

[3] Brian A Smith, Qi Yin, Steven K Feiner, and Shree K Nayar. Gaze locking: Passive eye contact detection for human-object interaction. In *UIST*, 2013. 3, 4

[4] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. It's written all over your face: Full-face appearance-based gaze estimation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops, Honolulu, HI, USA, July 21-26, 2017*, pages 2299–2308, 2017. 3, 4

Figure 3. **Sample images from gaze datasets**: samples were selected to show variance in visual quality, environment, illumination, subject appearance and gaze or head pose. Individual images belong to the authors of the respective datasets [3, 4, 2] and are re-printed here for illustration purposes. Our dataset (bottom) is unique in capturing a large number of subjects, in a range of lighting conditions and environments, with the widest range of gaze variation of any existing dataset.

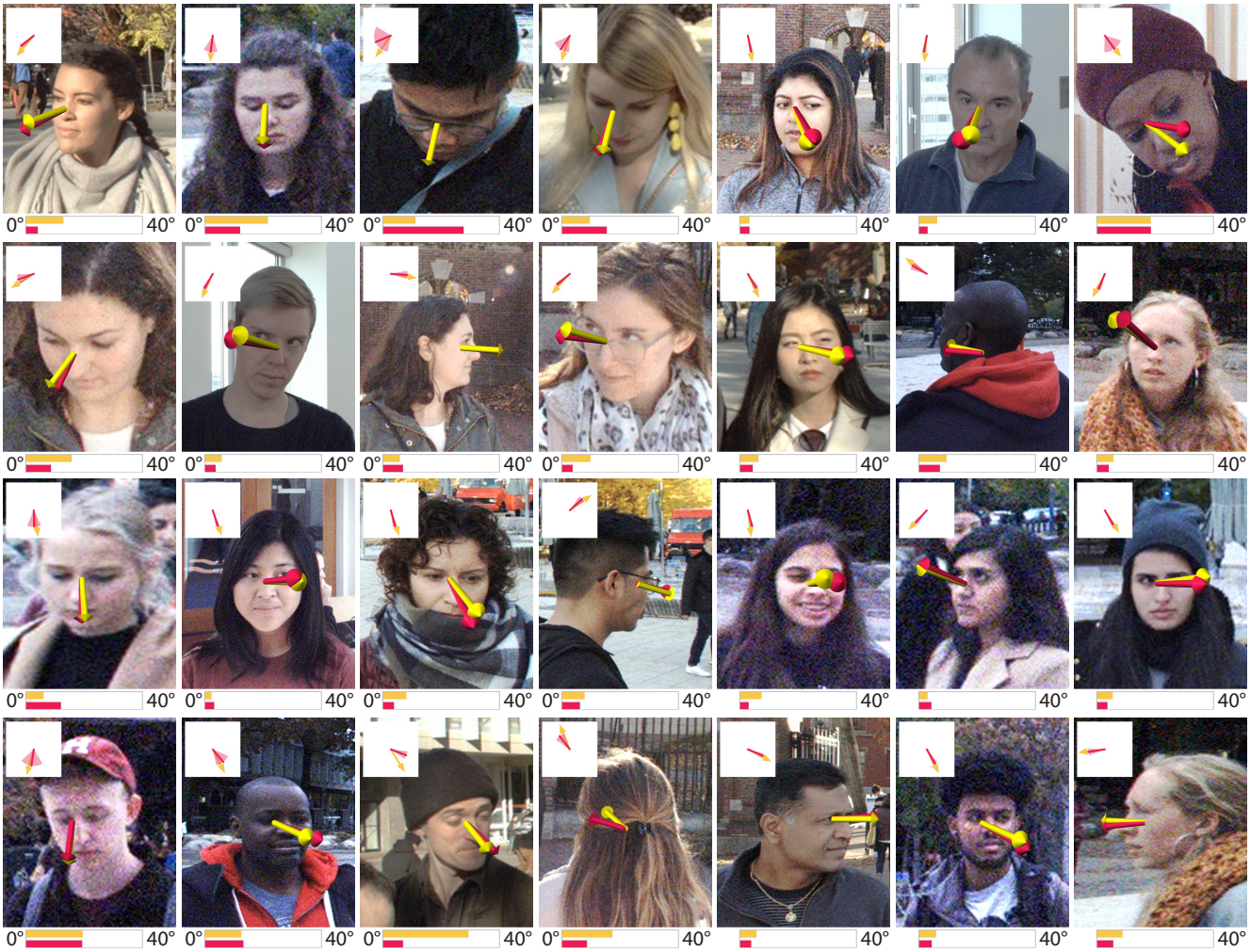

Figure 4. **Test set examples**: ground truth gaze (yellow) and Gaze360 predictions (red) are shown for unseen test subjects. The bars denote actual (yellow) and predicted (red) errors in degrees. The inset shows a top-down view of the gaze estimates and the predicted error versus ground truth.

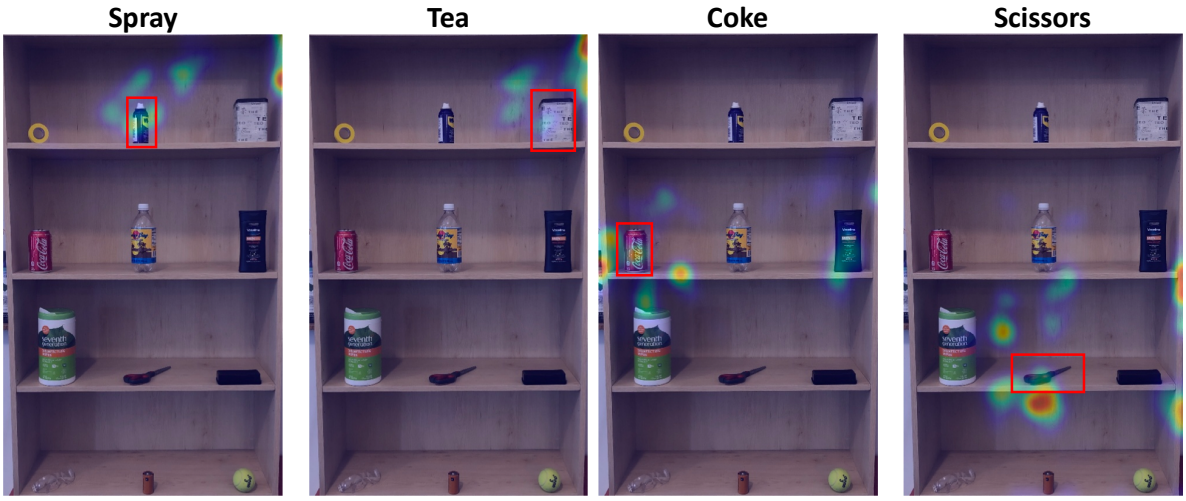

Figure 5. **Further examples from the supermarket application** from Section 7 of the main submission.