Gaze Estimation: MPIIGaze Dataset

Aakash K.
aakashln@iitk.ac.in
Advisor: Prof. Stefan Winkler, Deputy Director, AI Singapore

Abstract

Learning-based methods are believed to work well for unconstrained gaze estimation, i.e. gaze estimation from a monocular RGB camera without assumptions regarding user, environment, or camera. However, current gaze datasets were collected under laboratory conditions. In this report, we present various computer vision models and their ensembles for the prediction of gaze angle on the MPIIGaze dataset containing images shot from the laptop of 15 individuals.

1 Introduction

Gaze estimation is well established as a research topic in computer vision because of its relevance for several applications, such as gaze-based human-computer interaction [1] or visual attention analysis [2], [3]. Most recent learning-based methods leverage large amounts of both real and synthetic training data [4], [5], [6], [7] for person-independent gaze estimation. They have thus brought us one step closer to the grand vision of unconstrained gaze estimation: 3D gaze estimation in everyday environments and without any assumptions regarding users' facial appearance, geometric properties of the environment and camera, or image formation properties of the camera itself. Unconstrained gaze estimation using monocular RGB cameras is particularly promising given the proliferation of such cameras in portable devices [8] and public displays

While appearance-based gaze estimation techniques that use Convolutional Neural Networks (CNN) have significantly surpassed classical methods [57] for in the-wild settings, there still remains a significant gap towards applicability in high-accuracy domains [1]. Gaze estimation from images is difficult because it requires either explicit or implicit fitting of a person-specific eye-ball model to the image data and the estimation of their visual and optical axes. Moreover, it is well understood that inter-subject anatomical differences affect gaze estimation accuracy.

In this project we tried to overcome various factors affecting gaze estimation like illumination conditions, personal appearance variation and image quality variations. First, we try to test the Faze DT-ED model on the dataset where both training and testing data comes from MPIIGaze dataset using 1 person leave-out technique. Later we try to combine Faze model as well as resnet-preact model to create an ensemble model.

2 Related Work

2.1 Gaze Estimation Methods

Gaze estimation methods can generally be distinguished as model-based or appearance-based.

Model-based methods use a geometric eye model and can be further divided into corneal-reflection and shape-based methods. Corneal-reflection methods rely on eye features detected using reflections of an external infrared light source on the outermost layer of the eye, the cornea. Early works on corneal reflection-based methods were limited to stationary settings [23], [24], [25], [26] but were later extended to handle arbitrary head poses using multiple light sources or cameras [27], [28]. Shape-based methods

[29], [30], [31], [32] infer gaze directions from the detected eye shape, such as the pupil or iris edges. Although model-based methods have recently been applied to more practical application scenarios [8], [33], [34], [35], [36], their gaze estimation accuracy is still lower, since they depend on accurate eye feature detections for which high-resolution images and homogeneous illumination are required. These requirements have largely prevented these methods from being widely used in real-world settings or on commodity devices.

Appearance-based gaze estimation [46] methods that map images directly to gaze have recently surpassed classical model-based approaches [13] for in-thewild settings. Earlier approaches in this direction assume images captured in restricted laboratory settings and use direct regression methods [28, 27] or learning-by-synthesis approaches combined with random forests to separate headpose clusters [45]. More recently, the availability of large scale datasets such as MPIIGaze [57] and GazeCapture [22], and progress in CNNs have rapidly moved the field forward. MPIIGaze has become a benchmark dataset for in-the-wild gaze estimation. However, person-independent gaze errors are still insufficient for many applications [3, 43, 19, 2]. While significant gains can be obtained by training person-specific models, it requires many thousands of training images per subject [59].

2.2 The Face Gaze Paper

The authors design FAZE (Fig. 1) with the understanding that a person-specific gaze estimator must encode factors particular to the person, yet at the same time, leverage insights from observing the eye-region appearance variations across a large number of people with different head pose and gaze direction configurations. The latter is important for building models that are robust to extraneous factors such as poor image quality. Thus, the first step in FAZE is to learn a generalizable latent embedding space that encodes information pertaining to the gaze-direction, including person-specific aspects.

They extend the transforming encoder-decoder architecture [15, 53] to consider three distinct factors apparent in our problem setting: gaze direction, head orientation, and other factors related to the appearance of the eye region in given images (Fig. 2). They then disentangle the three factors by explicitly applying separate and known differences in rotations (eye gaze and head orientation) to the respective sub-codes. This architecture as the Disentangling Transforming Encoder-Decoder (DT-ED).

For a given input image x , we define an encoder $E: x ß z$ and a decoder $D: z ß \hat{x}$ such that $D(E(x)) = \hat{x}$. We consider the latent space embedding z as being formed of 3 parts representing: appearance (z^a), gaze direction or eyeball rotation (z^g), and head pose (z^h), which can be expressed as: $z = \{z^a; z^g; z^h\}$ where gaze and head codes are flattened to yield a single column. We define z^g as having dimensions $(3XF_g)$ and z^h as having dimensions $(3XF_h)$ with $F \in N$. With these dimensions, it is possible to apply a rotation matrix to explicitly rotate these 3D latent space embeddings using rotation matrices.

The Rotation Technique

The frontal orientation of eyes and heads in our setting can be represented as $(\theta = 0, \phi = 0)$ in Euler angles notation for azimuth and elevation, respectively assuming no roll(handled in preprocessing) , and using the x y convention. Then, the rotation of the eyes and the head from the frontal orientation can be described using (θ_g, ϕ_g) and (θ_h, ϕ_h), in Euler angles and converted to rotation matrices defined in

Now they calculate $R^g_{ab} = R^g_a (R^g_b)^{-1}$ and $R^h_{ab} = R^h_a (R^h_b)^{-1}$. These matrices can be calculated since we have all the angles using ground truth head pose and gaze directions. Now it is postulated that by applying $\hat{z_b}^g = R^g_{ab} z^g_a$ and $\hat{z_b}^h = R^h_{ab} z^h_a$ and then by passing $\{\hat{z_b}^a = z^a_a; \hat{z_b}^g; \hat{z_b}^h\}$ into decoder we get an image of person 'b' with similar gaze and head pose of person 'a'.

An experiment was performed by authors after training of DT-ED model to randomly apply pitch and yaw rotation of about 15 degrees to produce various synthetic images as shown in Figure 2.

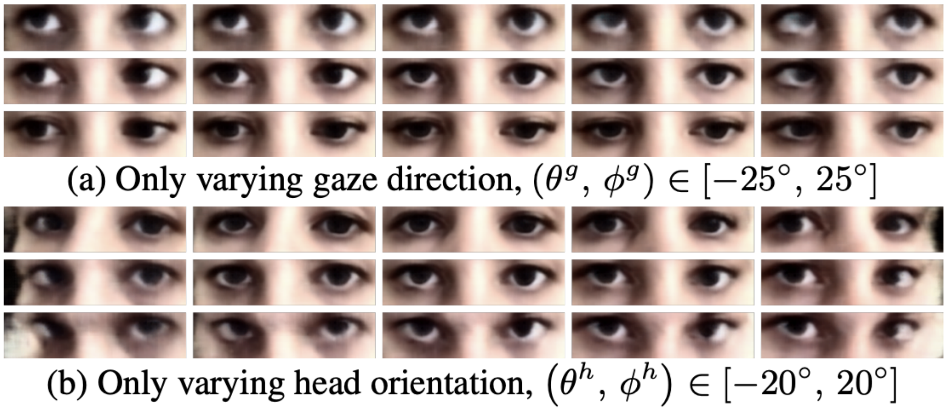
Note: This DT-ED model can now be used to generate images with varying gaze angles.

Training DT-ED:

The main aim of this model is to disentangle gaze angle, head pose and personal appearance of the participants. Firstly, for disentangling firstly for a training sample of person 'a' we get rotated codes

$$\mathbf{R}^{(\theta,\,\phi)} = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix}$$

Figure 1: Rotation Matrix Definition



(a) Only varying gaze direction, $(\theta^g,\, \phi^g) \in [-25^\circ,\, 25^\circ]$

(b) Only varying head orientation, $\left(\theta^h,\, \phi^h\right) \in [-20^\circ,\, 20^\circ]$

Figure 2: Rotation Experiment

for head pose and gaze direction from person 'b' features and decoder training pass is done using this synthetically generated data. For the training, authors used various loss functions:

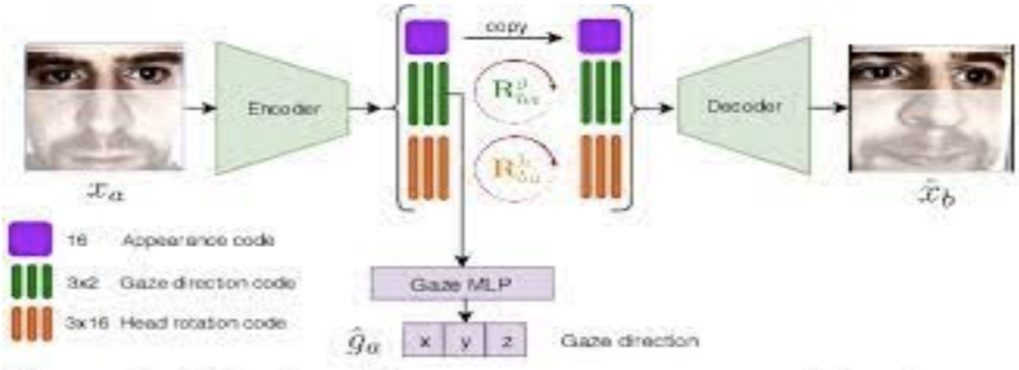


Figure 3: DT-ED Training

1. **Reconstruction Loss**: To guide learning of the encoding-decoding process, we apply a simple L_1 reconstruction loss. Given an input image x_b and reconstructed $\hat{x_b}$ obtained by decoding the rotated embeddings $\hat{z_b}$ of image x_a, the loss term is defined as: $L_recon(x_b, \hat{x_b}) = \frac{1}{|x_b|}\sum_{u \in x_b \& \hat{u} \in \hat{x_b}} |u - \hat{u}|$
2. **Embedding Consistency Loss**: To further disentangle, we compute $f(x) = (R_{g^{-1}z^g,x}$ is an image, which basically frontalizes the gaze angle, and then following loss function is applied between pair of intra-person images in batch of size B: $L_EC = \frac{1}{B}\sum_{i=1}^{B} max_{j\in\{1,2..B\},id(i)=id(j)} d(f(z_i^g, z_j^g))$

3. **Gaze Angle Loss:** This was introduce to further disentangle z^g to further predict gaze angle using a single MLP layer: $L_{gaze}(g, g') = arccos(g'.g/||g||||g'||)$

3 Dataset

The **MPIIGaze dataset**, which contains 213,659 images that we collected from 15 laptop users over several months in their daily life (see Fig. 2). To ensure frequent sampling during this time period, we opted for an experience sampling approach in which participants were regularly triggered to look at random on-screen positions on their laptop. This way, MPIIGaze not only offers an unprecedented realism in eye appearance and illumination variation but also in personal appearance – properties not available in any existing dataset. Methods for unconstrained gaze estimation have to handle significantly different 3D geometries between user, environment, and camera. To study the importance of such geometry information, we ground-truth annotated 37,667 images with six facial landmarks (eye andmouth corners) and pupil centres. These annotations make the dataset also interesting for closely related computer vision tasks, such as pupil detection. The full dataset including annotations is available **here.**

The following table describe the number of samples for each person and any specific qualities about dataset if any.

Person	**Image Count**	**Appearance Description**
Person 0	29961	Nothing special noticeable
Person 1	24143	Thick Rim Dark Colored Spectacles completely bordering eye region.
Person 2	28019	Nothing special noticeable
Person 3	35075	Nothing special noticeable
Person 4	16831	Light Colored spectacles which frequently interferes with eye region
Person 5	16577	Contains blurry images.
Person 6	18448	Nothing special noticeable
Person 7	15509	Light Colored spectacles which frequently interferes with eye region
Person 8	10701	Contains blurry images
Person 9	7995	Thick Rim Dark Colored Spectacles completely bordering eye region.
Person 10	2810	A significant percentage of them have spectacles.
Person 11	2982	Some images are blurry with very few of them with spectacles.
Person 12	1609	Nothing special noticeable
Person 13	1498	Nothing special noticeable
Person 14	1500	Hand and Hair interfere with eye region frequently.

Some of the examples can be shown in the following images:

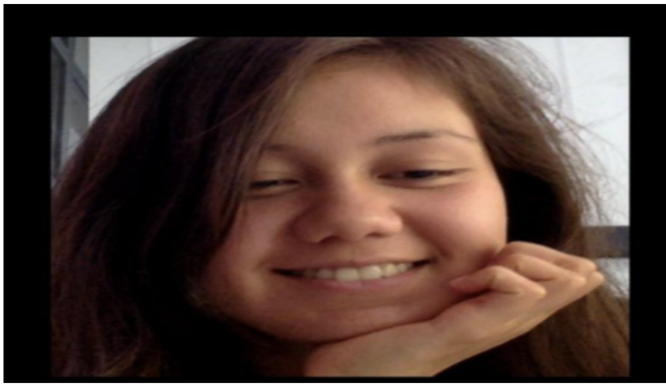


Figure 4: Person 14

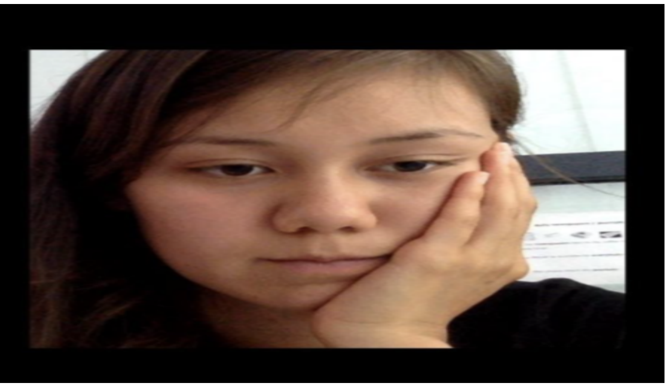


Figure 5: Person 14

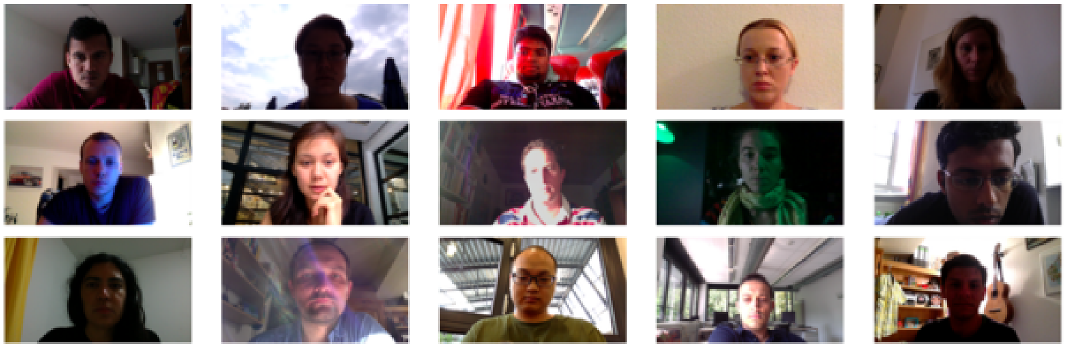


Figure 6: People in MPIIGaze Dataset

4 Experiments [1]

4.1 Basic Preprocessing:

Face Alignment and 3D Head Pose Estimation:While previous works assumed accurate head poses, we use a generic mean facial shape model F for the 3D pose estimation to evaluate the whole gaze estimation pipeline in a practical setting. The generic mean facial shape F is built as the averaged shape across all the participants, which could also be derived from any other 3D face models. We use the same definition of the face model and head coordinate system as [6]. The face model F consists of 3D positions of six facial landmarks (eye and mouth corners). As shown in Fig., the right-handed head coordinate system is defined according to the triangle connecting three midpoints of the eyes and mouth. The x-axis is defined as the line connecting midpoints of the two eyes in the direction from the right eye to the left eye, and the y-axis is defined to be perpendicular to the x-axis inside the triangle plane in the direction from the eye to the mouth. The z-axis is hence perpendicular to the triangle, and pointing backwards from the face. Obtaining the 3D rotation matrix Rr and translation vector tr of the face model from the detected 2D facial landmarks p is a classical Perspectiven-Point, problem which is estimating the 3D pose of an object given its 3D model and the corresponding 2D projections in the image.

We fit F to detected facial landmarks by estimating the initial solution using the EPnP algorithm [65] and further refine the pose by minimising the Levenberg-Marquardt distance for the eye-patch images. While for full face images we first use PnPRansac algorithm and then apply an iteration of EPnP algorithm to refine it.

Eye Image Normalisation Given the head rotation matrix R_r and the eye position in the camera coordinate system $e_r = t_r + e_h$ where e_h is the position of the midpoint of the two eye corners defined in the head coordinate system (Fig. 9 (a)), we need to compute the conversion matrix $M = SR$ for normalisation. As illustrated in Fig. 9 (b), R is the inverse of the rotation matrix that rotates the camera so that the the camera looks at e_r (i.e., the eye position is located along the z-axis of the rotated camera), the x-axis of the head coordinate system is perpendicular to the y-axis of the camera coordinate system. The scaling matrix $S = diag(1, 1, d_n/||e_r||)$ (Fig. 9 (c)) is then defined so that the eye position e_r is located at a distance d_n from the origin of the scaled camera coordinate system.

we denote the original camera projection matrix obtained from camera calibration as C_r and the normalised camera projection matrix as C_n, the same conversion can be applied to the original image pixels via perspective warping using the image transformation matrix $W = C_n M C_r^1$ (Fig. 9 (d)). $C_n = [f_x, 0, c_x; 0, f_y, c_y; 0, 0, 1]$, where f and c indicate the focal length and principal point of the normalised camera, which are arbitrary parameters of the normalised space. a head rotation matrix $R_n = MR_r$, and a gaze angle vector $g_n = Mg_r$ in the normalised space. g_r is the 3D gaze vector originating from e_r in the original camera coordinate system. The normalised head rotation matrix R_n is then converted to a three-dimensional rotation angle vector h_n. Since rotation around the z-axis is always zero after normalisation, h_n can be represented as a two-dimensional rotation vector (horizontal and vertical orientations) h. g_n is also represented as a two-dimensional rotation vector g assuming

[1] Code for both the models can be found at `https://github.com/visirion07/gaze_estimation`

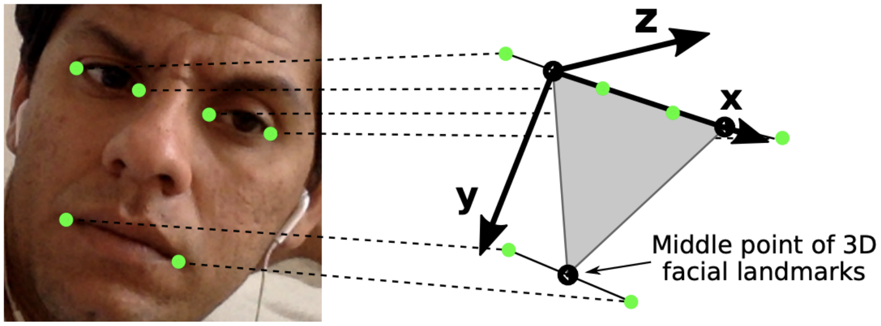


Figure 7: Definition of the head coordinate system defined based on the triangle connecting three midpoints of the eyes and mouth. The x-axis goes through the midpoints of both while the y-axis is perpendicular to the x-axis inside the triangle plane. The z-axis is perpendicular to this triangle plane

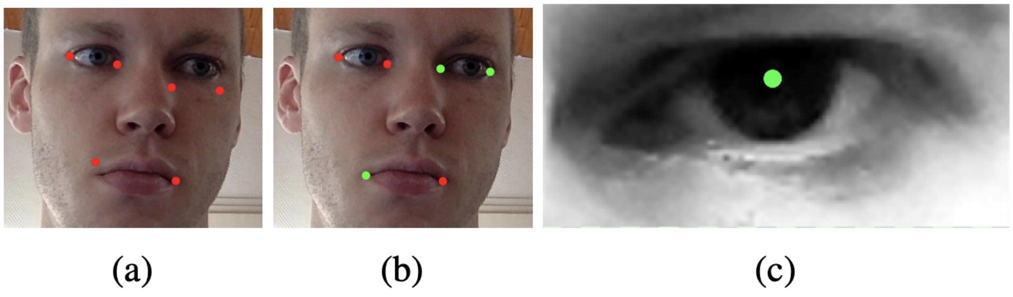


Figure 8: Annotations point of images

a unit length. Note that e_r is critically important to the calculation of perspective transformation matrix as well as g_n.

4.2 DT-ED and MLP

Preprocessing: Since in this model we only use full face images we first use **cv2.solvePnPRansac** method and then refine the results using **cv2.solvePnP** method for calculation of head pose vectors. Here since we use full face model we choose reference point for e_r as the mid point of right corner of left eye and left corner of right eye.
In camera normalization parameters that we choose $f_x = f_y = 1300$, $d_norm = 600mm$ and the resolution of normalized image to be $(256, 64)$.
Model: We use pretrained DT-ED model trained on GazeCapture Dataset. We then train an MLP network over the outputs from DT-ED model.

We train the model using the Gaze-Angle Loss itself which was also used in the training of DT-ED models.
Model Results:

We achieved a mean angle of about 5.6 degrees according to leave-one-out setting for the whole dataset. For each person error is shown in the following graph Figure 11.
Analysis

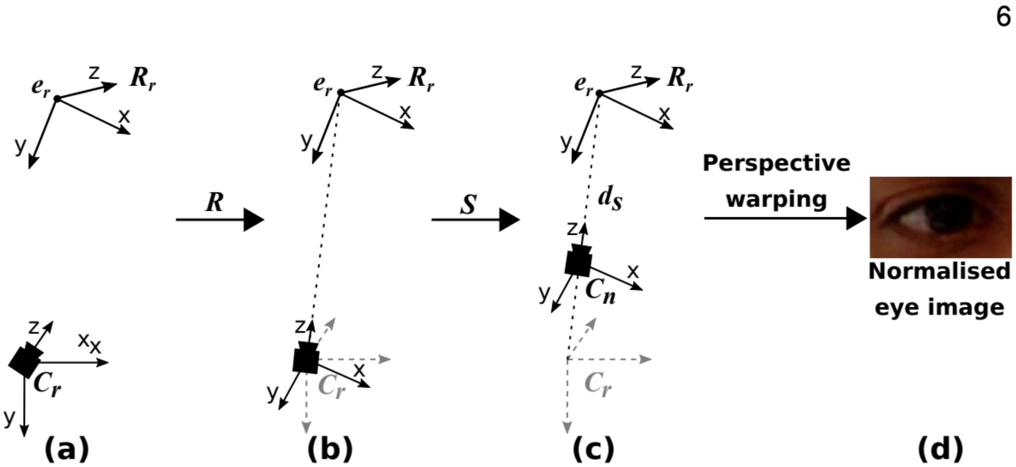


Figure 9: : Procedure for eye image normalisation. (a) Starting from the head pose coordinate system centred at one of the eye centres er (top) and the camera coordinate system (bottom); (b) the camera coordinate system is rotated with R; (c) the head pose coordinate system is scaled with matrix S; (d) the normalised eye image is cropped from the input image by the image transformation matrix corresponding to these rotations and scaling.

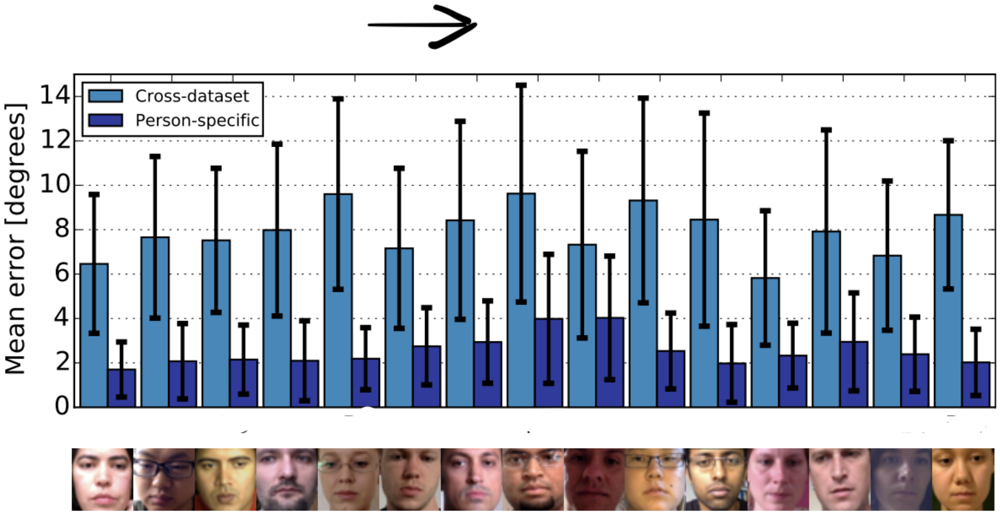


Figure 10: Gaze-Angle Error in Model used in MPIIGaze Paper

Disentanglement Analysis: The Faze model aims at disentangling head pose, gaze direction and personal-appearance based features from the image input. For encouraging disentanglement various loss functions and a different training style was introduced in the Face Gaze paper to train DT-ED model. To further analyze this disentanglement I performed the following experiments

- As a first check we only used z^h or the head pose latent features only to model the gaze direction. As expected, we were not able to properly model using those inputs and outputs using various models like linear regression and Neural Networks. Among the different models, Neural Networks gave the best result and we got a mean test error of about 65 degrees.
- As a second check we tried to model the gaze direction using only z^a or the appearance based

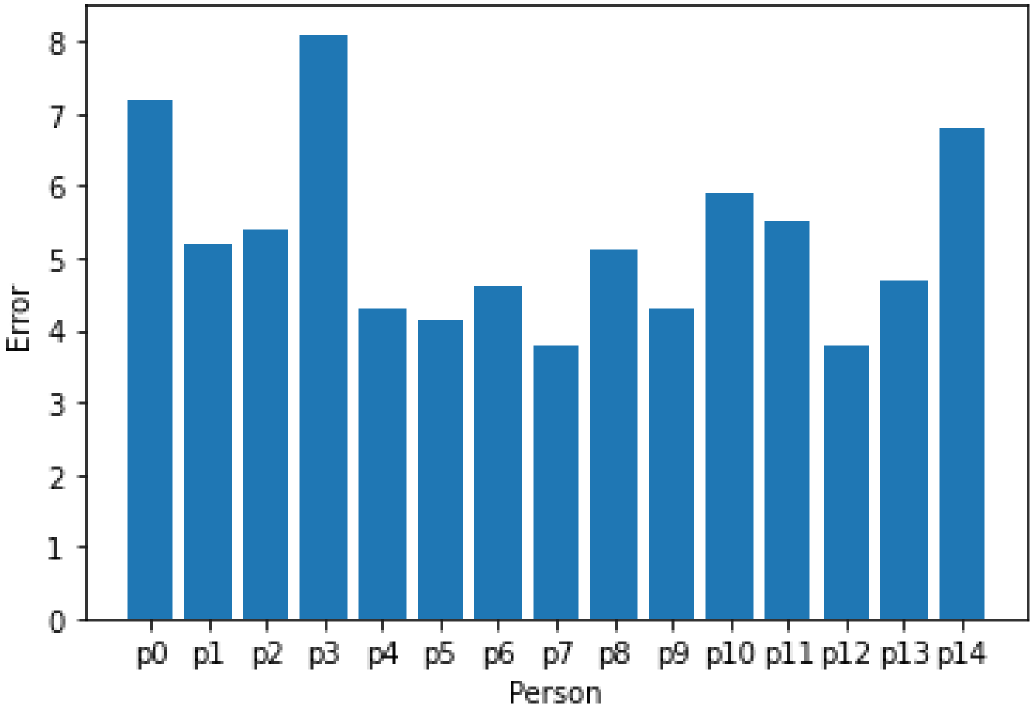


Figure 11: FaceGaze on entire MPIIGaze Dataset

latent features. For this task also we tried modelling using different models. Among the different models, Neural Networks gave the best result and we got a mean test error of about 43 degrees.

Inference: We can clearly make out the conclusion that these two features are quite unrelated to the gaze direction.

We got a shocking result, as we tried to model head pose using z^h, the latent representation for head pose information but were not able to model it properly using various model we tried. Again, neural network gave the best result with mean error of about 73 degrees.

Key points

1. We saw a significant decrements in the people wearing spectacles(1, 4, 7, 9) from a mean error of 8.9 degrees to about 5.53 degrees.
2. We again see that the problem with **Person 14** also reflects in this model. The issue is appearance of nuances in the eye region which is one of the most important region for gaze estimation.
3. For this observation we need to look in to the plots of gaze angle.
   - **Person 0**: About 13k of 29k images in the dataset of Person 0 had a pitch greater than +3 degrees. But in the remaining training set there were only 202 such images. This shows a lack of training data in that range is primarily the reason behind a large gaze angle in the case of person 0. When model is tested on the other part of the dataset error a significant decrement in gaze angle estimation is observed.
   - **Person 3**: About 25k of 35k images in the dataset of Person 3 had a pitch less than -8 degrees. But in the remaining training set there were only about a 1000 of such images. This shows a lack of training data in that range is primarily the reason behind a large gaze angle in the case of person 3.

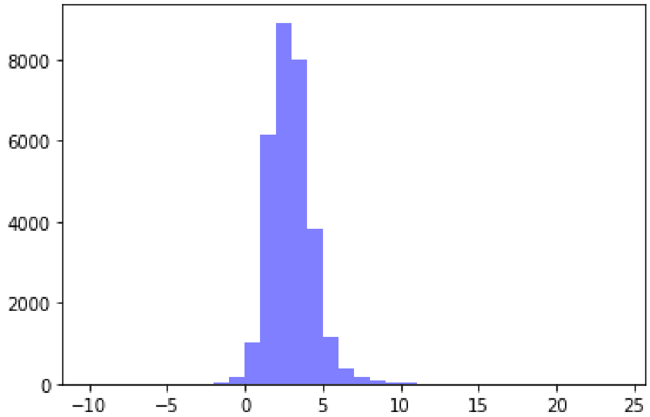


Figure 12: Person 0 pitch angle of Gaze Direction

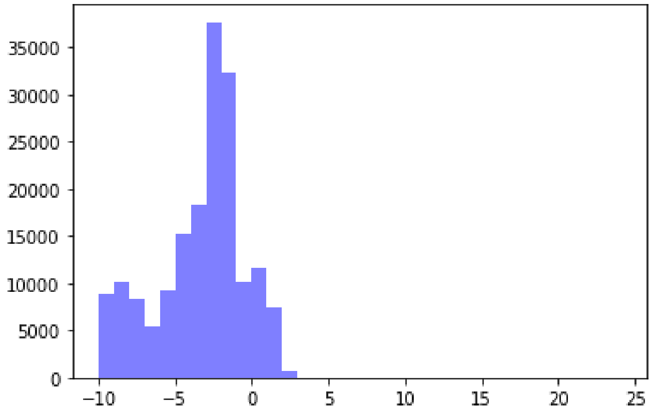


Figure 13: Everyone but Person 0 pitch angle of Gaze Direction

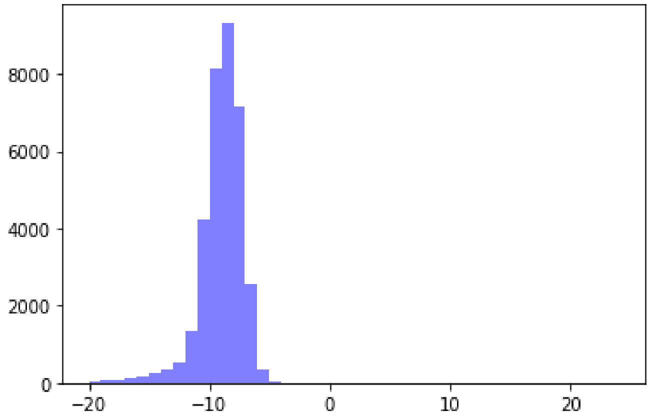


Figure 14: Person 3 pitch angle of Gaze Direction

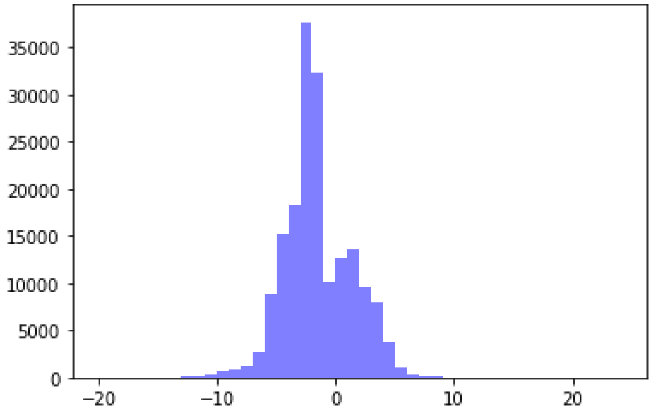


Figure 15: Everyone but Person 3 pitch angle of Gaze Direction

4.3 Model 2: ResNET-Preact Model for gaze-estimation

Preprocessing: Since here we only use the eye-patches we here use mean of the 2 points of each eye for the generation of the normalized image.

In camera normalization parameters we choose $f_x = f_y = 960$, $d_n orm = 600mm$ and the resolution of normalized image to be $(60, 36)$.

Model: This model is fairly described and trained **here.**. We tried a re implementation of the model on the eye-patches obtained from the MPIIGaze dataset.

Result: We achieved a mean angle of about 5.73 degrees according to leave-one-out setting for the whole dataset. For each person error is shown in the following graph:

Analysis

- **People 4 and 7** have specs in between the eye-region which is primarily the reason for large error.
- **Person 14** has a number of images in which hand and hair appears in the eyepatches and which appears to be the reason for a large error.
- **People 5 and 8** have blurry images: Results improved by augmenting training data with blurry images using cv2.blur() by 0.5 and 0.4 degrees.(Only current results shown).

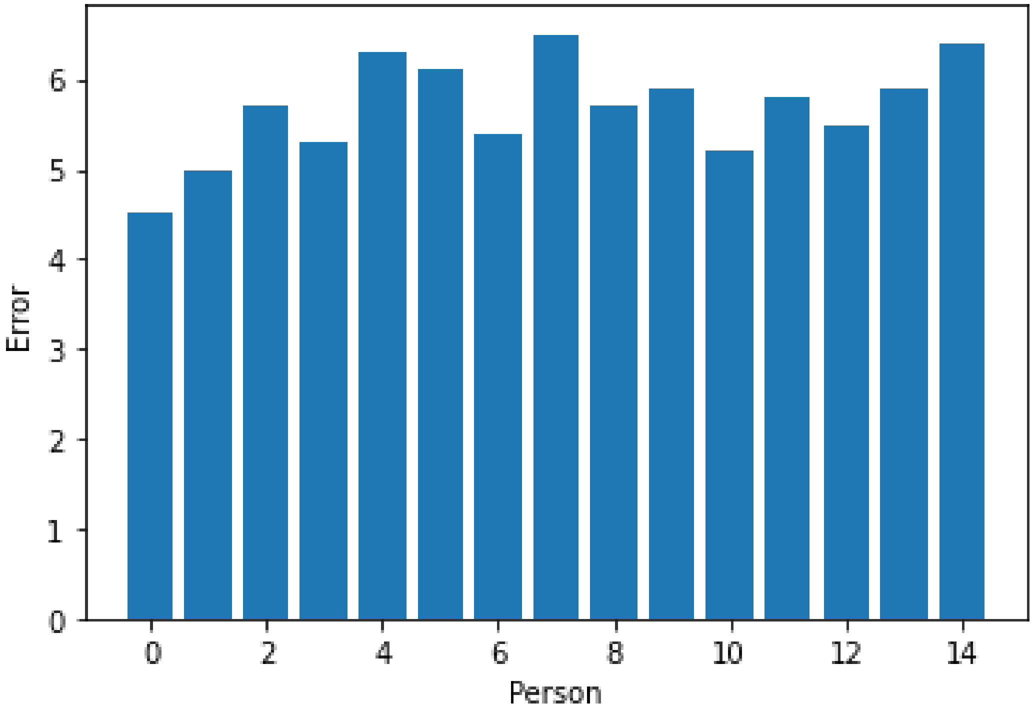


Figure 16: FaceGaze on entire MPIIGaze Dataset

4.4 Model 3: 'Ensemble' Model

Preprocessing: Many state-of-the-art algorithms on MPIIGaze dataset use ensemble models to acheive high result. We also tried to combine the above 2 models.
The problem is due to difference of e_r which makes the target vector g_n different for both the models. To overcome this problem we use vector algebra to match the targets.
The model we use, is not exactly an ensemble model. We just use the z^g or the gaze latent representation from the second model and use them as parametres for the model.

Let M matrix, g_o and g_t be the 'M' matrix as described in basic processing above, the corresponding e_r and the gaze target g_t. We want to change target g'_t from origin g'_o and with 'M' matrix M'.

The following eqution is used to transform: $g_{tnew} = M(||M'||||(M'^{-1})g'_t + g'_o - g_o)$. This vector now points to same target. Converting into unit norm vector we divide by its norm.
Model and Results: After processing as above we use the information from the 2nd model and then resnet is trained with additional features. We achieved a mean error of about 4.9 degrees with individual error as shown:

5 Conclusion and Future Work

In conclusion we have tried to model the gaze estimation from an image using Face Gaze and Resnet Models. We also got way to combine models which use eyepatches and full face since they can provide a significant of features which might not be possible from other kind of models.

In future, we can work on the following:

- Work on generating images with given gaze direction of a given person and with a given head pose. This can hugely solve the issues we saw with Person 0 and 3 in Model 1.

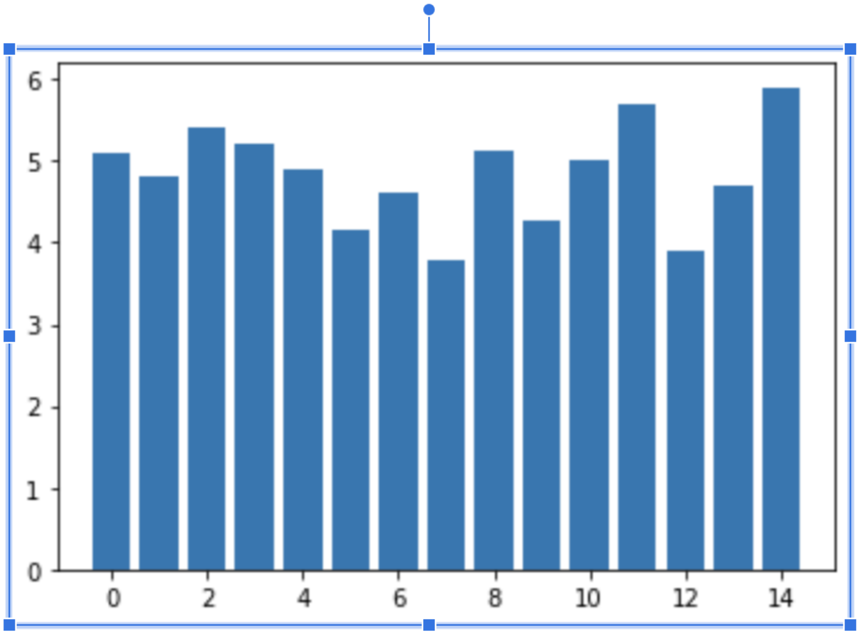


Figure 17: FaceGaze on entire MPIIGaze Dataset

Also we want to check this DT-ED model only if it can generate images. Then, we can just augment it with our current dataset.

- Apart from cv2.blur(), various other augmentation can be done. As we can see in our MPIIGaze dataset mean grayscale intensity varies quite a lot. We can address this issue by augmenting the dataset with images in the same intensity range.
- A big issue still after disentangling the appearance based features in DT-ED model are the inter-subject anatomical differences which can only be learnt only after seeing a few examples of the particular person. Therefore, we have to resort to some Few-shot methods in the future.

References

[1] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, “Explaining explanations: An overview of interpretability of machine learning,” in *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 80–89, 2018.