This CVPR workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

EFE: End-to-end Frame-to-Gaze Estimation

Haldun Balim[1] Seonwook Park[2*] Xi Wang[1*] Xucong Zhang[3*] Otmar Hilliges[1]
[1]Department of Computer Science, ETH Zürich [2]Lunit Inc.
[3]Computer Vision Lab, Delft University of Technology

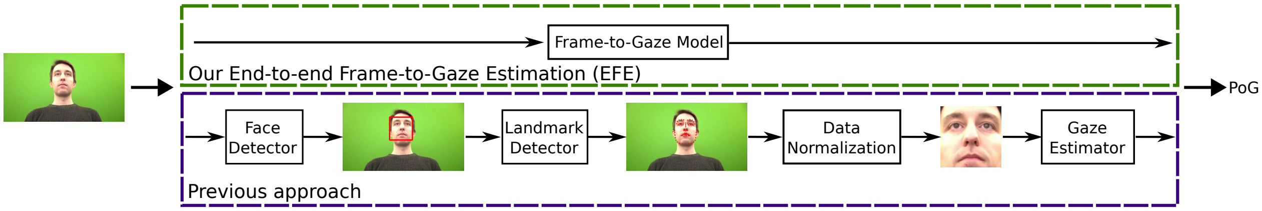

Figure 1. **Comparison of the proposed EFE method against conventional gaze estimation methods.** Our **E**nd-to-end **F**rame-to-Gaze **E**stimation approach (EFE) is trained to predict eye gaze directly from the input camera frame. In contrast, most existing approaches rely on additional pre-processing modules. For example, face and facial landmark detection methods are used in "data normalization" to obtain eye or face patches, which are then used as inputs to gaze estimation models. Our approach, EFE, demonstrates that it is possible to skip those steps while maintaining or improving performance.

Abstract

Despite the recent development of learning-based gaze estimation methods, most methods require one or more eye or face region crops as inputs and produce a gaze direction vector as output. Cropping results in a higher resolution in the eye regions and having fewer confounding factors (such as clothing and hair) is believed to benefit the final model performance. However, this eye/face patch cropping process is expensive, erroneous, and implementation-specific for different methods. In this paper, we propose a frame-to-gaze network that directly predicts both 3D gaze origin and 3D gaze direction from the raw frame out of the camera without any face or eye cropping. Our method demonstrates that direct gaze regression from the raw downscaled frame, from FHD/HD to VGA/HVGA resolution, is possible despite the challenges of having very few pixels in the eye region. The proposed method achieves comparable results to state-of-the-art methods in Point-of-Gaze (PoG) estimation on three public gaze datasets: GazeCapture, MPIIFaceGaze, and EVE, and generalizes well to extreme camera view changes.

1. Introduction

Remote webcam-based gaze estimation is a well-studied problem setting where images from a single user-facing and remotely-placed camera are used to estimate the gaze of a user. Effective solutions to this problem can enable novel applications in gaze-contingent human-computer interaction [3, 21], adaptive user interfaces [8, 14], and crowd-sourced attention studies [23]. Unlike infrared-light devices [45], webcam-based gaze estimation can allow for large operating distances [41]. With the introduction of large in-the-wild datasets [17, 42], many learning-based convolutional neural network-based (CNN) approaches [24, 34, 38, 42] have been proposed, enabling gaze estimation from a single front-facing camera.

Learning-based remote gaze estimation methods typically take small cropped patches as input to predict the gaze direction. These inputs as well as gaze origin must be generated with pre-defined processes according to the facial landmarks. Specifically, inputs to these methods are either simply cropped images [13, 17] or image patches yielded via a process known as "data normalization" [24, 29, 40]. Simple cropping methods usually simply crop the eye or face according to the facial landmarks. Since it is cropping directly on the 2D image without consideration of the 3D head pose, it could result in different sizes and image ratios in the case of large head rotations. This can introduce unnecessary appearance variations for gaze estimator training, reducing performance [40]. With the simple cropped face and eye images, recent gaze estimators can perform cross-person gaze estimation by leveraging a large amount of training data from multiple subjects [17]. To be able to

[*] These authors contributed equally to this work.

directly regress to on-screen Point-of-Gaze (PoG) without gaze origin and gaze direction predictions, these methods have to assume that the camera plane is coplanar to the screen plane [13, 17]. However, such a coplanarity assumption cannot be easily held for many application scenarios.

To eliminate the coplanar assumption, we could first calculate the gaze origin and then take the face/eye crop to predict the gaze direction as a two-step approach. To obtain the gaze origin and face/eye crop, data normalization is proposed as a pre-processing step [29, 40]. As shown in the bottom of Fig. 1, it crops the eye/face patch out of the input camera frame according to the facial landmarks and estimates a 3D head pose by fitting a generic 3D face model, which is further used to yield the 3D gaze origin. The gaze estimation method then only needs to output the gaze direction in the camera coordinate system. By composing the predicted gaze direction with the gaze origin acquired at the data normalization step, the gaze ray can be constructed. Note there is no joint optimization of gaze origin and gaze direction since the gaze origin is estimated separately from facial landmarks and explicit head pose estimation. The ill-posed problem of 3D head translation estimation from a 2D image could introduce extra error in the depth estimation of the gaze origin. Furthermore, the processes of data normalization are often implemented as an expensive offline procedure (see [24, 25] in particular).

Skipping the aforementioned pre-processing steps and directly taking the raw frame as input is a highly challenging task and has rarely been investigated in the literature due to small eye region and gaze original estimation. Since CNNs typically accept small images around 224×224 to 512×512 pixels, we have to resize the raw input frame to be much smaller from HD (720p) or Full HD (1080p) resolution. It results in a much smaller region of interest (face/eye) than the cropped patches. Without the facial landmark, the 3D location of the gaze origin needs to be accurately estimated from the monocular image which has not been investigated in previous works.

In this paper, we demonstrate that it is possible to train an "**E**nd-to-end **F**rame-to-Gaze **E**stimation" (EFE) method without making the aforementioned coplanar assumption. As shown in the top of Fig. 1, our approach avoids the need for expensive "data normalization" and directly regresses a 6D gaze ray (3D origin and 3D direction) from the camera frame without cropping the face/eye, allowing the trained method to adapt to new camera-screen geometries. The PoG is calculated by the 3D origin and 3D gaze direction. We observe that the gaze origin can be predicted accurately with a fully-convolutional U-Net-like architecture (shown in Fig. 2). The network predicts a 2D heatmap ($\mathbf{h}$), which captures the 2D location of the gaze origin inside of the frame, and a depth map ($\mathbf{d}$) that captures the distance to the gaze origin in the third dimension. We take the bottleneck features from the U-Net-like architecture and pass them through a multi-layer perception (MLP) to predict the 3D gaze direction. With camera intrinsic and extrinsic parameters, we intersect the gaze ray with the known screen plane in a differentiable manner to yield PoG. This architecture can be trained end-to-end and we evaluate it on three existing large datasets: EVE [24], GazeCapture [17], and MPIIFaceGaze [44]. We show that it achieves comparable performance with competitive baselines using "data normalized" inputs on the EVE, GazeCapture and MPIIFaceGaze datasets.

2. Related Works

2.1. Remote Gaze Estimation from RGB

Remote gaze estimation from RGB is a setting where a single RGB camera faces the user and no additional instruments such as IR light sources are used to make the gaze estimation problem more tractable. Due to the challenges imposed by this setting, even early methods for gaze estimation tend to apply machine learning techniques [1, 20, 27, 30] but are limited to tackling person-specific gaze estimation only. Recently released large-scale image datasets such as MPIIGaze [42] and GazeCapture [17] enable so-called cross-person (person-independent) gaze estimation, where models are evaluated on data from people that were unseen during training. Various CNN architectures have since been proposed to improve gaze estimation [6, 26, 33, 43], demonstrating their efficacy on additional in-the-wild [15], laboratory [9, 10, 24, 39], and synthetic [35, 36] datasets.

Most of these approaches take as input either cropped eye or face images [6, 9, 15, 17, 26] or so-called "data normalized" images [16, 25, 39, 43] and do not directly learn to predict gaze from the full camera frame. The GazeOnce method utilizes multi-task learning to output face existence, face location, facial landmarks, and 3D gaze direction with the raw frame as the input [38], however, this method does not output the gaze origin nor allow for eventual PoG estimation. While few-shot adaptation approaches [18, 25, 41] still have the opportunity to adapt to systematic errors due to the user or technical setup, the performance of most cross-person methods will vary greatly depending on the pre-processing adopted during inference time. Our paper demonstrates that it is possible to learn the complex mapping between frames and 6D gaze rays by designing appropriate modules for the sub-tasks of origin and direction regression. As we learn to side-step complex pre-processing such as facial landmark detection or data normalization, the performance of our approach should not depend on technical setup related factors.

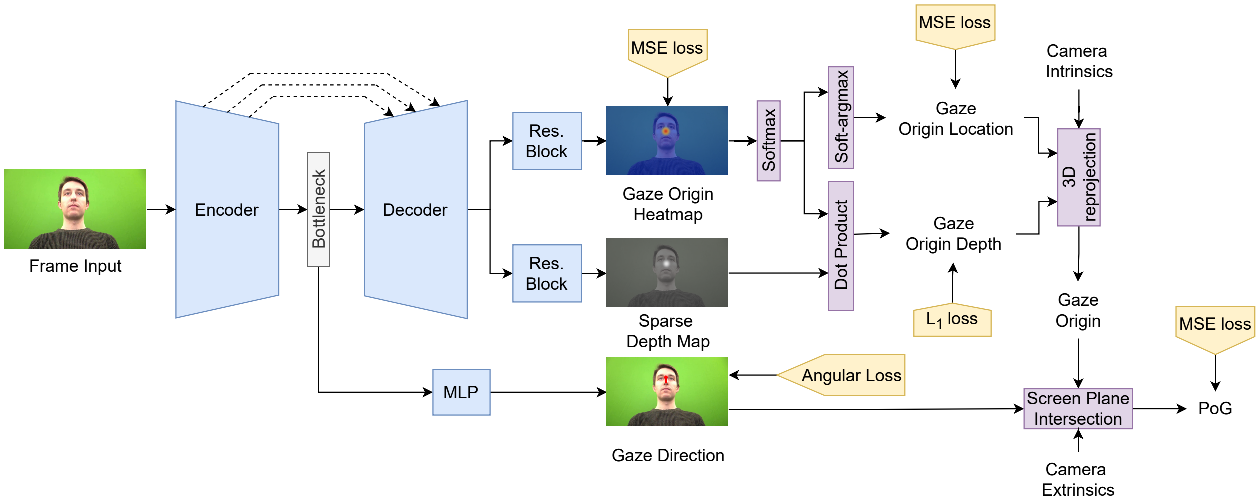

Figure 2. **The proposed end-to-end frame-to-gaze estimation architecture, EFE.** We propose a U-Net-like architecture where the output features are mapped to the 2D gaze origin location on the image and a sparse depth map, which are combined to produce the 3D gaze origin. The 3D gaze direction is predicted with an MLP using the bottleneck features as input. The PoG is calculated using predicted gaze origin and direction, together with camera transformation matrices (that define camera-to-screen geometry).

2.2. Learning-based PoG Estimation

Of the learning-based gaze estimation methods that take RGB input, very few study the task of directly predicting the Point-of-Gaze (PoG). For example, [17] assumes that all PoGs are on the z-plane of the camera coordinate system and directly regress PoG in centimeters, while [13, 43] directly regresses PoG (in cm or mm) regardless of changes in camera-to-screen geometry (rotation, translation, and scaling) between dataset participants. Other methods for estimating PoG take advantage of the displayed stimuli and evaluate their saliency [28] or visual features [24] to correct the errors in estimated PoG. In [24], estimated gaze direction is more explicitly composed with known pseudo-ground-truth gaze origin (which is produced as a result of "data normalization") and this is combined with known camera-to-screen geometry to produce PoG. We follow this explicit geometric decomposition in our work but propose to predict gaze origin via a neural network, removing the need for "data normalization" at both training and inference times.

2.3. End-to-end Learning in Gaze Estimation

In the field of gaze estimation, only very few works extend their methods beyond the eye/face patch input or gaze direction output. [17, 43] propose architectures that take eye/face crops as input and yield PoG. [38] proposes a multi-person gaze direction estimation method that begins from a large input image, but their method does not predict gaze origin and thus PoG cannot be directly computed. Our method, EFE, begins from camera frames and ends with PoG and can be trained end-to-end. As we learn the relation between camera output and final quantity of interest (PoG), we could consider our method to be truly end-to-end.

3. Method

Our aim is to learn a model that can estimate a 6D gaze ray, including a gaze origin and a gaze direction, directly from a camera frame in an end-to-end fashion. We call our method End-to-end Frame-to-gaze Estimation (EFE). To achieve this goal, we decompose the task into two by predicting a gaze origin and a corresponding gaze direction. We denote the input RGB image as $\mathbf{X} \in \mathbb{R}^{3\times H\times W}$ pixels, and define the 6D gaze ray as consisting of the gaze origin $\mathbf{o} \in \mathbb{R}^3$ and the gaze direction $\mathbf{r} \in \mathbb{R}^3$. Given camera parameters represented by the intrinsic camera matrix $\mathbf{K}$ and the extrinsic matrix $\mathbf{T}$, we compute the PoG $\mathbf{p} \in \mathbb{R}^3$ on screen. An overview of EFE is shown in Fig. 2.

3.1. Predicting Gaze Origin

A straightforward approach to predict the gaze origin $\mathbf{o} \in \mathbb{R}^3$ is directly regressing the 3D location coordinates. However, estimating depth through a monocular RGB image is ill-posed and challenging. Instead of regressing the 3D location, it is well-known that predicting the heatmaps could generate a better estimation, as shown in the areas of facial landmark localization [4] and human pose estimation [32]. Motivated by that, we propose a U-Net-like architecture to predict a 2D gaze origin heatmap $\mathbf{h} \in \mathbb{R}^{H\times W}$ and a sparse depth map $\mathbf{d} \in \mathbb{R}^{H\times W}$ (see Fig. 2 for a visual example).

The positions of gaze origins are not well defined. Most datasets' ground truth labels are created through facial landmark detection and data normalization [29]. Therefore, learning a distribution of the prediction is more appropriate than predicting a single point since the ground truth labels are likely to contain some degree of error. We use $\mathbf{h}$ to predict the 2D location of gaze origin $g \in \mathbb{R}^2$ on the camera frame. We use the mean squared error loss $\mathcal{L}_{\textbf{heatmap}}$ for heatmap prediction and the final $\mathbf{h}$ is obtained after soft-argmax operation [5, 37].

$$\mathcal{L}_{\textbf{heatmap}} = \frac{1}{n}\sum_{i=1}^{n}\|\mathbf{h} - \hat{\mathbf{h}}\|_2^2, \tag{1}$$

where $n = H \times W$, and $\hat{\mathbf{h}}$ is the ground truth heatmap generated by drawing a 2D Gaussian centered at the gaze origin. The predicted g location is similarly supervised by a mean squared error loss $\mathcal{L}_g$,

$$\mathcal{L}_{\mathbf{g}} = \|\mathbf{g} - \hat{\mathbf{g}}\|_2^2, \tag{2}$$

where the $\hat{\mathbf{g}}$ is the ground truth 2D gaze location on the camera frame. The dot product $\mathbf{h} \cdot \mathbf{d}$ is used to predict the gaze depth $z \in \mathbb{R}$. Note that we do not use any approximated depth map for supervision and that the $\mathbf{d}$ is solely learned from $\mathcal{L}_d$ which is defined as

$$\mathcal{L}_d = \|\mathbf{z} - \hat{\mathbf{z}}\|_1, \tag{3}$$

where $\hat{\mathbf{z}}$ is the ground truth depth value. Our experiments show that this approach outperforms a baseline of regressing the 3D location in the coordinates. The gaze origin $\mathbf{o} \in \mathbb{R}^3$ is then calculated by transforming the image coordinates to world coordinates utilizing the camera intrinsic matrix $\mathbf{K}$. Note that $\mathbf{o}_z = \mathbf{z}$.

3.2. Predicting Gaze Direction

Since predicting the gaze direction is mapping from image space to the 3D direction, it is not necessary to use heatmap for the estimation. As shown in Fig. 2, we use the bottleneck features in the middle of the U-Net-like architecture for the prediction of gaze direction $\mathbf{r} \in \mathbb{R}^3$ supervised by the angular loss $\mathcal{L}_{\mathbf{r}}$. The gaze origin and gaze direction prediction share the encoder that could benefit both tasks. The gaze vectors are predicted as Euler angles in spherical coordinates and transformed to 3-dimensional unit vector $\mathbf{r}$. Given the predicted gaze vector $\mathbf{r}$ and ground-truth gaze vector $\hat{\mathbf{r}}$ the angular loss is calculated as

$$\mathcal{L}_{\mathbf{r}} = \arccos\left(\frac{\hat{\mathbf{r}} \cdot \mathbf{r}}{\|\hat{\mathbf{r}}\|\|\mathbf{r}\|}\right). \tag{4}$$

3.3. Computing PoG

The PoG is defined by intersecting the 6D gaze ray (composed of origin and direction) with a pre-defined screen plane. The screen plane is defined based on the physical screen with its z-axis pointing outwards and towards the user. The estimated PoG should have a z-coordinate of zero in the screen coordinate system, i.e. $\mathbf{p}_z = 0$. Using the gaze origin $\mathbf{o}$ and gaze direction $\mathbf{r}$, we can obtain the distance to screen frame λ. Denoting screen frame normal as $\mathbf{n}_\mathrm{s}$ and a sample point on the screen plane as $\mathbf{a}_\mathrm{s}$ (e.g. the origin point [0, 0, 0]), we can calculate λ as follows:

$$\lambda = \frac{\mathbf{r} \cdot \mathbf{n}_s}{(\mathbf{a}_s - \mathbf{o}) \cdot \mathbf{n}_s}. \tag{5}$$

After the distance to the screen frame is calculated, we can find the intersection of the line of sight with the screen plane to compute the PoG $\mathbf{p}$ as follows:

$$\mathbf{p} = \mathbf{o} + \lambda\mathbf{r}. \tag{6}$$

During training, we use mean squared error to supervise PoG estimation:

$$\mathcal{L}_{\text{PoG}} = \|\mathbf{p} - \hat{\mathbf{p}}\|_2^2 \tag{7}$$

where the $\hat{\mathbf{p}}$ is the ground truth PoG.

3.4. End-to-end Frame-to-Gaze Estimation (EFE)

As shown in Fig. 2, the overall EFE takes a camera image (frame) as input and yields a gaze origin heatmap $\mathbf{h}$ and a sparse depth map $\mathbf{d}$ through two separate residual blocks. Using the intrinsic camera matrix, the origin $\mathbf{o}$ in 3D space is calculated. The gaze direction $\mathbf{r}$ is predicted using the bottleneck features. Lastly, using $\mathbf{o}$ and $\mathbf{d}$, a 6D gaze ray is formed and intersected with a given screen plane to compute PoG. The complete loss is

$$\mathcal{L}_{\textbf{Total}} = \lambda_g\mathcal{L}_{\mathbf{g}} + \lambda_h\mathcal{L}_{\textbf{heatmap}} + \lambda_d\mathcal{L}_{\mathbf{d}} + \lambda_r\mathcal{L}_{\mathbf{r}} + \lambda_{PoG}\mathcal{L}_{\textbf{PoG}}, \tag{8}$$

where the first three losses $\mathcal{L}_{\mathbf{g}}$, $\mathcal{L}_{\textbf{heatmap}}$ and $\mathcal{L}_{\mathbf{d}}$ are for gaze origin estimation.

4. Experiment

To demonstrate the effectiveness of EFE, we first compare it with three end-to-end gaze estimation baselines to show the advantage of our proposed method. We then conduct evaluations on three datasets, EVE [24], GazeCapture [17], and MPIIFaceGaze [43] to show that learning the mapping from frame to gaze is possible and that EFE does it in a competitive manner when compared to existing state-of-the-art methods.

4.1. Datasets and Preprocessing

EVE [24]. The EVE dataset is proposed for the end-to-end gaze estimation task. It consists of continuous videos collected from four cameras and uses a Tobii Pro Spectrum

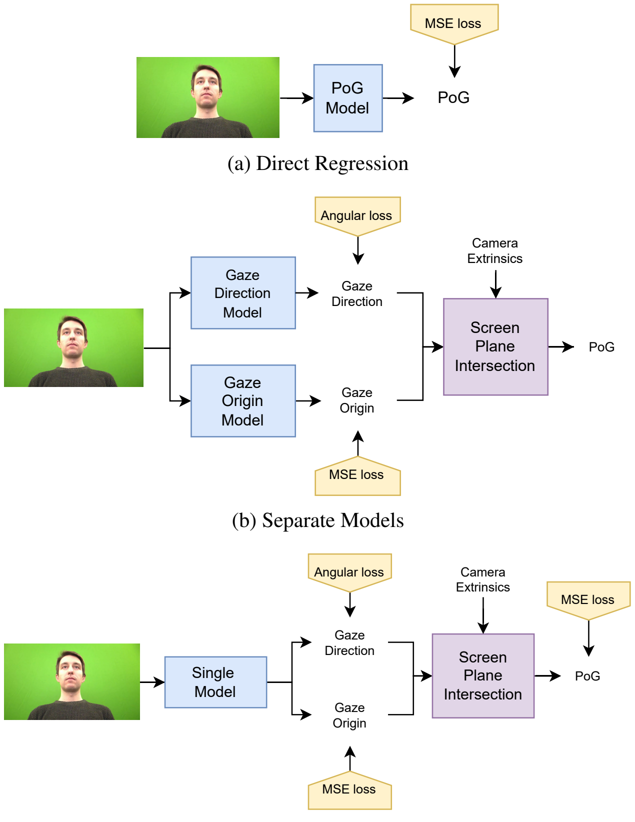

Figure 3. **Three end-to-end frame-to-gaze baseline models.** (a) Direct Regression: A model that directly estimates PoG, (b) Separate Models: Two separate models estimate gaze origin and gaze direction independently, and (c) Joint Prediction: A single model that jointly predicts both gaze origin and direction.

(150Hz) eye tracker to provide the ground truth gaze labels. Although the EVE dataset is proposed as a video dataset, we create an image dataset using its frames with a subsampling rate of 0.6 for ease of experimentation (only applied to the training set). The frames are resized to 480×270 pixels in size before using as input to the evaluated models.

GazeCapture [17]. The GazeCapture dataset is collected through crowdsourcing with phones and tablets. It consists of over 1450 people and almost $2.5M$ frames. The dataset provides the input raw frames and the PoG with respect to the camera while assuming that the camera and screen are coplanar.

MPIIFaceGaze [44]. The MPIIFaceGaze dataset is collected from 15 subjects with their laptop under natural head movements and diverse lighting conditions. There are 3000 face images for each subject. It provides both raw camera frames, the 3D gaze origins estimated by data normalization, the 3D gaze directions, and the 2D PoGs on the screen.

4.2. Implementation Details

Our U-Net-like architecture takes the EfficientNet-V2 small [31] architecture as the backbone. We observe that $\mathcal{L}_{\mathbf{PoG}}$ is significantly higher than the other losses in the early stages of training, as the model has not yet been able to estimate the gaze origin and direction with reasonable accuracy. Therefore, we do not optimize with respect to $\mathcal{L}_{\mathbf{PoG}}$ for the first two epochs of the training process and set $\lambda_{PoG} = 0$. We train EFE using the AdamW optimizer [19] for eight epochs unless otherwise mentioned, using a batch size of 32. An exponential learning rate decay of factor 0.9 is applied, beginning from a learning rate of 0.0003. The origin and PoG coordinates are standardized with train set metrics for each of the datasets. λ_g is set to 2 and the remaining loss weights are set to 1.

For the EVE dataset, we terminate the training after 8 epochs and set $\lambda_{PoG} = 0$ in the first 2 epochs. The inputs to EFE are 480×270 pixels in size resized from the original 1920×1080 pixels in the dataset.

For GazeCapture, we terminate the training after five epochs and set $\lambda_{PoG} = 0$ for the first epoch as it is a large dataset. The PoG prediction is truncated to the screen size similar to the original work using the device type and orientation information. The inputs to EFE are 512×512 raw camera frames created from the original dataset[*] by re-projecting using a common virtual camera with a focal length of 460 mm to consolidate images from diverse camera devices and orientations.

For MPIIFaceGaze, we terminate training after 15 epochs and set $\lambda_{PoG} = 0$ in the first three epochs. The inputs to EFE are 640×480 raw camera frame resized from 1280×720 pixels in the original dataset, and are created by re-projecting using a common virtual camera with a focal length of 550 mm.

4.3. End-to-End Frame-to-Gaze Baseline Models

Given the raw frame from the camera, we consider three baseline models: *direction regression*, *separate models*, and *joint prediction* as shown in Fig. 3. The direction regression method directly estimates PoG without predicting any gaze origin or gaze direction. The separate models method estimates the gaze origin and direction independently through two identical models. The joint prediction method estimates gaze origin and direction via a shared convolutional neural network and two separate MLPs. Importantly, the joint prediction method is optimized with a PoG loss as well which should improve PoG estimation performance. The joint prediction baseline is the most similar one to EFE. However, the main difference between EFE and these baselines stems from the gaze origin prediction architecture. The heatmap-

[*] The GazeCapture dataset originally consists of images of size 640×480 and 480×640 due to the diverse mobile device orientations used.

Model	Heatmap pred.	Depth map pred.	Gaze Origin (mm)	Gaze Dir. (°)	PoG (px)
Direct Regression			-	-	143.83
Separate Models			16.18	3.63	141.35
Joint Prediction			20.14	3.73	143.39
EFE w/o depth map	✓		18.28	3.82	146.82
EFE (ours)	✓	✓	**16.07**	**3.53**	**133.73**

Table 1. **Comparison to the baseline models on the EVE dataset.** Accuracy of estimated gaze origin is reported in millimeters (mm), gaze direction in degree (°), and PoG in screen pixels (px). The first three baselines correspond to the three models shown in Fig. 3. Note that the input frames are the raw frames from the camera without any face and facial landmark detection.

based prediction of gaze origin allows EFE to model the uncertainty, and the prediction of origin depth through sparse depth maps allows this uncertainty to be propagated to the depth prediction.

We evaluate the three baselines and EFE on the EVE dataset since the dataset is specifically designed for the end-to-end gaze estimation task. Also, EVE uses a complex and offline data pre-processing scheme (including per-subject calibration of a morphable 3D face model) and thus the gaze origin labels acquired via data normalization are of high quality, making the dataset a particularly challenging benchmark for Frame-to-Gaze methods. We show in Tab. 1 that EFE outperforms all three baselines. The direct regression neither predicts gaze origin nor gaze direction and achieves poor performance compared to the other methods. The separate models method achieves better results than the other two baselines due to the larger model capacity. The joint prediction is similar to our architecture and achieves worse results than the separate models, which indicates that it cannot learn gaze origin and direction jointly in an effective manner. In contrast, EFE achieves the best performance compared with all three baselines.

Furthermore, we can see from Tab. 1 that performance degrades greatly when we do not use the sparse depth map estimation module, instead, using a MLP to predict $\mathbf{z}$ from bottleneck features. This demonstrates that predicting a heatmap for gaze origin regression combined with the learning of a sparse depth map for distance estimation is important for Frame-to-Gaze architectures.

4.4. Comparison with State-of-the-art

In this section, we compare EFE with state-of-the-art methods on three datasets, i.e. EVE, GazeCapture, and MPIIFaceGaze.

Comparison with SotA on EVE. For the EVE dataset, we list the performance of EyeNet reported in the original EVE paper [24]. EyeNet uses either left or right-eye images as input, and predicts gaze direction and PoG independently for each eye. Note that EyeNet uses ground-truth gaze origins acquired via data normalization, similar to most other state-of-the-art baseline methods. In addition, we train a model with the face images acquired from the data normalization procedure and denote it *FaceNet*. It uses the ground truth gaze origin acquired through data normalization and the model itself only outputs gaze direction. This is the most common and high-performing problem formulation in learning-based gaze estimation. As in EFE, we use the same EfficientNet-V2 small [31] as the backbone of FaceNet for a fair comparison.

Model	Inputs	Gaze Dir. (°)	PoG (px)
EyeNet (static) [24]	Right Eye	4.75	181.0
EyeNet (static) [24]	Left Eye	4.54	172.7
FaceNet	Face	**3.47**	134.10
EFE (ours)	Frame	3.53	**133.73**

Table 2. **Comparison with state-of-the-art methods on the EVE dataset.** Accuracy of estimated gaze direction is reported in degree (°), and PoG in screen pixels (px).

As shown in Tab. 2, EyeNet achieves the worst results even with the normalized eye images. By taking a normalized face image as input, FaceNet achieves better results, in line with the literature on face-based gaze estimation [17, 43]. The input to FaceNet is a normalized face image of size 256×256, while the input to EFE is a resized raw frame of size 480×270. Note that the effective face resolution is much smaller in the resized frame than in the normalized face image. Nonetheless, EFE has comparable performance to FaceNet. Thanks to end-to-end learning with a direct PoG loss, we find that EFE has a slightly better PoG estimate despite having worse performance on gaze direction. This could be attributed to the fact that the ground-truth gaze labels provided in EVE are themselves estimates and contain some degree of error. We expect higher accuracy of the provided ground truth PoG as it is measured by a desktop eye tracker, and a direct loss using this PoG ground-truth would result in better learning and consequent model performance. Indeed, our EFE performs best in terms of PoG prediction.

Model	Inputs	Phone PoG	Tablet PoG
iTracker [17]	Face&Eyes	2.04	3.32
iTracker (train aug) [17]	Face&Eyes	1.86	2.81
SAGE [12]	Eyes	1.78	2.72
TAT [11]	Face	1.77	2.66
AFF-Net [2]	Face&Eyes	1.62	**2.30**
EFE (ours)	Frame	**1.61**	2.48

Table 3. **Comparison with state-of-the-art methods on the GazeCapture dataset.** Accuracy of estimated PoG is reported in centimeters (cm).

Model	Input	Gaze Dir. (°)	PoG (mm)
Full-Face [43]	Face	4.8	42.0
FAR-Net* [7]	Face&Eyes	**4.3**	-
AFF-Net [2]	Face&Eyes	4.4	39.0
EFE (ours)	Frame	4.4	**38.9**

Table 4. **Comparison with state-of-the-art methods on the MPIIFaceGaze dataset.** Accuracy of estimated gaze direction is reported in degrees (°), and PoG in millimeters (mm).

Comparison with SotA on GazeCapture. We show the comparison between EFE and other state-of-the-art on the GazeCapture dataset in Tab. 3. The results of iTracker [17] are reported from the original paper and we list performances for both *phone* and *tablet* on the GazeCapture dataset. The iTracker method takes multiple inputs: the cropped face, left eye, right eye, and face occupancy grid. In contrast, EFE takes only the resized raw frame as input. From the table, we can see that EFE outperforms iTracker by a large margin. It shows that the proposed end-to-end method is better than using multiple cropped images for direct PoG regression.

In comparison with more advanced methods such as a method using knowledge distillation [11] and a method taking eye corner landmarks as input [12], our EFE still performs competitively, with lower PoG error on both *phone* and *tablet* subsets. Note that EFE only requires the full frame as input (albeit re-projected to a common set of camera parameters) while all other methods require multiple specifically prepared inputs such as cropped face image [11,17], cropped eye image [12,17], “face grid” image [17], and/or eye corner landmarks [12, 17]. EFE achieves comparable performance to the latest PoG estimation method AFF-Net [2], which not only takes the cropped face and eye patches as input but also incorporates complex feature fusion between these patches.

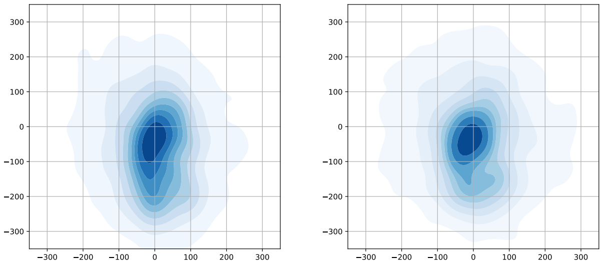

Figure 4. **2D histogram of PoG residuals on EVE.** PoG error distribution on the screen in pixels with the same predicted gaze direction but different gaze origins. **Left:** gaze origin is calculated by the data normalization. **Right:** gaze origin is predicted by EFE. The comparison shows that our end-to-end learning approach exhibits less bias in its PoG errors.

Comparison with SotA on MPIIFaceGaze. For the MPIIFaceGaze dataset, we compare EFE with state-of-the-art PoG estimation methods, Full-Face [43], FAR-Net* [7], and AFF-Net [2], under a 15-fold cross-validation evaluation scheme. The numbers are copied from the corresponding papers. As seen in Tab. 4, EFE outperforms the Full-Face approach and is comparable to the FAR-Net* approach and AFF-Net. Note that Full-Face, FAR-Net*, and AFF-Net use face images after data normalization as input as well as the ground truth gaze origin computed via data normalization, therefore, the model itself only outputs the gaze direction. Although the proposed EFE takes as input the raw frame, which has fewer effective pixels on the face region, it still achieves comparable results. Once again, the results show that the proposed EFE gaze estimation pipeline is effective even without the complex data normalization procedure.

4.5. Qualitative Analysis

In this section, we analyze our proposed method, EFE, in a qualitative manner.

Bias in PoG residuals. The majority of gaze direction estimation methods use gaze origin determined via the data normalization procedure as ground truth. Blindly relying on this gaze origin value can result in systematic errors. In contrast, EFE trains with a direct PoG loss which is more reliable and can correct the systematic errors caused by data normalization. We observe this in Fig. 4 where we find that EFE shows lower bias in terms of PoG residuals, in that its error distribution is more centered and isotropic.

Visualization of Predicted Depth Maps. The depth map predicted by EFE is supervised via a point-like loss ($\mathcal{L}_{\mathrm{d}}$), which only refers approximately to the face region of the visible user. Yet surprisingly, Fig. 5 shows that somewhat plausible depth maps can be predicted. More specifically, a rough notion of whether the user's face is further away or closer to the camera is captured. It is interesting that such weak supervision can still produce plausible dense outputs.

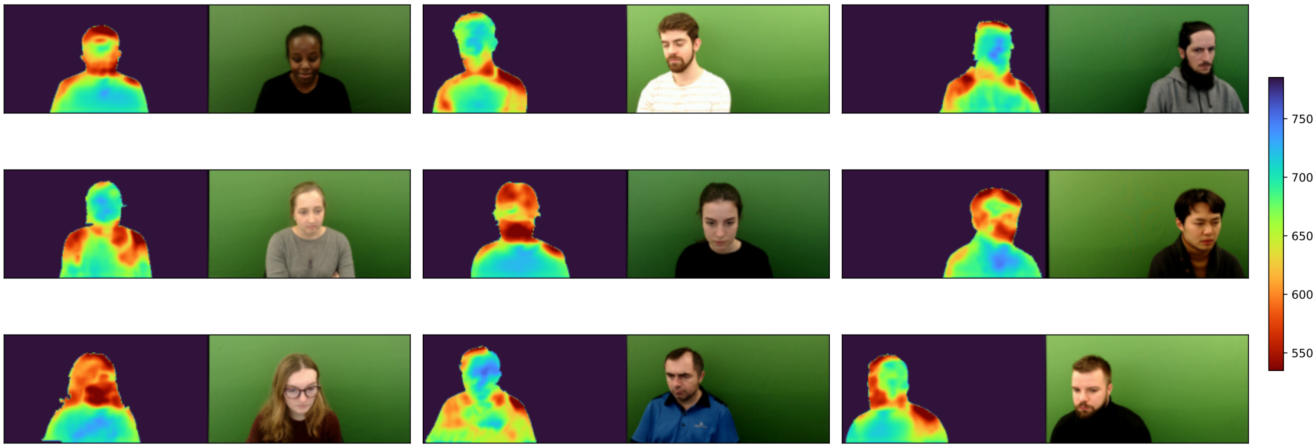

Figure 5. **Depth maps predicted by EFE.** The depth maps are colored using the turbo colormap [22] and the green background is subtracted for EVE to make the visualizations easier to interpret. Though not all depth values match real-world expectations, the model has an approximate notion of whether the head and torso are further away or closer to the camera.

4.6. Cross-camera Evaluation

The data normalization procedure is introduced with the motivation that it produces cropped patches that are more agnostic to the specific camera used. Consequently, the models trained using cropped patches would generalize across cameras. To evaluate the generalization across cameras, we conduct a cross-camera evaluation on the EVE dataset with its four cameras, comparing EFE and the data normalization-based approach FaceNet.

We show the result in Tab. 5. From the table, we find that EFE generalizes surprisingly well to extreme camera view changes, exhibiting large performance differences to a data normalization-based approach (FaceNet). For example, EFE achieves 13.2° gaze direction error when training on the W_L camera and testing on the W_R camera, which is a 16.8° improvement compared to FaceNet. This is likely due to the way in which EFE decomposes the PoG estimation problem into the two sub-tasks of gaze origin and gaze direction estimation.

Train / Test	MVC	W$_C$	W$_L$	W$_R$
MVC	-	31.6	36.8	42.0
W$_C$	30.6	-	10.9	11.6
W$_L$	31.3	**6.9**	-	30.0
W$_R$	33.1	**7.3**	22.1	-

(a) Errors achieved by FaceNet

Train / Test	MVC	W$_C$	W$_L$	W$_R$
MVC	-	**23.4**	**24.8**	**28.4**
W$_C$	**23.0**	-	**10.1**	**11.0**
W$_L$	**23.3**	11.1	-	**13.2**
W$_R$	**28.0**	11.8	**14.5**	-

(b) Errors achieved by EFE

Table 5. **Cross-camera gaze direction error (°).** We conduct a challenging cross-camera and cross-person evaluation where models are only trained with one camera view and tested on another camera view. We use four cameras in the EVE dataset including the machine vision camera (MVC) under the display, the webcam on the top (W$_C$), top left (W$_L$), and top right (W$_R$) of the display. We see that EFE outperforms the data normalization-based FaceNet in most cross-camera configurations, by large margins in many cases.

5. Conclusion

Existing methods in gaze estimation have typically relied on simple cropping [17] or data normalization [29, 40] as a scheme for simplifying the gaze direction estimation problem and for achieving higher performances. In this paper, we challenged this paradigm and proposed an architecture that can predict gaze origin from input camera frames directly. To the best of our knowledge, we are the first to demonstrate a solution to this problem in a learning-based approach and end-to-end manner. Furthermore, we were able to show that despite the smaller effective face size in the frame-to-gaze setting, the proposed EFE is able to achieve comparable performances to state-of-the-art methods. Future work could expand our proof-of-concept by proposing alternative methods of breaking down the complex problem of PoG estimation on mobile and edge devices in many practical settings.

Limitations. The camera-to-screen geometry must be known *a priori* for EFE – a limitation that we share with data normalization-based works. Few-shot learning methods could be adopted in the future to adapt to novel camera-to-screen geometries with minimal user input.

References

[1] Shumeet Baluja and Dean Pomerleau. Non-intrusive gaze tracking using artificial neural networks. *NeurIPS*, 6, 1993. 2

[2] Yiwei Bao, Yihua Cheng, Yunfei Liu, and Feng Lu. Adaptive feature fusion network for gaze tracking in mobile tablets. In *ICPR*, pages 9936–9943. IEEE, 2021. 7

[3] Ralf Biedert, Georg Buscher, Sven Schwarz, Jörn Hees, and Andreas Dengel. Text 2.0. In *CHI EA*, pages 4003–4008. 2010. 1

[4] Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks). In *ICCV*, Oct 2017. 3

[5] Olivier Chapelle and Mingrui Wu. Gradient descent optimization of smoothed information retrieval metrics. *Information retrieval*, 13(3):216–235, 2010. 4

[6] Yihua Cheng, Feng Lu, and Xucong Zhang. Appearance-based gaze estimation via evaluation-guided asymmetric regression. In *ECCV*, pages 100–115, 2018. 2

[7] Yihua Cheng, Xucong Zhang, Feng Lu, and Yoichi Sato. Gaze estimation by exploring two-eye asymmetry. *IEEE Transactions on Image Processing*, 29:5259–5272, 2020. 7

[8] Anna Maria Feit, Lukas Vordemann, Seonwook Park, Caterina Berube, and Otmar Hilliges. Detecting relevance during decision-making from eye movements for ui adaptation. In *ETRA*, pages 1–11, 2020. 1

[9] Tobias Fischer, Hyung Jin Chang, and Yiannis Demiris. Rt-gene: Real-time eye gaze estimation in natural environments. In *ECCV*, pages 334–352, 2018. 2

[10] 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*, pages 255–258, 2014. 2

[11] Tianchu Guo, Yongchao Liu, Hui Zhang, Xiabing Liu, Youngjun Kwak, Byung In Yoo, Jae-Joon Han, and Changkyu Choi. A generalized and robust method towards practical gaze estimation on smart phone. In *ICCV Workshops*, pages 0–0, 2019. 7

[12] Junfeng He, Khoi Pham, Nachiappan Valliappan, Pingmei Xu, Chase Roberts, Dmitry Lagun, and Vidhya Navalpakkam. On-device few-shot personalization for real-time gaze estimation. In *ICCV workshops*, pages 0–0, 2019. 7

[13] Qiong Huang, Ashok Veeraraghavan, and Ashutosh Sabharwal. Tabletgaze: dataset and analysis for unconstrained appearance-based gaze estimation in mobile tablets. *Machine Vision and Applications*, 28(5):445–461, 2017. 1, 2, 3

[14] Rudolf Kajan, Adam Herout, Roman Bednarik, and Filip Povolnỳ. Peeplist: Adapting ex-post interaction with pervasive display content using eye tracking. *Pervasive and Mobile Computing*, 30:71–83, 2016. 1

[15] Petr Kellnhofer, Adria Recasens, Simon Stent, Wojciech Matusik, and Antonio Torralba. Gaze360: Physically unconstrained gaze estimation in the wild. In *ICCV*, pages 6912–6921, 2019. 2

[16] Rakshit Kothari, Shalini De Mello, Umar Iqbal, Wonmin Byeon, Seonwook Park, and Jan Kautz. Weakly-supervised physically unconstrained gaze estimation. In *CVPR*, pages 9980–9989, 2021. 2

[17] Kyle Krafka, Aditya Khosla, Petr Kellnhofer, Harini Kannan, Suchendra Bhandarkar, Wojciech Matusik, and Antonio Torralba. Eye Tracking for Everyone. In *CVPR*, 2016. 1, 2, 3, 4, 5, 6, 7, 8

[18] Erik Lindén, Jonas Sjostrand, and Alexandre Proutiere. Learning to personalize in appearance-based gaze tracking. In *ICCV Workshops*, pages 0–0, 2019. 2

[19] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2018. 5

[20] Feng Lu, Yusuke Sugano, Takahiro Okabe, and Yoichi Sato. Adaptive linear regression for appearance-based gaze estimation. *TPAMI*, 36(10):2033–2046, 2014. 2

[21] Päivi Majaranta and Andreas Bulling. Eye tracking and eye-based human–computer interaction. In *Advances in physiological computing*, pages 39–65. Springer, 2014. 1

[22] Anton Mikhailov. Turbo, an improved rainbow colormap for visualization. *Google AI Blog*, 10:15–16, 2019. 8

[23] Alexandra Papoutsaki, Patsorn Sangkloy, James Laskey, Nediyana Daskalova, Jeff Huang, and James Hays. Webgazer: Scalable webcam eye tracking using user interactions. In *IJCAI*, 2016. 1

[24] Seonwook Park, Emre Aksan, Xucong Zhang, and Otmar Hilliges. Towards end-to-end video-based eye-tracking. In *ECCV*, pages 747–763. Springer, 2020. 1, 2, 3, 4, 6

[25] Seonwook Park, Shalini De Mello, Pavlo Molchanov, Umar Iqbal, Otmar Hilliges, and Jan Kautz. Few-shot adaptive gaze estimation. In *ICCV*, 2019. 2

[26] Seonwook Park, Adrian Spurr, and Otmar Hilliges. Deep pictorial gaze estimation. In *ECCV*, pages 721–738, 2018. 2

[27] Laura Sesma, Arantxa Villanueva, and Rafael Cabeza. Evaluation of pupil center-eye corner vector for gaze estimation using a web cam. In *ETRA*, pages 217–220, 2012. 2

[28] Yusuke Sugano, Yasuyuki Matsushita, and Yoichi Sato. Appearance-based gaze estimation using visual saliency. *TPAMI*, 35(2):329–341, 2012. 3

[29] Yusuke Sugano, Yasuyuki Matsushita, and Yoichi Sato. Learning-by-Synthesis for Appearance-based 3D Gaze Estimation. In *CVPR*, 2014. 1, 2, 4, 8

[30] Yusuke Sugano, Yasuyuki Matsushita, Yoichi Sato, and Hideki Koike. An incremental learning method for unconstrained gaze estimation. In *ECCV*, pages 656–667. Springer, 2008. 2

[31] Mingxing Tan and Quoc Le. Efficientnetv2: Smaller models and faster training. In *ICML*, pages 10096–10106. PMLR, 2021. 5, 6

[32] Jonathan J Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. *NeurIPS*, 27, 2014. 3

[33] Kang Wang, Rui Zhao, Hui Su, and Qiang Ji. Generalizing eye tracking with bayesian adversarial learning. In *CVPR*, pages 11907–11916, 2019. 2

[34] Yaoming Wang, Yangzhou Jiang, Jin Li, Bingbing Ni, Wenrui Dai, Chenglin Li, Hongkai Xiong, and Teng Li. Contrastive regression for domain adaptation on gaze estimation. In *CVPR*, pages 19376–19385, 2022. 1

[35] Erroll Wood, Tadas Baltruaitis, Xucong Zhang, Yusuke Sugano, Peter Robinson, and Andreas Bulling. Rendering of eyes for eye-shape registration and gaze estimation. In *ICCV*, pages 3756–3764, 2015. 2

[36] 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*, pages 131–138, 2016. 2

[37] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua. Lift: Learned invariant feature transform. In *ECCV*, pages 467–483. Springer, 2016. 4

[38] Mingfang Zhang, Yunfei Liu, and Feng Lu. Gazeonce: Real-time multi-person gaze estimation. In *CVPR*, pages 4197–4206, 2022. 1, 2, 3

[39] Xucong Zhang, Seonwook Park, Thabo Beeler, Derek Bradley, Siyu Tang, and Otmar Hilliges. Eth-xgaze: A large scale dataset for gaze estimation under extreme head pose and gaze variation. In *ECCV*, pages 365–381. Springer, 2020. 2

[40] Xucong Zhang, Yusuke Sugano, and Andreas Bulling. Revisiting data normalization for appearance-based gaze estimation. In *ETRA*, 2018. 1, 2, 8

[41] Xucong Zhang, Yusuke Sugano, and Andreas Bulling. Evaluation of appearance-based methods and implications for gaze-based applications. In *CHI*, pages 1–13, 2019. 1, 2

[42] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. Appearance-based gaze estimation in the wild. In *CVPR*, 2015. 1, 2

[43] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. It's written all over your face: Full-face appearance-based gaze estimation. In *CVPR Workshops*, 2017. 2, 3, 4, 6, 7

[44] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. Mpiigaze: Real-world dataset and deep appearance-based gaze estimation. *TPAMI*, 2019. 2, 5

[45] Zhiwei Zhu and Qiang Ji. Novel eye gaze tracking techniques under natural head movement. *IEEE Transactions on biomedical engineering*, 54(12):2246–2260, 2007. 1