Rotation-Constrained Cross-View Feature Fusion for Multi-View Appearance-based Gaze Estimation

Yoichiro Hisadome, Tianyi Wu, Jiawei Qin, Yusuke Sugano
Institute of Industrial Science, The University of Tokyo
Komaba 4-6-1, Tokyo, Japan
{hisadome, twu223, jqin, sugano}@iis.u-tokyo.ac.jp

Abstract

Appearance-based gaze estimation has been actively studied in recent years. However, its generalization performance for unseen head poses is still a significant limitation for existing methods. This work proposes a generalizable multi-view gaze estimation task and a cross-view feature fusion method to address this issue. In addition to paired images, our method takes the relative rotation matrix between two cameras as additional input. The proposed network learns to extract rotatable feature representation by using relative rotation as a constraint and adaptively fuses the rotatable features via stacked fusion modules. This simple yet efficient approach significantly improves generalization performance under unseen head poses without significantly increasing computational cost. The model can be trained with random combinations of cameras without fixing the positioning and can generalize to unseen camera pairs during inference. Through experiments using multiple datasets, we demonstrate the advantage of the proposed method over baseline methods, including state-of-the-art domain generalization approaches. The code will be available at https://github.com/ut-vision/Rot-MVGaze.

1. Introduction

In anticipation of various applications such as robotics and accessibility, gaze estimation techniques have long been the subject of active research [1, 4, 17, 58]. Appearance-based methods take the machine learning approach to directly estimate 3D gaze directions from input eye or face images, and have shown great potential for robust real-world gaze estimation [11,21]. One of the key challenges in appearance-based gaze estimation is its limited generalization performance to unseen conditions. The performance of gaze estimation models is often affected by various factors, including individuality, illumination, and the distributions of gaze and head pose. Although various datasets have been proposed [18, 19, 26, 35, 57, 63], creating a generic model that can handle arbitrary conditions is still not a trivial task.

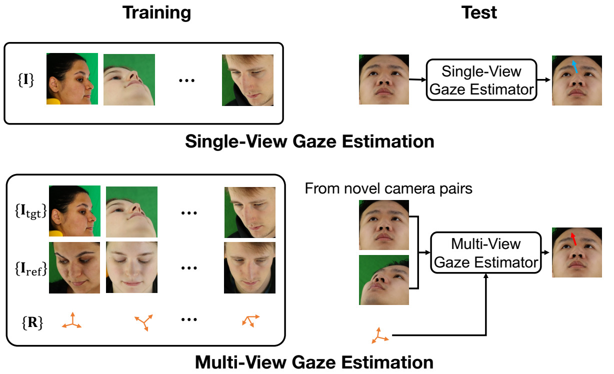
Training
Test
{I}
...
Single-View Gaze Estimator
Single-View Gaze Estimation
$\{I_{tgt}\}$
$\{I_{ref}\}$
{R}
From novel camera pairs
Multi-View Gaze Estimator
Multi-View Gaze Estimation

Figure 1. Overview of the proposed multi-view gaze estimation task. We estimate the 3D gaze direction from multiple synchronized images. The model can be generalized to unseen camera combinations unavailable during training by leveraging the relative rotation between cameras.

Most of the existing appearance-based methods take monocular images as input and formulate gaze estimation as a task to estimate the gaze direction vector defined in the input image coordinate system. For this reason, generalization difficulties in appearance-based gaze estimation vary greatly depending on the factors. Specifically, unseen people and lighting conditions affect the facial appearance but do not fundamentally change the pattern of faces in the image. In contrast, unseen head poses and their associated unseen gaze directions lead to entirely new patterns and have a more direct impact on the input-output relationship. Therefore, it is usually more difficult for existing gaze estimation models to generalize to unseen head poses.

If not limited to appearance-based methods, multi-camera geometry-based eye tracking systems have long been the subject of active research [2, 39, 43, 45, 50]. Such a multi-view approach may solve the above problems in appearance-based methods. For many application scenarios, such as driver monitoring and public displays, using multiple synchronized machine vision cameras is a sufficiently realistic assumption for appearance-based gaze estimation. The expected effect of multi-view input is not limited to simply increasing information and improving accuracy. The model could acquire a head pose-independent feature representation by training a gaze estimation model considering the geometric positional relationship between input images. However, considering the cost of training data acquisition, training a model specialized for cameras in a particular positional relationship is not practical. The key challenge is to train a model that can perform accurate gaze estimation even if the camera's positional relationship changes between inference and training.

This work proposes a multi-view appearance-based gaze estimation method that utilizes the relative rotation between cameras as additional input information (Fig. 1). Assuming the normalization process used in appearance-based gaze estimation [59], a relative rotation matrix can always express the interrelationship of camera positions. The main idea of the proposed method is to use the rotation matrix as a constraint for feature fusion between images. The proposed method consists of stacked rotation-constrained feature fusion blocks that can be combined with arbitrary feature extraction backbones. In each block, one of the features is multiplied by the rotation matrix to transfer to the other image. Although the physical rotation is not originally applicable to the feature space, the model is expected to learn to extract rotatable features through the explicit training process incorporating the rotation operation. We demonstrate that our method acquires rotatable feature representation through experimental analyses on multiple datasets [42,57]. The proposed method achieves better generalizability than baseline approaches, including state-of-the-art domain generalization methods.

Our key contributions are threefold. First, this paper addresses the camera-independent multi-view appearance-based gaze estimation task for the first time in the literature. Second, we propose a novel cross-view feature fusion approach incorporating the relative rotation matrix into multi-view gaze estimation. Our method uses the rotation matrix as a constraint to transfer features between images, and we provide a thorough analysis of the internal feature representation. Third, we demonstrate that multi-view gaze estimation improves generalization performance for unseen head poses. Through experiments, we show that the accuracy gains from multi-view training are superior to state-of-the-art methods. We also provide thorough analyses and visualizations of the internal feature representation obtained through the rotation constraint.

2. Related Work

Appearance-based Gaze Estimation. Appearance-based gaze estimation is a task to regress 3D gaze directions from full-face [7, 9, 29, 42, 57, 59, 62] or eye-region images [10, 12, 19, 41, 48, 56, 61, 63]. In recent years, the advances in deep neural networks have enabled gaze estimation techniques with decent within-dataset performance [6, 8, 9, 57, 62]. However, noticeable performance degradation can be observed when deployed in real-world applications. Consequently, the performance of state-of-the-art approaches would be limited under unconstrained conditions, primarily when the gap arising from the aforementioned factors is significantly large during training and testing. While there are ongoing attempts on personalization [40, 55], and domain adaptation [5, 32, 46], training models that can generalize to unknown environments is a significant challenge. This study addresses this domain gap issue by improving the generalization ability to unseen head poses.

Given the various factors to consider for generalizing appearance-based gaze estimation, previous studies have proposed many datasets. Because it is difficult to cover all factors concurrently, many datasets were constructed focusing on a certain diversity [19, 35, 63]. Covering a variety of head poses is challenging, and each recent dataset still has limitations, such as the lighting conditions diversity [57] and the gaze labels accuracy [26]. Although intended for a monocular estimation task, the ETH-XGaze dataset [57] was created using multiple synchronized cameras and can be used for multi-view purposes. Recent work on synthesizing face images with ground-truth gaze directions further indicates the possibility of acquiring training data for multi-view estimation [42, 44, 54, 64]. This study examines the novel task of multi-view gaze estimation on these datasets.

Domain Generalization for Gaze Estimation. There are some prior attempts to alleviate the gap between training and testing environments through domain generalization. Some prior work [16, 28] use large-scale unlabeled face images for pretraining or as an additional training signal to generalize gaze estimator. However, these approaches still require extra samples from either the target domain or the Internet, which is often nontrivial to be prepared in practice even without ground-truth gaze labels. The proposed method differs from these approaches because it does not use extra data for achieving generalization.

Another line of work on domain generalization directly improves model robustness on unseen domains by removing person-dependent factors during training and is thus closer to our objective [7, 40]. We note that the goal of multi-view gaze estimation is not only domain generalization, and the direction of this work is not strictly consistent with them.

Nevertheless, our approach improves robustness on unseen head poses, which none of the above-mentioned methods have explicitly proven.

Multi-view Feature Fusion. Most previous works on multi-view eye tracking take the model-based approach [2, 3, 39, 43, 45], which requires a more complex setup with external light sources. Although some image-only multi-view methods exist [50], they still rely on geometric eyeball models. Prior research has shown that such geometry or shape-based approaches are inferior to appearance-based methods in terms of performance [60]. In contrast, appearance-based multi-view gaze estimation [22, 30] has been understudied. Lian *et al*. [30] proposed directly concatenating features from stereo images to predict 2D on-screen gaze positions. Gideon *et al*. [22] proposed disentangling image features through feature swapping between multi-view videos, different from our frame-by-frame setting. However, these methods require fixed cameras during training and testing, and their effectiveness in unknown camera configurations remains unproven. The proposed method uses the relative rotation matrix between camera pairs as additional input to overcome this drawback, achieving multi-view gaze estimation generalizable to unseen camera pairs.

Multi-view input has also been explored in many computer vision tasks, but the direct use of these methods for gaze estimation is not straightforward. Recent work on multi-view stereo constructs and uses 3D voxel representation from multi-view features [23, 33, 52, 53]. While such a voxel representation is suited to geometry-related tasks, it is not directly applicable to gaze estimation, which is rather an attribute regression task. NeRF [15, 20, 31, 34, 36, 37] can also be applied to gaze redirection and training data synthesis [44, 54], but it is not yet directly related to estimation tasks. Unlike these approaches strongly based on physical geometry, our method uses relative camera rotation as a soft constraint for learnable feature extraction and fusion blocks.

3. Method

This work aims to design a network to efficiently perform feature transitions and fusions between images according to the input rotation matrix. Since rotation has an extremely low dimensionality compared to image features, it is not optimal to feed it into the network as another feature. The proposed method achieves this goal via stacked rotation-constrained feature fusion blocks.

3.1. Overview

Fig. 2 shows the overview of the proposed network, which consists of stacked rotation-constrained feature fusion blocks. The inputs are the target image $\mathbf{I}_{\mathrm{tgt}}$ and reference image $\mathbf{I}_{\mathrm{ref}}$. As described earlier, these face images are assumed to be normalized for appearance-based gaze estimation [59], and their mutual relationship can be fully described by the rotation matrix $\mathbf{R}$. $\mathbf{R}$ indicates the rotation from the reference camera to the target camera coordinate systems and can be obtained either from the extrinsic camera calibration, or head poses estimated through the normalization process. While the goal is to estimate the gaze direction $\mathbf{g}_{\mathrm{tgt}}$ of the target image $\mathbf{I}_{\mathrm{tgt}}$, the role of the target $\mathbf{I}_{\mathrm{tgt}}$ and the reference $\mathbf{I}_{\mathrm{ref}}$ are symmetrical in our method. The network components on both sides thus share weights, but the stacked fusion blocks do not share weights.

Given input images, the proposed method first extracts two features. The network first extracts the backbone feature vectors $\mathbf{f}_{\mathrm{tgt}}$ and $\mathbf{f}_{\mathrm{ref}}$ using the *Backbone Extractor* module. The *Rotatable Feature Extractor* module then takes these backbone features as inputs and outputs D three-dimensional vectors. These vectors are stacked to form rotatable feature tensors $\mathbf{F}^{(0)}_{\mathrm{ref}}, \mathbf{F}^{(0)}_{\mathrm{tgt}} \in \mathbb{R}^{3\times D}$. While backbone features are used unaltered throughout the process, the rotatable features are updated through the rotation-constrained feature fusion blocks. The output gaze vectors $\mathbf{g}_{\mathrm{tgt}}$ and $\mathbf{g}_{\mathrm{ref}}$ are defined as 3D unit vectors in each normalized camera coordinate system.

3.2. Rotation-Constrained Feature Fusion

As discussed earlier, the basic idea behind the rotation-constrained feature fusion block is directly applying the rotation (multiplying the rotation matrix) in the feature space. We expect the network to learn to extract rotatable feature representation by introducing a rotation-constrained feature fusion mechanism. The rotated and transferred features are expected to complement the information in each image, and therefore the optimal features cannot be obtained simply by observing individual images. The proposed method is designed to achieve optimal feature transfer by repeating the process of fusing the rotated features with the backbone features of the destination.

In the i-th block, the model first multiplies the rotation matrix $\mathbf{R}$ to the reference rotatable feature $\mathbf{F}^{(i-1)}_{\mathrm{ref}}$. The *Fuser* module then fuses the rotated feature with the backbone feature $\mathbf{f}_{\mathrm{tgt}}$ of the target image to obtain an updated rotatable feature $\mathbf{F}^{(i)}_{\mathrm{tgt}}$ of the target image. The network is symmetric and applies the same operation to update the rotatable feature of the reference image using the back-rotated reference feature $\mathbf{R}^{\top}\mathbf{F}^{(i)}_{\mathrm{tgt}}$. The *Gaze Estimator* modules then output intermediate estimates of gaze directions $\mathbf{g}^{(i)}_{\mathrm{tgt}}$ and $\mathbf{g}^{(i)}_{\mathrm{ref}}$ using both backbone and rotatable features. The model repeats the above fusion process until l blocks, and the l-th estimated gaze $\mathbf{g}^{(l)}_{\mathrm{tgt}}$ becomes the final output.

The loss function $\mathcal{L}_{\mathrm{total}}$ is defined for all intermediate

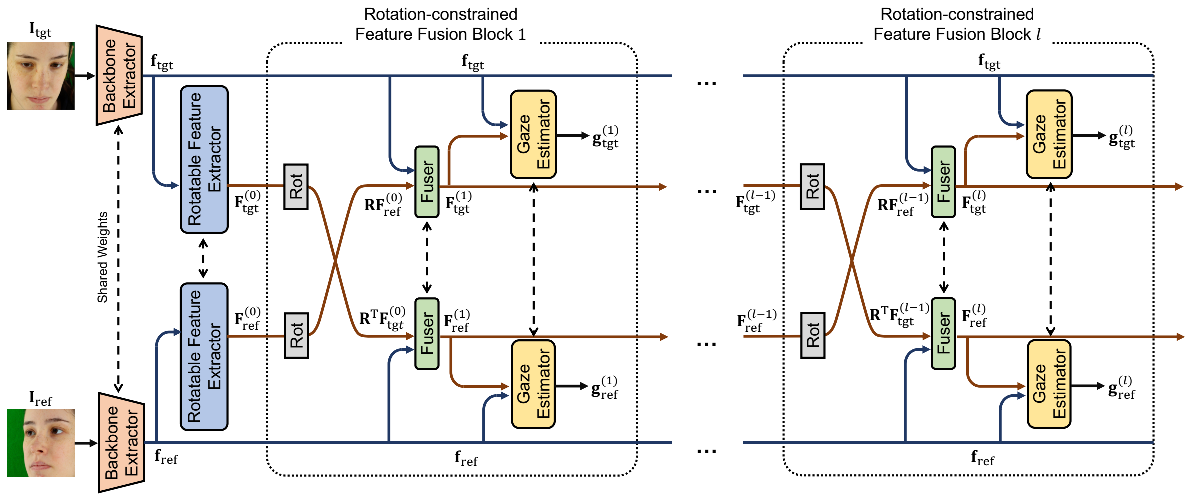
$\mathbf{I}_{\text{tgt}}$
Backbone Extractor
$\mathbf{f}_{\text{tgt}}$
Rotation-constrained Feature Fusion Block 1
Rotatable Feature Extractor
$\mathbf{F}^{(0)}_{\text{tgt}}$
Rot
$\mathbf{R}\mathbf{F}^{(0)}_{\text{ref}}$
Fuser
$\mathbf{F}^{(1)}_{\text{tgt}}$
Gaze Estimator
$\mathbf{g}^{(1)}_{\text{tgt}}$
Shared Weights
Rotatable Feature Extractor
$\mathbf{F}^{(0)}_{\text{ref}}$
Rot
$\mathbf{R}^{\top}\mathbf{F}^{(0)}_{\text{tgt}}$
Fuser
$\mathbf{F}^{(1)}_{\text{ref}}$
Gaze Estimator
$\mathbf{g}^{(1)}_{\text{ref}}$
$\mathbf{I}_{\text{ref}}$
Backbone Extractor
$\mathbf{f}_{\text{ref}}$
...
Rotation-constrained Feature Fusion Block l
$\mathbf{F}^{(l-1)}_{\text{tgt}}$
$\mathbf{R}\mathbf{F}^{(l-1)}_{\text{ref}}$
$\mathbf{F}^{(l)}_{\text{tgt}}$
$\mathbf{g}^{(l)}_{\text{tgt}}$
$\mathbf{F}^{(l-1)}_{\text{ref}}$
$\mathbf{R}^{\top}\mathbf{F}^{(l-1)}_{\text{tgt}}$
$\mathbf{F}^{(l)}_{\text{ref}}$
$\mathbf{g}^{(l)}_{\text{ref}}$

Figure 2. The overview of the proposed network which consists of stacked rotation-constrained feature fusion blocks. The subscripts *ref* and *tgt* indicate reference and target images, respectively, and the superscripts denote block id. While network components share weights between the target and reference sides, they do not share weights across stacked blocks.

outputs $\mathbf{g}^{(i)}$, not just the final output $\mathbf{g}^{(l)}$ as

$$\mathcal{L}_{\text{total}} = \sum_{i=1}^{l} \alpha^{l-i} \cdot \mathcal{L}_i(\mathbf{g}^{(i)}_{\text{tgt}}, \mathbf{g}^{(i)}_{\text{ref}}). \tag{1}$$

$\mathcal{L}_i$ is the angular loss for the i-th block defined as

$$\mathcal{L}_i = \arccos\left(\mathbf{g}^{(i)\top}_{\text{tgt}}\hat{\mathbf{g}}_{\text{tgt}}\right) + \arccos\left(\mathbf{g}^{(i)\top}_{\text{ref}}\hat{\mathbf{g}}_{\text{ref}}\right), \tag{2}$$

where $\hat{\mathbf{g}}_{\text{tgt}}$ and $\hat{\mathbf{g}}_{\text{ref}}$ indicate the ground-truth 3D gaze directions corresponding to target and reference images. α is a hyperparameter indicating the decay to gaze estimated from earlier blocks.

3.3. Rotation Matrix

As discussed earlier, the relative rotation matrix $\mathbf{R}$ can be obtained either from camera calibration or head pose estimation. $\mathbf{R}$ describes the rotation term between the normalized cameras. The meaning of $\mathbf{R}$ is the same in either approach, and the results are the same if there is no head pose estimation error. Although a translation vector $\mathbf{t}$ is also needed to describe the relationship between two cameras completely, it is uniquely determined by $\mathbf{R}$ and can be ignored under the assumption of data normalization [59].

For the first option based on camera calibration, the rotation matrix can be calculated using the camera extrinsic parameters and the normalization matrices [59] as $\mathbf{R} = \mathbf{N}_{\text{tgt}}\tilde{\mathbf{R}}\mathbf{N}^{\top}_{\text{ref}}$. $\tilde{\mathbf{R}}$ is the rotation matrix obtained via camera calibration and therefore corresponds to the coordinate systems of the original camera before normalization. $\mathbf{N}_{\text{tgt}}$ and $\mathbf{N}_{\text{ref}}$ are the normalization matrices, which indicate the transformation from the original to the normalized camera coordinate systems. For the second option, the rotation is calculated using head poses estimated through the normalization process. If we denote the rotation from the head coordinate system to the normalized camera coordinate system as $\mathbf{H}_{\text{tgt}}$ and $\mathbf{H}_{\text{ref}}$, the rotation matrix is $\mathbf{R} = \mathbf{H}_{\text{tgt}}\mathbf{H}^{\top}_{\text{ref}}$.

3.4. Implementation Details

Unless otherwise noted, we set the number of blocks $l = 3$ in all experiments. We used ResNet-50 [24] as the *Backbone Extractor* module, which was initialized with pretrained weights on ImageNet [13] and fine-tuned through the training process. We used two-layered MLPs for the *Rotatable Feature Extractor* and the *Gaze Estimator*, and a three-layered MLP for the *Fuser*, with $D = 512$. The *Rotatable Feature Extractor* receives the backbone feature vector from the *Backbone Extractor* and outputs three D-dimensional vectors. These feature vectors are stacked to form a $3 \times D$ rotatable feature matrices. The *Fuser* and *Gaze Estimator* first flatten the $3 \times D$ shaped rotatable feature matrices. The flattened matrices are then concatenated with the backbone feature vectors from the other view. Like the *Rotatable Feature Extractor*, the *Fuser* also outputs three D-dimensional vectors stacked to form a $3 \times D$ matrix. All MLPs use ReLU [38] as the activation layer.

During the training, we applied random mask data augmentation to the input images to force the model to exploit features from another view. Inspired by random erasing [65], we masked the input image with multiple randomly sized (5–30% of the image width) small squares with a 50% probability. The quantity of squares was determined randomly so that the proportion of the total area covered by the squares was restricted to 50–60% of the face image. We also applied color jitter, translation, and scaling. We set the saturation, brightness, and contrast range to 0.1. The intensities of translation and scaling were set to be small (0.01 for translation, 0.99–1.01 for scaling) to represent possible face alignment errors during the normalization process. We apply the same data augmentation to all baseline methods for fair comparisons.

We set the training batch size to 256. We used Adam [27] optimizer with a weight decay of 1×10^{-6}. We used CyclicLR [47] as the learning rate scheduler, with the base and maximum learning rate of 1×10^{-6} and 1×10^{-3}, decaying 0.5 per cycle. The cycle steps were determined so that one cycle was completed in one epoch. We used mixed precision training and set the block decay α to 0.5.

4. Experiments

4.1. Experimental Setting

We use two datasets that provide synchronized multiple views of participants and corresponding 3D gaze directions. **ETH-XGaze** [57] contains 110 participants, each captured with 18 cameras simultaneously. Since one of our goals is to evaluate the generalization performance against unseen head poses, we split the training subset of 80 participants instead of using their official test data. We directly use the camera extrinsic parameters and head pose estimation results provided with the dataset. **MPII-NV** is a synthetic dataset created by following Qin *et al*.'s approach [42]. The dataset is based on the MPIIFaceGaze dataset [63] that consists of monocular images of 15 participants. 3D face meshes are first reconstructed from the original MPIIFaceGaze images and then rotated with their ground-truth gaze vector to generate face images of new head poses. We synthesized the data so that the head pose distribution is the same as that of the ETH-XGaze training set. Since it is a purely synthetic dataset, we can use the camera position for image rendering to compute the relative rotation matrix. After data normalization, the input image resolution for both datasets is 224×224. Both datasets were collected with IRB approval or consent from participants.

We used k-fold cross-validation regarding participant IDs ($k = 4$ for ETH-XGaze and $k = 3$ for MPII-NV), and all methods were trained for 15 epochs without validation data. Given the practical scenario where the camera positions at the deployment are not necessarily known at the training time, generalization performance for unseen camera positions is an important metric. Therefore, we further split the cameras into training and test sets in each fold to evaluate the generalization performance. Specifically, we split the 18 cameras of both ETH-XGaze and MPII-NV into 12 for training and 6 for testing, three of which are within the head-pose range of the training camera set (interpolation), and the other three are outside (extrapolation). Please note that head poses with respect to the camera position are nearly fixed under the ETH-XGaze setup, and there is a strong correlation between the head poses and camera positions. Training and testing image pairs are constructed by randomly selecting two cameras from each set.

Concat represents the approach of Lian *et al*. [30] as the baseline multi-view appearance-based method. It extracts features from two images using weight-shared ResNet-50 and then estimates the gaze direction using concatenated features. **Single** is the single-image baseline method corresponding to the one reported in ETH-XGaze paper [57]. It extracts features from the monocular input image using ResNet-50 [24] followed by a fully-connected layer to output gaze direction. **Gaze-TR** [9] is one of the state-of-the-art methods for single-image gaze estimation. We adopted the hybrid version containing a ResNet-50 [24] extractor and a transformer encoder [14,51]. It extracts features from ResNet-50 [24] and feeds the feature maps to the transformer encoder, followed by an MLP to output the gaze directions. **Frontal Selection** is fundamentally a single-view model, but multi-view information is utilized naively during inference. It predicts the gaze based on the more frontal image from the reference and target images.

Since our goal is generalizable gaze estimation, we also include some single-image domain generalization approaches as baselines. Since our method requires no prior knowledge of the target domain, we excluded domain adaptation methods which require target domain data. Please note that unsupervised domain adaptation still requires target domain images. **PureGaze** [7] introduced an extra CNN-based image reconstruction module to the ResNet-18 backbone and MLP. We followed the official implementation for MLP and the reconstruction module while replacing the backbone with ResNet-50. **DT-ED** [40] first extracts the latent codes of appearance, gaze, and head pose from the source image, then a decoder is used to reconstruct the target image from the rotated head pose and gaze features, and an MLP is used to predict gaze directions from the gaze features only. We follow the original structure of 4-block DenseNet [25] and a growth rate of 32, and the target image was randomly chosen from the same subject.

4.2. Performance Comparison

Within-Dataset Evaluation. We first conduct a within-dataset evaluation on ETH-XGaze and MPII-NV. For both datasets, we present the cases where the relative rotation matrix $\mathbf{R}$ is from the camera extrinsic calibration. We report the angular error averaged over the k folds in Table 1a. The *Head pose* column corresponds to the split of the cameras as described in Sec. 4.1. For ETH-XGaze under the seen head pose condition, multi-view approaches (*Concat* and *Proposed*) consistently outperform single-image methods. However, the *Concat* model shows lower accuracy in the unseen head pose condition than the single-image

Train/Test	XGaze (Calib.)		MPII-NV	
Head pose	Seen	Unseen	Seen	Unseen
Single	4.07°	6.97°	7.45°	8.60°
Gaze-TR [9]	4.16°	7.24°	7.38°	8.47°
DT-ED [40]	5.07°	7.88°	7.91°	9.44°
PureGaze [7]	3.99°	7.98°	7.59°	9.13°
Frontal Selection	3.70°	5.58°	7.04°	7.40°
Concat [30]	3.71°	8.60°	**6.78°**	8.41°
Proposed	**3.50°**	**4.95°**	6.81°	**7.06°**

(a) Within-dataset evaluation.

Train	MPII-NV		XGaze (Calib.)	
Test	XGaze (Calib.)		MPII-NV	
Head pose	Seen	Unseen	Seen	Unseen
Single	18.44°	18.40°	19.71°	18.10°
Gaze-TR [9]	18.90°	17.35°	15.34°	17.06°
DT-ED [40]	14.39°	17.20°	24.08°	27.05°
PureGaze [7]	18.63°	22.59°	14.52°	14.86°
Frontal Selection	19.23°	19.23°	16.16°	14.71°
Concat [30]	17.73°	17.82°	14.43°	14.23°
Proposed	**14.27°**	**14.55°**	**12.60°**	**12.07°**

(b) Cross-dataset evaluation.

Table 1. Within- and cross-dataset evaluation using rotation from camera calibration. Each number indicates the mean angular error of the proposed and baseline methods.

baselines. This shows the difficulty of learning a generic head pose-independent gaze feature under the multi-view condition. Our proposed method with feature rotation and repetitive fusion achieves the best performance. In particular, it improves 0.57° (14.0%) over the *Single* baseline in the seen head pose condition, and more prominently, improves 1.54° (29.0%) in the unseen head pose condition. This demonstrated the significance of our proposed method for novel-view-generalizable gaze estimation.

For MPII-NV, the proposed method outperforms all single-image and multi-view baselines under unseen head pose conditions. The performance improvement is 8.6% and 17.9% over the *Single* baseline in the seen and unseen head pose conditions, respectively. It is also worth noting that while *Concat* and our proposed method perform almost equivalently well in the *seen* condition, our proposed method outperforms *Concat* by 16.1% in the *unseen* condition. This proves the advantage of the proposed feature rotation and repetitive feature fusion in fusing head-pose-independent representations.

Fig. 3 further visualizes the difference of mean gaze error between the proposed and *Single* model in the ETH-XGaze unseen head pose condition. We can see that the gaze estimation errors drastically decrease when the target head pose corresponds to extrapolation (cameras 11, 14, 17) and the reference head pose corresponds to interpolation (cameras 2, 5, 8). We can also confirm a decent error reduction even when the head poses correspond to extrapolation.

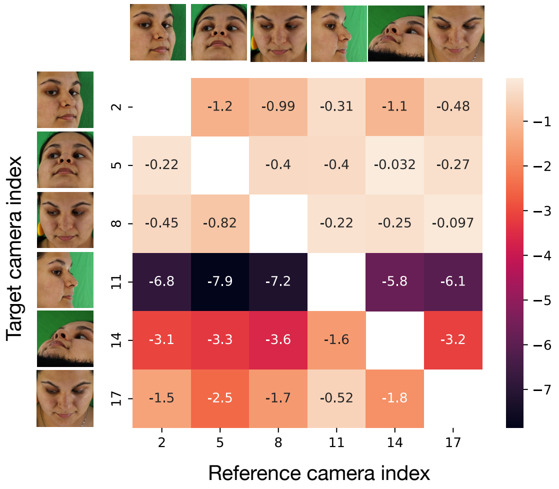

Figure 3. Visualization of the performance gain from multi-view input. The numbers on the x and y axis indicate the camera index in ETH-XGaze. The numbers and colors in the matrix indicate the mean gaze error between the gaze estimation errors of the proposed method and *Single* model.

Cross-Dataset Evaluation. We further evaluate the performance of the proposed method in the cross-dataset setting. We train on one of ETH-XGaze and MPII-NV and evaluate angular errors on the other dataset. Since the two datasets contain different participants, we use all participants in one dataset for training and all participants in the other for the test. Table 1b shows the results of the cross-dataset evaluation in the unseen and seen head pose conditions. We can observe the same tendencies as within-dataset evaluation in Table 1a. In both conditions, the accuracy of our proposed method is much higher than any of the baseline methods. From these results, it can be seen that the proposed method is also effective in reducing the inter-domain gap. The proposed method performs better than the *Concat* baseline, indicating that the reduction of the inter-domain gap owes to the rotation-constrained feature fusion rather than multi-view estimation.

4.3. Detailed Performance Analyses

Ablation Studies. In Table 2, we compare different usage of the rotation matrix. The second row (*w/o Rotation matrix* corresponds to the model that uses the stacked architecture but concatenates the features without rotation. The third row (*MLP Encoding*) is the case where the rotation matrix is used as an additional feature instead of multiplication at each block. In this case, we concatenated the flattened rotation matrices with the features and then fed them to an

Head pose	Seen	Unseen	Infer. time
w/o Rotation matrix	3.64°	9.00°	-
MLP encoding	3.66°	8.22°	-
# block = 1	**3.49°**	5.22°	22.9 ms
# block = 2	3.50°	4.99°	23.0 ms
# block = 3 (Proposed)	3.50°	**4.95°**	24.8 ms
# block = 4	3.50°	4.97°	25.2 ms

Table 2. Ablation studies of the rotation encoding approaches and the number of fusion blocks. Inference time is benchmarked on a single NVIDIA V100 GPU.

Test	XGaze (Pose)	
Head pose	Seen	Unseen
Concat [30]	3.71°	8.60°
Proposed (Calib.)	4.63°	**5.68°**
Proposed (Pose.)	**3.63°**	7.12°

Table 3. ETH-XGaze within-dataset evaluation using rotation matrices obtained from head pose without calibration. Each number indicates the mean angular error.

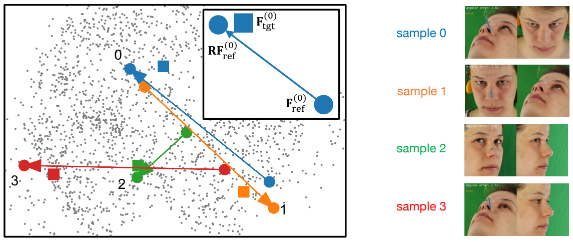

Figure 4. Isomap embedding of the initial rotatable features. The right side shows the example input samples. $\mathbf{F}_{\text{ref}}^{(0)}$, $\mathbf{F}_{\text{tgt}}^{(0)}$, and $\mathbf{RF}_{\text{ref}}^{(0)}$ of the same sample are represented in the same color on the left side plot.

MLP encoder which is almost the same as *Fuser* except for the input feature dimension. As with the proposed model, the learnable weights are shared for target and reference but not shared across blocks for both cases. Although feeding flattened rotation matrices (*MLP Encoding*) improves the accuracy from the *Single*, it is still inferior to the proposed method. We also change the number of fusion blocks from the fourth to the last row in Table 2. The impact of the stacked fusion blocks is different under unseen head pose conditions. However, the three-block model has the best performance for unseen head poses and is almost the best for the seen head pose condition.

The rightmost column in Table 2 shows the inference times of our method on NVidia V100. *Single* model's inference time is 10.7 ms and *Gaze-TR* is 35.0 ms. We can observe that the additional inference cost from additional fusion blocks is relatively minor, and the inference time is almost double that of the *Single* baseline. Although the increase in computational cost is one limitation of the proposed method, it is still faster than more complex single-image methods such as *Gaze-TR* and is considered to be well within the practical range.

Accuracy of Rotation Matrix. In Table 3, we compare the performance of the proposed method using a rotation matrix obtained from estimated head poses without camera calibration. Since the head pose is expected to be perfectly accurate on synthetic MPII-NV, we only evaluate the cases using real images from ETH-XGaze. *Proposed (Calib.)* and *Proposed (Pose)* correspond to the cases where the rotation matrices for training data are obtained from calibration and head pose, respectively. As a reference, we also show the performance of the *Concat* model. Please note that all baseline methods, including *Concat*, do not use a rotation matrix as input. Thus, the numbers are the same as Table 1a.

It can be seen that the proposed model trained with calibration is sensitive to the noise of the rotation matrix at inference times. However, *Proposed (Calib.)* method still performs best for the unseen head pose condition. If the model is trained with rotation matrices from head pose (*Proposed (Pose)*), unseen head pose performance degrades while the performance is improved for the seen head pose condition.

While the *Frontal Selection* in Table 1 shows that simply utilizing multi-view information can improve performance from the *Single*, our proposed approach performs best in most cases, demonstrating the significance of our rotation-constrained feature fusion.

4.4. Rotatable Feature Representation

Fig. 4 shows the features $\mathbf{F}_{\text{ref}}^{(0)}$, $\mathbf{F}_{\text{tgt}}^{(0)}$, and $\mathbf{RF}_{\text{ref}}^{(0)}$ embedded in Isomap [49]. We use the XGaze dataset under the unseen head pose condition. Isomap embedding was generated from the initial rotatable features $\mathbf{F}^{(0)}$ obtained from 1000 test samples with a neighborhood size of 30. Marker shape indicates different feature types, and color indicates the sample ID. The arrows indicate the feature position before and after rotation. Since the rotation is symmetric, we only visualize the rotation from the reference to the target. We can clearly observe that the rotation operation brings the feature closer to the other. This indicates that, as intended, the *3D Feature Extractor* module learns to extract rotatable features through our rotation constraint.

Fig. 5 shows an example of the rotatable features. In this plot, we interpret the rotatable features as a set of D 3D vectors and transform each 3D vector into the pitch-yaw coordinate system. The size and color of the dots represent the magnitude of the norm of the 3D vectors, and large yellow dots indicate vectors with larger norms. The distributions of $\mathbf{RF}_{\text{ref}}^{(0)}$ and $\mathbf{F}_{\text{tgt}}^{(0)}$ become closer by rotation before the

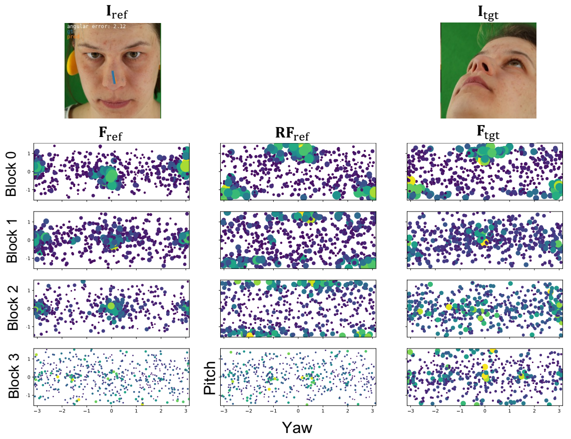

Figure 5. Scatter plot visualization of the rotatable features. Each of the D 3D vectors is represented in a pitch-yaw coordinate system. Each row corresponds to the rotatable features at different fusion stages from $\mathbf{F}^{(0)}$ to $\mathbf{F}^{(3)}$. Larger and yellower dots represent elements with a larger norm.

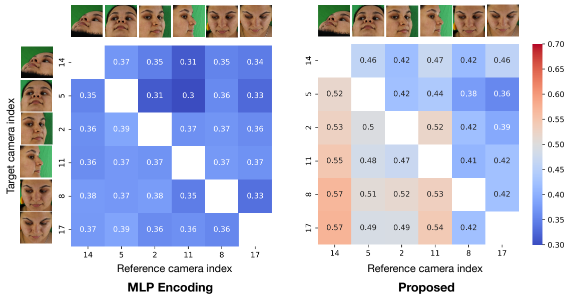

Figure 6. Analysis of contributions of features from each view. The values represent the contribution ratio of the feature from the reference images, which is calculated as the sum of the gradient of the backbone features.

first block, being consistent with Fig. 4. Meanwhile, subsequent fusions with backbone features make the two features different in later blocks, *e.g.*, $\mathbf{RF}_{\text{ref}}^{(3)}$ and $\mathbf{F}_{\text{tgt}}^{(3)}$. We hypothesize that rotatable features adaptively evolve through stacked fusion blocks into complementary representations of the backbone features.

4.5. Contribution of Reference Images

Fig. 6 illustrates the contribution ratio of the reference features for each camera pair. As a metric for feature contribution, we calculated the sum of the gradient of the backbone features. We use the XGaze dataset under the unseen head pose condition as the experimental setup. A larger number represents more contribution of the reference image to the estimation result. Fig. 6 shows the visualization results of the *MLP Encoding* baseline (left) and our proposed method (right). Comparing the two visualization results, we can confirm that our method adaptively uses the reference images. While *MLP Encoding* model always ignores the reference images, our method uses the reference information mainly depending on its head poses.

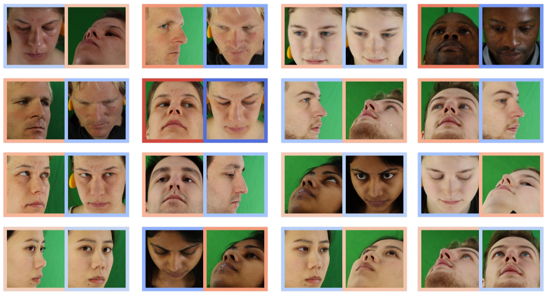

Figure 7. Example of paired images with their contribution to gaze estimation. The left and right images correspond to the target and the reference. The edge color shows the contribution ratio, where red indicates a higher contribution.

Fig. 7 further shows sample images with their corresponding contribution ratios. The edge color of each image represents its contribution as in Fig. 6. Overall, images with a view that captures the face from below have a higher contribution. This is consistent with Fig. 6 where, *e.g.*, camera 14 shows a more significant contribution. Occlusion caused by eyelids is possibly a significant factor in the minor contribution of images captured from the top view.

5. Conclusion

In this paper, we presented a novel multi-view appearance-based gaze estimation task. We propose a cross-view feature fusion approach using the relative rotation matrix between input images as a constraint when transferring the features to the other image. In addition to its practical significance, the proposed method has the advantage of improving generalization performance for unseen head poses. Through experiments, we demonstrated the advantage of our method over state-of-the-art baseline, including single-image domain generalization methods.

The limitation of our approach compared to a single-image baseline is the slightly increased hardware requirements. The requirements of our method are not particularly unrealistic compared to existing eye trackers, and this is ultimately a matter of trade-offs. The same can be said about the effect of camera calibration. It is also essential for future work to develop lightweight models that are robust to errors in the rotation matrix and time synchronization.

Acknowledgement

This work was supported by JSPS KAKENHI Grant Number JP21K11932.

References

[1] Yasmeen Abdrabou, Ahmed Shams, Mohamed Omar Mantawy, Anam Ahmad Khan, Mohamed Khamis, Florian Alt, and Yomna Abdelrahman. Gazemeter: Exploring the usage of gaze behaviour to enhance password assessments. In *ETRA*, 2021. 1

[2] Nuri Murat Arar, Hua Gao, and Jean-Philippe Thiran. Robust gaze estimation based on adaptive fusion of multiple cameras. In *FG*, volume 1, pages 1–7. IEEE, 2015. 1, 3

[3] Nuri Murat Arar and Jean-Philippe Thiran. Robust real-time multi-view eye tracking. *arXiv preprint arXiv:1711.05444*, 2017. 3

[4] Mihai Bace, Vincent Becker, Chenyang Wang, and Andreas Bulling. Combining gaze estimation and optical flow for pursuits interaction. In *Proc. ETRA*, 2020. 1

[5] Yiwei Bao, Yunfei Liu, Haofei Wang, and Feng Lu. Generalizing gaze estimation with rotation consistency. In *Proc. CVPR*, 2022. 2

[6] Zhaokang Chen and Bertram E. Shi. Appearance-based gaze estimation using dilated-convolutions. In C.V. Jawahar, Hongdong Li, Greg Mori, and Konrad Schindler, editors, *Proc. ACCV*, pages 309–324, 2018. 2

[7] Yihua Cheng, Yiwei Bao, and Feng Lu. Puregaze: Purifying gaze feature for generalizable gaze estimation. *Proc. AAAI*, 2022. 2, 5, 6

[8] Yihua Cheng, Shiyao Huang, Fei Wang, Chen Qian, and Feng Lu. A coarse-to-fine adaptive network for appearance-based gaze estimation. In *Proc. AAAI*, 2020. 2

[9] Yihua Cheng and Feng Lu. Gaze estimation using transformer. In *Proc. ICPR*, 2022. 2, 5, 6

[10] Yihua Cheng, Feng Lu, and Xucong Zhang. Appearance-based gaze estimation via evaluation-guided asymmetric regression. In *Proc. ECCV*, pages 105–121, 2018. 2

[11] Yihua Cheng, Haofei Wang, Yiwei Bao, and Feng Lu. Appearance-based gaze estimation with deep learning: A review and benchmark. *arXiv preprint arXiv:2104.12668*, 2021. 1

[12] 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. 2

[13] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. CVPR*, 2009. 4

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. ICLR*, 2021. 5

[15] S. M. Ali Eslami, Danilo Jimenez Rezende, Frederic Besse, Fabio Viola, Ari S. Morcos, Marta Garnelo, Avraham Ruderman, Andrei A. Rusu, Ivo Danihelka, Karol Gregor, David P. Reichert, Lars Buesing, Theophane Weber, Oriol Vinyals, Dan Rosenbaum, Neil Rabinowitz, Helen King, Chloe Hillier, Matt Botvinick, Daan Wierstra, Koray Kavukcuoglu, and Demis Hassabis. Neural scene representation and rendering. *Science*, 2018. 3

[16] Arya Farkhondeh, Cristina Palmero, Simone Scardapane, and Sergio Escalera. Towards self-supervised gaze estimation. *arXiv preprint arXiv:2203.10974*, 2022. 2

[17] Wenxin Feng, Jiangnan Zou, Andrew Kurauchi, Carlos H Morimoto, and Margrit Betke. Hgaze typing: Head-gesture assisted gaze typing. In *ETRA*, 2021. 1

[18] Tobias Fischer, Hyung Jin Chang, and Yiannis Demiris. RT-GENE: Real-Time Eye Gaze Estimation in Natural Environments. In *Proc. ECCV*, pages 339–357, 2018. 1

[19] 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. 1, 2

[20] Guy Gafni, Justus Thies, Michael Zollhöfer, and Matthias Nießner. Dynamic neural radiance fields for monocular 4d facial avatar reconstruction. In *Proc. CVPR*, 2021. 3

[21] Shreya Ghosh, Abhinav Dhall, Munawar Hayat, Jarrod Knibbe, and Qiang Ji. Automatic gaze analysis: A survey of deep learning based approaches. *arXiv preprint arXiv:2108.05479*, 2021. 1

[22] John Gideon, Shan Su, and Simon Stent. Unsupervised multi-view gaze representation learning. In *Proc. CVPRW*, pages 5001–5009, 2022. 3

[23] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proc. CVPR*, 2020. 3

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, 2016. 4, 5

[25] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proc. CVPR*, 2017. 5

[26] Petr Kellnhofer, Adria Recasens, Simon Stent, Wojciech Matusik, , and Antonio Torralba. Gaze360: Physically unconstrained gaze estimation in the wild. In *Proc. ICCV*, October 2019. 1, 2

[27] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. ICLR*, 2015. 5

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

[29] Kyle Krafka, Aditya Khosla, Petr Kellnhofer, Harini Kannan, Suchendra Bhandarkar, Wojciech Matusik, and Antonio Torralba. Eye tracking for everyone. In *Proc. CVPR*, 2016. 2

[30] Dongze Lian, Lina Hu, Weixin Luo, Yanyu Xu, Lixin Duan, Jingyi Yu, and Shenghua Gao. Multiview multitask gaze estimation with deep convolutional neural networks. *TNNLS*, 2019. 3, 5, 6, 7

[31] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *Proc. NIPS*, 2020. 3

[32] Yunfei Liu, Ruicong Liu, Haofei Wang, and Feng Lu. Generalizing gaze estimation with outlier-guided collaborative adaptation. In *Proc. ICCV*, 2021. 2

[33] Xinjun Ma, Yue Gong, Qirui Wang, Jingwei Huang, Lei Chen, and Fan Yu. Epp-mvsnet: Epipolar-assembling based depth prediction for multi-view stereo. In *Proc. ICCV*, 2021. 3

[34] Ricardo Martin-Brualla, Noha Radwan, Mehdi S. M. Sajjadi, Jonathan T. Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *Proc. CVPR*, 2021. 3

[35] Christopher D. McMurrough, Vangelis Metsis, Jonathan Rich, and Fillia Makedon. An eye tracking dataset for point of gaze detection. In *ETRA*, 2012. 1, 2

[36] Marko Mihajlovic, Aayush Bansal, Michael Zollhoefer, Siyu Tang, and Shunsuke Saito. KeypointNeRF: Generalizing image-based volumetric avatars using relative spatial encoding of keypoints. In *Proc. ECCV*, 2022. 3

[37] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 3

[38] Vinod Nair and Geoffrey E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proc. ICML*, 2010. 4

[39] Takehiko Ohno and Naoki Mukawa. A free-head, simple calibration, gaze tracking system that enables gaze-based interaction. In *Proc. ETRA*, pages 115–122, 2004. 1, 3

[40] Seonwook Park, Shalini De Mello, Pavlo Molchanov, Umar Iqbal, Otmar Hilliges, and Jan Kautz. Few-shot adaptive gaze estimation. In *Proc. ICCV*, 2019. 2, 5, 6

[41] Seonwook Park, Adrian Spurr, and Otmar Hilliges. Deep pictorial gaze estimation. In *Proc. ECCV*, 2018. 2

[42] Jiawei Qin, Takuru Shimoyama, and Yusuke Sugano. Learning-by-novel-view-synthesis for full-face appearance-based 3d gaze estimation. In *Proc. CVPRW*, 2022. 2, 5

[43] Ravikrishna Ruddarraju, Antonio Haro, Kris Nagel, Quan T Tran, Irfan A Essa, Gregory Abowd, and Elizabeth D Mynatt. Perceptual user interfaces using vision-based eye tracking. In *Proceedings of the 5th international conference on Multimodal interfaces*, pages 227–233, 2003. 1, 3

[44] Alessandro Ruzzi, Xiangwei Shi, Xi Wang, Gengyan Li, Shalini De Mello, Hyung Jin Chang, Xucong Zhang, and Otmar Hilliges. Gazenerf: 3d-aware gaze redirection with neural radiance fields. *arXiv preprint arXiv:2212.04823*, 2022. 2, 3

[45] Sheng-Wen Shih and Jin Liu. A novel approach to 3-d gaze tracking using stereo cameras. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(1):234–245, 2004. 1, 3

[46] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. Learning from simulated and unsupervised images through adversarial training. In *Proc. CVPR*, 2017. 2

[47] Leslie N. Smith. Cyclical learning rates for training neural networks. In *Proc. WACV*, 2017. 5

[48] Yusuke Sugano, Yasuyuki Matsushita, and Yoichi Sato. Learning-by-synthesis for appearance-based 3d gaze estimation. In *Proc. CVPR*, pages 1821–1828, 2014. 2

[49] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000. 7

[50] Akira Utsumi, Kotaro Okamoto, Norihiro Hagita, and Kazuhiro Takahashi. Gaze tracking in wide area using multiple camera observations. In *Proc. ETRA*, pages 273–276, 2012. 1, 3

[51] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proc. NIPS*, 2017. 5

[52] Jiayu Yang, Wei Mao, Jose M. Alvarez, and Miaomiao Liu. Cost volume pyramid based depth inference for multi-view stereo. In *Proc. CVPR*, 2020. 3

[53] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. *Proc. ECCV*, 2018. 3

[54] Pengwei Yin, Jiawu Dai, Jingjing Wang, Di Xie, and Shiliang Pu. Nerf-gaze: A head-eye redirection parametric model for gaze estimation. *arXiv preprint arXiv:2212.14710*, 2022. 2, 3

[55] Yu Yu, Gang Liu, and Jean-Marc Odobez. Improving few-shot user-specific gaze adaptation via gaze redirection synthesis. In *Proc. CVPR*, pages 11937–11946, 2019. 2

[56] Yu Yu and Jean-Marc Odobez. Unsupervised representation learning for gaze estimation. In *Proc. CVPR*, June 2020. 2

[57] 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 *Proc. ECCV*, 2020. 1, 2, 5

[58] Xucong Zhang, Seonwook Park, and Anna Maria Feit. *Eye Gaze Estimation and Its Applications*, pages 99–130. Springer International Publishing, 2021. 1

[59] Xucong Zhang, Yusuke Sugano, and Andreas Bulling. Revisiting data normalization for appearance-based gaze estimation. In *Proc. ETRA*, 2018. 2, 3, 4

[60] Xucong Zhang, Yusuke Sugano, and Andreas Bulling. Evaluation of appearance-based methods and implications for gaze-based applications. In *Proceedings of the 2019 CHI conference on human factors in computing systems*, pages 1–13, 2019. 3

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

[62] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. It's written all over your face: Full-face appearance-based gaze estimation. In *Proc. CVPRW*, 2017. 2

[63] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. Mpiigaze: Real-world dataset and deep appearance-based gaze estimation. *TPAMI*, 2019. 1, 2, 5

[64] Yufeng Zheng, Seonwook Park, Xucong Zhang, Shalini De Mello, and Otmar Hilliges. Self-learning transformations for improving gaze and head redirection. In *Proc. NIPS*, 2020. 2

[65] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. *Proc. AAAI*, 34(07):13001–13008, 2020. 4