SAPIENZA
UNIVERSITÀ DI ROMA

Towards Self-Supervised Gaze Estimation

Department of Computer, Control and Management Engineering
Corso di Laurea Magistrale in Data Science

Arya Farkhondeh
ID number 1860768

Advisors
Prof. Sergio Escalera
Prof. Simone Scardapane

Co-Advisor
Cristina Palmero

Academic Year 2021-2022

Thesis defended on 28 October 2022
in front of a Board of Examiners composed by:

Prof. Tardella Luca (chairman)
Prof. Battaglia Francesco
Prof. Brutti Pierpaolo
Prof. Daraio Cinzia
Prof. De Felice Massimo
Prof. Maggi Bernardo
Prof. Scardapane Simone

Towards Self-Supervised Gaze Estimation
Sapienza University of Rome

This thesis has been typeset by LaTeX and the Sapthesis class.

Author's email: farkhondeh.1860768@studenti.uniroma1.it

Abstract

Recent joint embedding-based self-supervised methods have surpassed standard supervised approaches on various image recognition tasks such as image classification. These self-supervised methods aim at maximizing agreement between features extracted from two differently transformed views of the same image, which results in learning an invariant representation with respect to appearance and geometric image transformations. However, the effectiveness of these approaches remains unclear in the context of gaze estimation, a structured regression task that requires equivariance under geometric transformations (e.g., rotations, horizontal flip). In this thesis, we propose SwAT, an equivariant version of the online clustering-based self-supervised approach SwAV, to learn more informative representations for gaze estimation. We identify the most effective image transformations for self-supervised pretraining and demonstrate that SwAT, with ResNet-50 and supported with uncurated unlabeled face images, outperforms state-of-the-art gaze estimation methods and supervised baselines in various experiments. In particular, we achieve up to 57% and 25% improvements in cross-dataset and within-dataset evaluation tasks on existing benchmarks (ETH-XGaze, Gaze360, and MPIIFaceGaze).

Acknowledgements

First, I would like to express my deepest appreciation to Prof. Sergio Escalera, Prof. Simone Scardapane, and Cristina Palmero for their dedicated guidance and support during this research. Special thanks to Computer Vision Center (CVC), Spain and HuPBA research group for allocating resources to my research. I am also grateful for the support of the faculty of Information Engineering, Computer Science, and Statistics, which provided me with a thesis abroad scholarship. Finally, I would like to thank my family and my friends for their continued support during my research and studies.

Contents

Chapter 1

Introduction

Eye gaze is a fundamental non-verbal signal in human communication and a powerful tool to infer attention and intention. Model-based and appearance-based are the main approaches to estimate gaze direction. Model-based approaches [18, 34, 49, 66] estimate gaze direction via constructing a geometric 3D model of the eye along with a specific personal calibration process. Such methods promise a high degree of accuracy in presence of a high-resolution image, thus they are typically found in head-mounted devices and fixed settings with dedicated lighting. However, they are less suitable for assessing gaze behavior in everyday scenarios, where a non-obtrusive solution capable of functioning in more challenging conditions is preferred. On the flip side, appearance-based approaches [45, 47] can be used with remote camera systems, which offer a trade-off between accuracy and usability, and can be applied in general everyday applications under real-world conditions. Appearance-based gaze estimation, consisting in regressing the gaze direction directly from face or eye images, is the preferred approach for remote camera scenarios. It enables many applications including human-computer interaction (HCI) [26, 50, 62], cognitive and behavioral understanding [43, 48], and autonomous driving [42]. However, appearance-based gaze estimation remains a non-trivial problem to solve within the computer vision field due to the large variability across appearance and geometric factors. Convolutional neural network (CNN) based methods [6, 10, 23, 25, 36, 39, 63] have achieved promising performances fueled by large-scale datasets [23, 25, 27, 60]. Nonetheless, there is still a large gap to achieve a desirable performance especially when it comes to generalizing to unseen distributions with novel head poses, appearances, geometry, and illuminations. One way to address this problem is through the acquisition of even larger in-the-wild, gaze-annotated datasets with more variability. However, collecting data with accurate gaze annotations is an unscalable and laborious process that requires controlled conditions, complicated setups, tedious camera calibration, and subject recruitment. An inexpensive solution is therefore needed to extend variability in terms of appearance and geometric factors.

Recently, joint embedding-based self-supervised methods, including contrastive and non-contrastive, have obtained remarkable accuracy on various vision tasks, such as image classification [2–5, 17], object detection [54], and hand-pose estimation [44]. These approaches have proven successful at learning generalizable features by leveraging large-scale unlabeled data [22, 30].

Similarly, these methods could leverage the vast amount of unlabeled face images that are publicly available on the Internet to learn useful representations for appearance-based gaze estimation. However, no attention has been paid to investigate their effectiveness for the gaze estimation task to the best of our knowledge. Therefore,

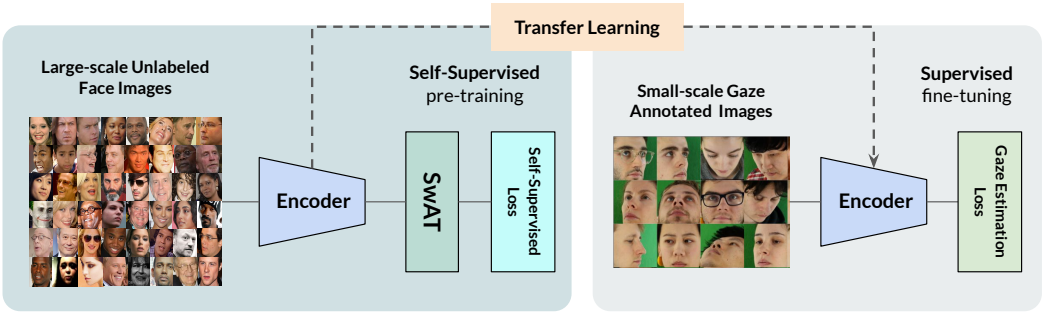

Figure 1.1. Global view of our approach. In the first stage, we pre-train an encoder via an online clustering approach on a large-scale set of unlabeled face images while encouraging equivariance through our proposed method (SwAT). In the second stage, we transfer the learned knowledge from the first stage and fine-tune on a small-scale set of gaze-annotated images.

the main goal of this work is to explore the efficacy of a self-supervised approach in the context of gaze estimation to reduce the reliance on large-scale gaze-annotated data that is laborious to acquire.

In a nutshell, self-supervised learning aims at solving a pretext task to learn a useful representation. The representation is then used in downstream tasks via transfer learning. The common pretext task among (non-)contrastive self-supervised methods (e.g., SimCLR [4], MoCo [20], SwAV [3], BYOL [17], and VICReg [2]) is to enforce consistency between features extracted from two differently transformed views of the same image. As a result, the feature extractor is encouraged to learn an invariant representation with respect to the image-space transformations, such as appearance (e.g., color jitter) and geometric (e.g., horizontal flip). Although invariance might be a desired property for most image recognition tasks, the structured regression task of gaze estimation requires equivariance under geometric transformations. In fact, applying geometric transformations to a face/eye image results in respective changes in gaze direction. Thus, in this work, our goal is to learn an equivariant representation under geometric transformations to align with our downstream task of interest i.e., gaze estimation. By definition, a representation is equivariant with respect to an input image transformation when the transformation is reflected in the representation output, whereas in the invariance scenario, the transformation is not transferred to the representation output.

In this thesis, we propose **Sw**apping **A**ffine **T**ransformations (**SwAT**), a novel method to achieve the desired property of equivariance. It can be thought of as a plug-and-play method that can be added to any joint embedding-based self-supervised approach. As Fig. 1.1 depicts, we perform self-supervised pretraining on large-scale unlabeled face images while encouraging equivariance through SwAT. Then, we transfer the learned knowledge to the downstream gaze estimation task and finetune with gaze labels. More precisely, we first extract two views of the same face image by applying appearance and geometric transformations. Then we map the transformed views to embedding vectors using an encoder. SwAT encourages equivariance via equalization of geometric transformations in feature space, performed by a feature transform layer. The consistency between equalized feature vectors is then enforced via a non-contrastive online clustering-based approach called SwAV [3]. Intuitively, SwAT allows the feature extractor to transfer the image-space geometric transformation to the representation output which preserves the intrinsic structure

of the transformations.

Our proposed self-supervised approach potentially deconcentrates research in gaze estimation from the non-trivial process of large-scale annotated data collection towards effectively leveraging widely available large-scale unlabeled data. More importantly, leveraging such unlabeled data with more variety enhances the generalizability of gaze estimation models upon novel distributions. We show that the equivariance property provided by SwAT leads to learning better representations for gaze estimation, compared to other pretraining regimes. In addition, we show that the unsupervised features provided by SwAT surpass the commonly used ImageNet supervised features in gaze estimation. We perform extensive experiments to verify the effectiveness of our approach under various challenging evaluation settings. In particular, we demonstrate that SwAT outperforms the supervised baselines in low-data regimes where only a few annotations (10% and 30%) are available. Supported with unlabeled data, SwAT achieves state-of-the-art results on existing benchmarks and improves the supervised baselines for cross- and within- dataset evaluation tasks by 57% and 25%, respectively. In summary, our main contributions are:

- As far as we know, this work is the first that systematically explores the effectiveness of a self-supervised approach for full-face appearance-based gaze estimation.
- We extend a self-supervised approach with equivariance to learn a more informative representation for gaze estimation.
- We identify the top-performing transformations for self-supervision and propose an effective policy to compose them.
- Our proposed self-supervised approach outperforms supervised baselines in the semi-supervised setting where only a few annotations are available.
- We perform extensive evaluations that show that our approach achieves state-of-the-art results on the existing benchmarks.

Chapter 2

Related Work

2.1 Self-Supervised Learning

Self-supervised learning aims at learning informative representations without relying on manual annotations such that the reliance of downstream tasks on large-scale data is diminished. Early self-supervised approaches attempted to learn useful representations from unlabeled data via solving handcrafted pretext tasks such as Jigsaw puzzle [35], colorization [59], transformation prediction [1, 16], and inpainting [41]. More recently, contrastive-based methods [4, 20, 33] have achieved notable results on various computer vision tasks such as image classification. However, these methods are inherently computationally inefficient as they require pairwise contrasts with a large set of negative examples. Consequently, non-contrastive approaches [2, 3, 5, 17] are receiving special attention. Clustering-based approaches such as SwAV [3] discriminate between groups of images with similar features instead of individual images while achieving state-of-the-art results in image classification on ImageNet. However, both contrastive and non-contrastive methods are designed to learn invariant representations under image transformations, while gaze estimation requires equivariance under geometric transformations. Hence, in this work, we extend SwAV [3] via introducing equivariance under geometric transformations.

2.2 Equivariance in Self-Supervised Learning

Equivariance in self-supervised learning is starting to attract attention [13, 44, 55]. Despite their proven effectiveness, these methods bear some limitations that do not align with our assumptions. While our goal is to promote equivariance for multiple affine transformations, Dangovski et al. [13]'s work is limited to a single transformation and Xie et al. [55]'s method is not scalable as the number of transformations increments. Most similarly, Spurr et al. [44] propose an equivariance formulation for the task of 3D hand-pose estimation. However, their equivariance formulation together with a contrastive loss explicitly pushes apart the pseudo-negative pairs that may include faces with similar affine information, gaze, and head directions.

2.3 Appearance-based Gaze Estimation

Recent progress in appearance-based gaze estimation has been mainly achieved via collecting large-scale datasets [23, 25, 27, 60], task-specific tailored architectures [6, 9,

12,39], and data normalization methods [61,65]. Nevertheless, appearance-based gaze estimation still suffers from performance degradation when it comes to distribution-shift. To overcome this problem, various variants of domain adaptation approaches have been proposed [7,8,19,28,29,31,38,51,57] to adapt a source domain to a target domain. Apart from supervised gaze estimation, weakly-supervised and unsupervised methods have started to receive more attention in gaze estimation. Kothari et al. [24] propose a weakly-supervised approach based on videos of people looking at each other. MTGLS [15] utilizes off-the-shelf models to obtain pseudo labels for unlabeled eye images in order to learn a gaze representation. Recent generative-based unsupervised gaze estimation approaches [46,58] make use of unlabeled eye images to learn gaze representations. Nevertheless, these approaches have limitations as they require supervision in the form of paired eye images of the same person [46,58] with similar head-pose [58]. Wu et al. [53] employ self-supervision as an auxiliary task for supervised gaze estimation. Unlike the previous methods, we pretrain a standard CNN architecture for gaze estimation in a self-supervised fashion via leveraging large-scale unlabeled face images. Our approach is less complex while more scalable as it does not make any assumption on the kind of unlabeled data and does not require multiple auxiliary losses for training as in [46,58]. Furthermore, in contrast to previous unsupervised works that use eye images, we use full-face images, which have been proven to contain useful auxiliary information (e.g., head-pose, geometric features) for gaze estimation [25,37,64].

Chapter 3

Method

To learn an unsupervised representation for the gaze estimation task (Sec. 3.1), as Fig. 1.1 depicts, our goal is to pretrain an encoder on large-scale unlabeled face images using a self-supervised approach (Sec. 3.2) while encouraging equivariance via SwAT (Sec. 3.3). Afterward, we transfer the knowledge to the gaze estimation task via supervised fine-tuning (Sec. 3.4).

3.1 Supervised Gaze Estimation

Gaze estimation is a regression task that aims at learning a mapping function $\mathcal{H} : \boldsymbol{x} \rightarrow \boldsymbol{g}$ that maps the high-dimensional RGB images $\boldsymbol{x} \in \mathbb{R}^{H \times W \times 3}$ to low-dimensional 2D angles $\boldsymbol{g} \in \mathbb{R}^2$ i.e., yaw and pitch. The 2D angles are a compact representation of the 3D gaze direction vector in the camera coordinate system, the origin of which is the center of the face or the midpoint between the eyes, depending on the dataset. $\mathcal{H}$ is a parameterized function, composed of a backbone encoder (e.g., ResNet) as well as a linear head (e.g., MLP). Given image and gaze vector pairs $\{\boldsymbol{x}_i, \boldsymbol{g}_i\}_{i=1}^{N}$, we minimize the following loss function (L_{gaze}) to train $\mathcal{H}$,

$$\mathcal{L}_{\text{gaze}} = \frac{1}{N} \sum_{i=1}^{N} ||\boldsymbol{g}_i - \hat{\boldsymbol{g}}_i||_1, \tag{3.1}$$

where N is the number of samples and $\hat{g}_i = \mathcal{H}(\boldsymbol{x})$.

3.2 Self-Supervised Pretraining

In this work, as the pretext task, we aim at maximizing the mutual information between the features from two different views of the same image. As shown in Fig. 3.1 (left), two differently transformed views of an image $\boldsymbol{x}$ are computed via applying two different sets of transformations i.e., $\boldsymbol{x}_1 = t_1(\boldsymbol{x})$ and $\boldsymbol{x}_2 = t_2(\boldsymbol{x})$ where, $t_1 \sim \mathcal{T}$ and $t_2 \sim \mathcal{T}$ are sampled from the same transformation catalog $\mathcal{T}$. An encoder $f_\phi(.)$ parameterized by ϕ is used to map the transformed views to vector representations, $\boldsymbol{z}_1 = f_\phi(\boldsymbol{x}_1)$ and $\boldsymbol{z}_2 = f_\phi(\boldsymbol{x}_2)$. The encoder $f_\phi(.)$ is composed of a backbone (e.g., ResNet) and a projection head (e.g., MLP). Then, we maximize agreement between the feature vectors $Max(\boldsymbol{z}_1, \boldsymbol{z}_2)$ using an online clustering-based self-supervised approach called SwAV [3]. SwAV enforces agreement using intermediate cluster assignments computed in an online fashion, where the cluster assignments are treated as the targets to predict from feature vectors. To compute the cluster assignments $\boldsymbol{c}_1$

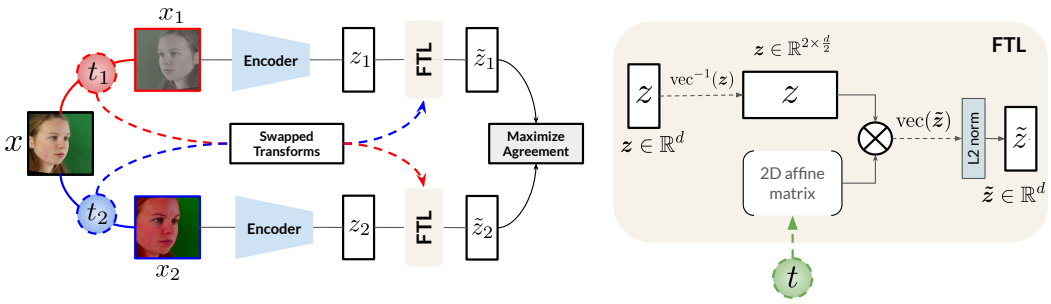

Figure 3.1. Left. Two differently transformed versions of the same image $\boldsymbol{x}$ are obtained via applying two different sets of transformations i.e., $\boldsymbol{x}_1 = t_1(\boldsymbol{x})$ and $\boldsymbol{x}_2 = t_2(\boldsymbol{x})$. An encoder is used to map the transformed views to vector representations, $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$. To achieve equivariance, we equalize $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ in terms of affine information. To do so, we swap the affine transformations applied in image space t_1 and t_2, then we use the feature transform layer (FTL) to apply the swapped transformations to the feature vectors i.e., $\tilde{\boldsymbol{z}}_1 = \mathbb{FTL}(t_2, \boldsymbol{z}_1)$ and $\tilde{\boldsymbol{z}}_2 = \mathbb{FTL}(t_1, \boldsymbol{z}_2)$. Then, we maximize agreement between the resulting feature vectors, $\tilde{\boldsymbol{z}}_1$ and $\tilde{\boldsymbol{z}}_2$. **Right.** Details of the feature transform layer (FTL). $\text{vec}^{-1}(\boldsymbol{z})$ transforms $\boldsymbol{z}$ from 1D to 2D in order to enable matrix-matrix multiplication with the 2D affine matrix, resulting in $\tilde{\boldsymbol{z}}$. Then, $\text{vec}(\tilde{\boldsymbol{z}})$ transforms back $\tilde{\boldsymbol{z}}$ from 2D to 1D. L2-norm is then applied.

and $\boldsymbol{c}_2$, the vector representations ($\boldsymbol{z}_1$ and $\boldsymbol{z}_2$) are compared to a set of M learnable prototype vectors $\boldsymbol{P}_\psi = \{\boldsymbol{p}_1, ..., \boldsymbol{p}_M\}$, parameterized by ψ. Maximizing agreement is achieved via swapping the computed cluster assignments and predicting them using feature vectors. The idea is to predict the cluster assignment $\boldsymbol{c}_1$ from the feature $\boldsymbol{z}_2$, and $\boldsymbol{c}_2$ from $\boldsymbol{z}_1$. Intuitively, if two feature vectors contain mutual information then it should be possible to predict the cluster assignment $\boldsymbol{c}_1$ ($\boldsymbol{c}_2$) from the other feature $\boldsymbol{z}_2$ ($\boldsymbol{z}_1$). The self-supervised loss function is as follows:

$$\mathcal{L}_{\text{SwAV}} = \ell(\boldsymbol{z}_1, \boldsymbol{c}_2) + \ell(\boldsymbol{z}_2, \boldsymbol{c}_1), \tag{3.2}$$

where $\ell(\boldsymbol{z}, \boldsymbol{c})$ is the cross entropy loss between the cluster assignments and the probability computed by applying softmax to the dot products of $\boldsymbol{z}_i$ and prototypes ($\boldsymbol{P}_\psi$), as in Eq. 3.3. The cross entropy loss measures agreement between a feature and cluster assignment. $\ell(\boldsymbol{z}_i, \boldsymbol{c}_j)$ is defined as follows:

$$\ell(\boldsymbol{z}_i, \boldsymbol{c}_j) = -\sum_m \boldsymbol{c}_j^{(m)} \log\left(\frac{\exp(\frac{1}{\tau}\boldsymbol{z}_i{}^\top \boldsymbol{p}_m)}{\sum_{m'}\exp(\frac{1}{\tau}\boldsymbol{z}_i^\top \boldsymbol{p}_{m'})}\right), \tag{3.3}$$

where τ is a temperature parameter and m denotes the mth prototype. The overall loss function (Eq. 3.2) is minimized with respect to both parameters of the encoder ϕ and trainable prototypes ψ. The method is online since only the features within a batch are used to compute the cluster assignments. To avoid trivial solutions i.e., assigning the same cluster for every image within a batch, score adjustment is performed using an optimal transport algorithm, namely Sinkhorn-Knopp [11]. It encourages equipartition guaranteeing that the cluster assignments are distinct for images within a batch.

3.3 Equivariant Representation Learning

Similar to other (non-)contrastive self-supervised approaches, the SwAV formulation (Sec. 3.2) encourages invariance under appearance and geometric transformations. In image recognition tasks such as image classification, applying geometric transformations (t^g) to an image does not change the label. However, in the gaze estimation task, applying geometric transformations in image space results in respective changes in label space. Thus, instead of learning an *invariant* representation, we aim at learning an *equivariant* representation.

Definition (Equivariance) *A mapping function $f_\phi : \boldsymbol{x} \to \boldsymbol{z}$ is said to be equivariant with respect to image-space transformation t_I^g when mapping the transformed input image, $f_\phi(t_I^g(\boldsymbol{x}))$, produces the same result as transforming the vector representation of the input image, i.e., $t_F^g(f_\phi(\boldsymbol{x}))$:*

$$f_\phi(t_I^g(\boldsymbol{x})) = t_F^g(f_\phi(\boldsymbol{x})), \tag{3.4}$$

where transformations t_I^g and t_F^g are used to apply the same transformation in different spaces i.e., image space and feature space, respectively. Intuitively, *equivariance* property enables f_ϕ to learn a direct relationship between image space and feature space, thereby preserving the intrinsic structure of the transformations [52].

Swapping Affine Transformations. Eq. 3.2 enforces consistent mapping between two transformed views via intermediate cluster assignments. Abstractly, it aims to maximize the mutual information between the features from two views. Thus, ideally,

$$f_\phi(t_1^g(\boldsymbol{x})) = f_\phi(t_2^g(\boldsymbol{x})). \tag{3.5}$$

The only way that the above equality is satisfied is through encouraging f_ϕ to be invariant with respect to the applied geometric transformations t_1^g and t_2^g. Instead, to let the mapping function f_ϕ be equivariant under affine transformations applied in image space, we propose the **Sw**apping **A**ffine **T**ransformations (**SwAT**) method. SwAT achieves equivariance via equalization of vector representations in terms of applied image-space affine transformations. To achieve that, as in Eq. 3.6 and Fig. 3.1, we swap the affine transformations applied in image-space, and then we apply them in feature-space via a feature transform layer ($\mathbb{FTL}$), detailed later. Thus,

$$\tilde{\boldsymbol{z}}_1 = \mathbb{FTL}(t_2^g, \boldsymbol{z}_1), \quad \tilde{\boldsymbol{z}}_2 = \mathbb{FTL}(t_1^g, \boldsymbol{z}_2). \tag{3.6}$$

Intuitively, $\tilde{\boldsymbol{z}}_1$ and $\tilde{\boldsymbol{z}}_2$ contain the same affine transformation information. Thus, enforcing consistency between transformation-equalized vector representations prevents f_ϕ from becoming invariant with respect to transformations. In contrast, since unequalized vector representations $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ contain different transformation information, enforcing consistency would result in invariance as in Eq. 3.5. The self-supervised loss (Eq. 3.2) becomes:

$$\mathcal{L}_{\text{SwAT}} = \ell(\tilde{\boldsymbol{z}}_1, \tilde{\boldsymbol{c}}_2) + \ell(\tilde{\boldsymbol{z}}_2, \tilde{\boldsymbol{c}}_1), \tag{3.7}$$

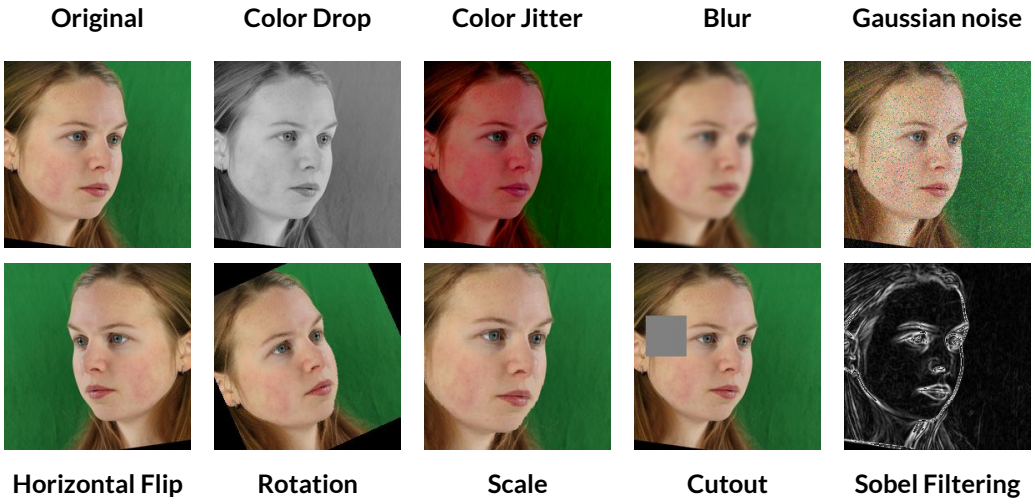

Figure 3.2. Transformation Catalog. The appearance and geometric transformations explored in this work for self-supervised representation learning.

where,

$$\tilde{\boldsymbol{c}}_2 = \tilde{\boldsymbol{z}}_2 \, \boldsymbol{P}_\psi, \quad \tilde{\boldsymbol{z}}_2 \in \mathbb{R}^d, \quad \boldsymbol{P}_\psi \in \mathbb{R}^{d \times \mathrm{M}}. \tag{3.8}$$

Feature Transform Layer. As Fig. 3.1 (right) depicts, to be able to apply the feature-space equivalent (t_F^g) of the image-space transformation (t_I^g), we introduce a non-trainable feature transform layer ($\mathbb{FTL}$). This layer takes as input the 2D affine transformation matrix $\boldsymbol{T}_\theta$ (e.g., 2D rotation matrix with angle θ) and 1D feature vector $\boldsymbol{z}$. It first transforms $\boldsymbol{z}$ from 1D to 2D via an inverse vectorization, $\mathrm{vec}^{-1}_{2\times k}(\boldsymbol{z})$, where $k = \frac{d}{2}$ and d is the dimensionality of the projection head. Afterward, it performs a matrix-matrix multiplication, resulting in $\tilde{\boldsymbol{z}}$. Finally, $\tilde{\boldsymbol{z}}$ is transformed back to a 1D feature vector via a vectorization, $\mathrm{vec}(\tilde{\boldsymbol{z}})$, and then L2-norm is applied.

Transformations. Fig. 3.2 shows the explored transformations in this work which fall into two groups, namely appearance and geometric transformations. For appearance transformation, we consider color drop, color jitter, Gaussian blur, Gaussian noise, cutout, and Sobel filtering. As geometric transformations, we examine horizontal flip, rotation, and scale. In the context of gaze estimation, appearance and scale transformations do not change the 3D gaze direction label with respect to the camera coordinate system. In contrast, applying horizontal flip and rotation in image space results in respective changes in label space. As an example, for a horizontally flipped view of an image, the sign of the yaw angle is reversed. Thus, for our proposed SwAT method, we only swap horizontal flip and rotation transformations.

3.4 Fine-tuning for Gaze Estimation

To perform gaze estimation, we first initialize the weights of the backbone network with the pretrained weights previously learned through self-supervised pretraining. Then, the whole network is finetuned with gaze-annotated data using the L1 loss

between the estimated angles $\hat{\boldsymbol{g}} = \mathcal{H}(x)$ and actual angles g, as follows:

$$\mathcal{L}_{\text{gaze}} = \frac{1}{N} \sum_{i=1}^{N} ||\boldsymbol{g}_i - \hat{\boldsymbol{g}}_i||_1, \tag{3.9}$$

where N is the number of samples.

Chapter 4

Implementation details

In this chapter, we provide the full implementation details. The datasets used in this thesis, the default implementation details, and the experimental protocol are presented in Sec. 4.1. Supplementary details of the pretraining stage can be found in Sec. 4.2. Details of evaluation settings namely, linear evaluation (Sec. 4.3), semi-supervised setting (Sec. 4.4), and supervised fine-tuning (Sec. 4.5) are explained. Details of the evaluation protocol on each dataset are provided in Sec. 4.6. Lastly, details of explored transformations in this thesis can be found in Sec. 4.7. Throughout all the experiments related to linear evaluation (Sec. 4.3), semi-supervised setting (Sec. 4.4), and supervised fine-tuning (Sec. 4.5), we use Adam as the optimizer, batch size of 512, an input size of 224×224 pixels, and a learning rate decay factor of 0.1 unless otherwise stated.

4.1 Experimental Setting

Datasets. For the self-supervised pretraining stage, we use a curated dataset i.e., ETH-XGaze [60] without labels. It contains 756,540 images and 80 subjects for training, captured under controlled laboratory conditions. Since ETH-XGaze is specifically collected for the task of gaze estimation under controlled conditions, it is unclear whether the quality of unsupervised features remains the same while using an uncurated dataset. To shed light on this, we also use the VGG-Face dataset [40] for pretraining. VGG-Face is collected from the web, including 2,622 identities and about 1.5 M face images. For the fine-tuning phase, throughout various experiments, we use other publicly available datasets in addition to ETH-XGaze, such as Gaze360 [23] and MPIIFaceGaze [64]. Gaze360 is a physically unconstrained dataset consisting of 238 subjects collected in indoor and outdoor environments with a wide range of head poses. MPIIFaceGaze is a subset of the MPIIGaze [63] dataset that contains 15 subjects and 3000 samples per subject, recorded while doing activities on the laptop.

Implementation Details. For the pretraining phase, we use SGD + LARS [56] optimizer with a batch size of 1024 distributed over 8 NVIDIA GeForce RTX 3090 GPUs. We pretrain for 100 epochs and experimentally found it to be sufficient. We use a weight decay of 10^{-6} and the learning rate is set to 0.45 followed by an initial linear warmup stage for 10 epochs. Afterward, we use cosine learning rate decay [32] with a final value of 0.00045. As the encoder, we use ResNet [21] and the projection head consists of a 2-layer MLP that maps the encoder output to 128-D. We experimentally set the number of prototypes M to 500.

Experimental protocol. We use the dataset partitions provided by each dataset. A prior data normalization stage is commonly applied by creating a virtual camera with fixed intrinsic and extrinsic camera parameters, which reduces head pose variability and hence the training space [61]. However, this normalization may conceal the benefits of enforcing equivariance for geometric transformations, especially for already constrained datasets with little geometric variability. Furthermore, this stage cannot be applied accurately if camera parameters are not provided. Therefore, for the finetuning part of our methods (baselines and SwAT) we apply data normalization only to ETH-XGaze, since its test evaluation assumes normalized data, and to MPIIFaceGaze, to compare against previous approaches that performed the normalization stage. We also evaluate the unnormalized version of MPIIFaceGaze (referred to as MPIIFaceGaze*) to better quantify the benefits of SwAT and compare its performance against the normalized counterpart. Throughout this work, we use average angular gaze error in degrees to measure performance. Hence, the 2D angles in the spherical coordinate system are transformed back to 3D gaze direction vectors in the camera coordinate system.

4.2 Supplementary details of pretraining

As mentioned in Sec. 3.2, SwAV [3] performs score adjustment using the Sinkhorn-Knopp [11] algorithm to avoid trivial solutions. We refer the reader to the SwAV paper [3] for the details of the Sinkhorn-Knopp algorithm. This algorithm has two hyperparameters, namely, the number of iterations and Sinkhorn regularization parameter (ϵ). We perform 3 Sinkhorn iterations as in SwAV and set $\epsilon = 0.03$. Note that a high value of ϵ leads to trivial solutions, i.e., same cluster assignment for every image within a batch, whereas a too low value results in numerical instability.

4.3 Implementation details of linear evaluation

For linear evaluation (Sec. 5.1 and Sec. 5.2), we freeze the backbone and train a linear regressor on top for 100 epochs. We set the initial learning rate to 0.01, which is decayed using cosine decay with a final value of 0.0001. We also used a weight decay of 0.0001.

4.4 Implementation details of semi-supervised learning

In semi-supervised learning (Sec. 5.3), we finetune the whole network using two subsets (10% and 30%) from the ETH-XGaze dataset, at the subject level. We finetune SwAT for 100 epochs with an initial learning rate of 0.001 for the backbone and 0.01 for the linear regression head. Then, we decay the learning rates after 40 and 80 epochs. We also used a weight decay of 0.0001. The supervised baseline is trained in the same manner except we initialize the learning rate of the backbone with 0.01.

4.5 Implementation details of supervised fine-tuning

This section provides implementation details of Sec. 5.4. For supervised fine-tuning, we use different hyperparameters for each dataset. In the case of ETH-XGaze, we finetune SwAT for 25 epochs following [60]. The learning rates of both the

Method	Color	Blur	Noise	Flip	Rotate	Scale	Cutout	Sobel
SwA**V**	1.0	0.7	0.8	0.0	0.4	0.6	0.5	0.9
SwA**T**	1.0	0.5	0.6	0.6	0.4	0.3	0.0	0.8

Table 4.1. Computed probabilities (p) using the soft assignment policy.

backbone and linear regressor are set to 0.001, which are then decayed at epoch 15. In addition, we use a weight decay of 0.0001. However, we train the supervised counterpart for 100 epochs with an initial learning rate of 0.01, decayed after 40 and 80 epochs. On Gaze360, we finetune SwAT for 80 epochs following [23]. The learning rate of backbone and the linear head are set to 0.001 and 0.01, respectively, which are decayed using cosine decay with a final value of 0.0001. In the case of MPIIFaceGaze, we perform fine-tuning for 40 epochs with an initial learning rate of 0.0005, decayed at 20 and 30 epochs. For MPIIFaceGaze*, we finetune for 25 epochs, decaying the learning rate at 10 and 20 epochs. For all datasets, we use horizontal flip and scaling $s \in [0.7, 1.4]$ as data augmentation.

4.6 Details of evaluation protocol on each dataset

ETH-XGaze contains 756K images and 80 subjects for training. The test set is composed of 15 subjects with a total of 159K samples. Since the dataset does not have an official validation set, we manually split the training set into two subject-independent sets, i.e., 90% (72 subjects) training set and 10% (8 subjects). We selected the subjects via visual inspection ensuring diversity across gender, ethnicity, and eyewear accessories. The validation set was only used for ablation study, evaluation of transformations, and evaluating the unsupervised features. The rest of the experiments are trained using 100% of the training data and evaluated with the test set. Note that the test set of ETH-XGaze is kept private and online evaluation is performed via the dedicated submission webpage. For semi-supervised learning, we selected two subsets from the training data at subject level, i.e., 10% (8 subjects) and 30% (24 subjects). The ID of the subjects are as follows:

- **10% subset (8 subjects)** = {3, 32, 48, 52, 80, 88, 101, 109}
- **30% subset (24 subjects)** = {0, 3, 8, 9, 13, 24, 28, 32, 33, 36, 38, 40, 45, 48, 52, 62, 79, 80, 88, 92, 101, 103, 109, 111}

Gaze360 contains 129K training, 17K validation, and 26K test samples collected from 238 subjects. We use a subset of the dataset whose faces come with bounding boxes, resulting in around 85K, 11K, and 16K samples for training, validation, and test, respectively.

MPIIFaceGaze comes with 45K samples and 15 subjects each having 3K samples. We perform a leave-one-person-out cross-validation for each subject to evaluate and compare the methods.

MPIIFaceGaze* is the unnormalized version of the MPIIFaceGaze dataset. We performed 3-fold cross-validation where the folds were chosen uniformly at random.

4.7 Details of Transformations

In this section, we provide the details of transformations for self-supervised pretraining (Sec. 5.1). When composed together, each transformation is applied with probability p_t, which is determined by soft assignment policy. The computed probabilities for each transformation depending on the pretraining approach (SwAV or SwAT) can be found in Tab. 4.1. In the following, we provide the details of each transformation in the same order they are applied during implementation.

Sobel. Since the two transformed views are assumed to be different, we apply the Sobel filter to only one view.

Blur. Gaussian blur is applied using a Gaussian kernel where we randomly sample the radius $\sigma \in [0.1, 2.0]$. We do not apply the blur transformation to views with Sobel transformation applied.

Color. We apply color transformation following SimCLR [4]. More concretely, this transformation is composed of two sub-transformations, i.e., color jittering (brightness, contrast, saturation, and hue) and color dropping (grayscale). We randomly apply color jittering with probability of 0.8, and color dropping with 0.2. We do not apply the color transformation to views with Sobel transformation applied.

Noise. We add Gaussian noise $N \sim (0, 30)$ to an image. Once Sobel is applied, we do not apply Gaussian noise.

Cutout. We randomly cutout a patch of size $h \times h$ pixels, where $h = 64 \times (H_{\boldsymbol{x}}/224)$ and $H_{\boldsymbol{x}}$ is the height (or width) of the input image ($\boldsymbol{x}$).

Flip. Applying horizontal flip to both views results in the same image. Thus, we only applied it to one view.

Rotate. We apply rotation via randomly sampling the rotation angle (θ) from $\theta \in [-45, 45]$ degrees.

Scale. Scaling (s) is applied via randomly sampling the scale factor $s \in [0.7, 1.4]$.

Chapter 5

Experiments and Results

In this chapter, we assess the usefulness of the image transformations considered in this thesis for self-supervised learning (Sec. 5.1), and the performance of the proposed SwAT method through an exhaustive experimental evaluation. In particular, we first compare SwAT to other pretraining schemes to determine the utility of the equivariance property (Sec. 5.2). We then evaluate SwAT under low-data regimes (Sec. 5.3), and compare it to state-of-the-art approaches for within- (Sec. 5.4) and cross-dataset (Sec. 5.5) settings. We analyze the accuracy of SwAT as a function of gaze direction and head pose (Sec. 5.6). We quantify the equivariance capability, showing the relative improvements made by SwAT over SwAV (Sec. 5.7). Then, we visualize the estimated gaze directions of SwAT and supervised baseline (Sec. 5.8). We provide the ablation studies of key hyperparamters such as number of prototypes (Sec. 5.9.1), number of pretraining epochs (Sec. 5.9.2), and the dimensionality of projection-head (Sec. 5.9.3). Lastly, we compare our equivariance solution to PeCLR [44] (Sec. 5.10).

5.1 Evaluation of Transformations

Fig. 3.2 shows the studied transformations in this work. To identify the most effective transformations, we perform individual transformation evaluation. To do so, we pretrain an encoder on the ETH-XGaze dataset (without labels) using each individual transformation. Then, we freeze the backbone and train a linear gaze regressor on top. For this experiment, we use ResNet-50 as the backbone and we set the input size to 112×112. We manually create a validation set from the training set of ETH-XGaze by splitting the data into two subject-independent sets i.e., 90% training set and 10% validation set.

Individual Transformation Evaluation. Tab. 5.1 shows the results of individual transformations for SwAV and SwAT methods. Note that both methods behave identically under appearance and scale transformations, whereas they differ in terms of horizontal flip and rotation. As can be seen, SwAT outperforms SwAV in the case of horizontal flip and rotation, achieving around 15% and 5% relative improvements, respectively. This demonstrates the benefit of enforcing equivariance under affine transformations via SwAT, producing feature representations that are more aligned with the gaze estimation task.

Composition of Transformations. A stronger image distortion can be realized via composing multiple transformations in a sequential manner. To achieve that, we

Method	Color	Blur	Noise	Flip	Rotate	Scale	Cutout	Sobel	Composition
SwA**V**	27.1	28.9	28.4	33.8	30.8	29.7	30.7	27.7	26.4
SwA**T**	27.1	28.9	28.4	**28.6**	**29.2**	29.7	30.7	27.7	**26.0**

Table 5.1. Evaluation of Transformations. Performance of SwAV and SwAT for each individual transformation on the validation set of ETH-XGaze, in terms of average angular gaze error (degrees). Note that SwAV and SwAT only differ in terms of Flip and Rotation while behaving identically in the case of other transformations. The last column shows the results of composition of transformations using the soft assignment policy.

compose the transformations using a soft assignment policy. Let us denote p as the probability of applying a transformation. We compute p by mapping the individual performances (Tab. 5.1) to [0,1] via scaling, such as:

$$p_t = 1 - \frac{\mathrm{e}_t - \mathrm{e}_{min}}{\mathrm{e}_{max} - \mathrm{e}_{min}}, \tag{5.1}$$

where t is a given transformation chosen from the transformation catalog i.e., $t \in \{\text{color, blur, ..., sobel}\}$, e_t corresponds to the gaze error of the transformation during individual transformation evaluation, and e_{max} and e_{min} are the maximum and minimum gaze error across all the individual transformations of the method i.e., SwAV and SwAT (rows of Tab. 5.1). This way, all the transformations contribute to data augmentation with respect to their individual performances. The last column of Tab. 5.1 (*Composition*) shows that the soft assignment policy improves the performance compared to individual transformations. We find such a soft assignment policy more promising than a hard assignment counterpart such as selecting top-k transformations and then performing an exhaustive search as in [44]. In particular, we get 0.9° improvement using the soft assignment policy compared to the best composition of the hard assignment method.

5.2 Evaluating the Unsupervised Features

After assessing the effectiveness of each individual transformation and finding an optimal composition for SwAV and SwAT (Sec. 5.1)), we evaluate the quality of unsupervised features. More precisely, the goal of this experiment is twofold: to explore whether the equivariance property provided by SwA**T** leads to a better representation compared to the invariance counterpart (SwA**V**), and to shed light on the quality of the unsupervised features with the curated (ETH-XGaze) and uncurated datasets (VGG-Face), used for pretraining. To do so, we perform a linear evaluation, where we freeze the backbone (ResNet-50) after pretraining and train a linear gaze regressor on top. Then, we measure the performance on the validation set that we manually create by splitting the available ETH-XGaze training set intro training and validation sets. We also compare the unsupervised features with ImageNet supervised features, which are widely used in current gaze estimation works as initialization.

Fig. 5.1 (left) shows the results of the linear evaluation on the validation set of ETH-XGaze. We can see that SwAT outperforms SwAV with both curated (ETH-XGaze) and uncurated (VGG-Face) datasets. More importantly, SwAT surpasses the supervised features pretrained on ImageNet, decreasing the gaze error from 22.8° to 20.6°. In the next experiments, we focus on comparing and evaluating SwAT in presence of labels for finetuning.

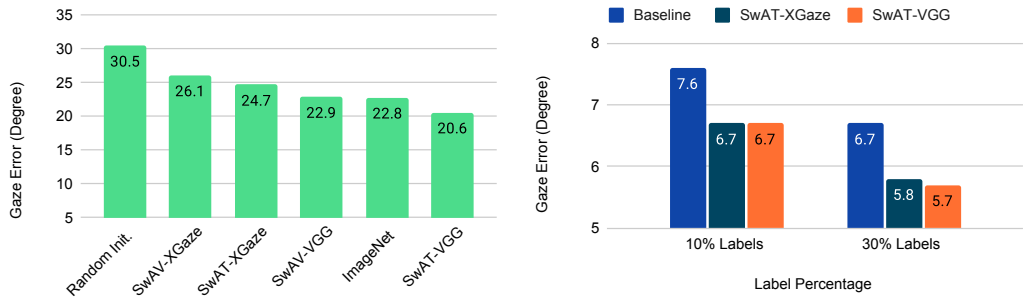

Figure 5.1. Left. Results of evaluating the unsupervised features of SwAV and SwAT pretrained with ETH-XGaze and VGG-Face datasets compared to random and ImageNet-based initializations. Performance is measured on the validation set of ETH-XGaze. **Right.** Results of semi-supervised learning using two subsets (10% and 30%) of the ETH-XGaze dataset, at the subject level. Performance is measured on the test set of ETH-XGaze.

Method	Pretrain	Arch.	ETH-XGaze	Gaze360	MPIIFace	MPIIFace*
Full-Face [64]	ImageNet	AlexNet+SW	N/A	N/A	4.8	N/A
Dilated-Net [6]	ImageNet	Dilated-CNN	N/A	N/A	4.8	N/A
RT-GENE [14]	ImageNet	VGG-16	N/A	N/A	4.8	N/A
Gaze360 [23]	ImageNet	ResNet-18	N/A	13.2	N/A	N/A
MTGLS [15]	MS-Celeb-1M	ResNet-50	N/A	12.8	N/A	N/A
ETH-XGaze [60]	ImageNet	ResNet-50	4.5	N/A	4.8	7.1†
Baseline (ours)	Random Init.	ResNet-50	5.9	12.2	5.7	8.5
SwAT (ours)	ETH-XGaze	ResNet-50	4.5	11.9	5.2	7.5
SwAT (ours)	VGG-Face	ResNet-50	**4.4**	**11.6**	5.0	**6.9**

Table 5.2. Comparison of SwAT with state-of-the-art appearance-based gaze estimation works, reported as average angular gaze error (degrees). Best results are bolded. Performances of the state-of-the-art approaches are shown as reported by their authors, except values marked with †. MPIIFaceGaze* denotes the unnormalized version of MPIIFaceGaze.

5.3 Semi-supervised Learning

In this evaluation, we examine the label-efficiency of SwAT. To achieve that, we perform semi-supervised learning on two subsets of the ETH-XGaze dataset. More precisely, we define two subsets i.e., 10% and 30% at subject level, and finetune the whole network on these subsets. As a baseline, we train a counterpart on the same subsets and with the same architecture but instead of using pretrained SwAT weights, we randomly initialize the weights. Fig. 5.1 (right) depicts the results of the semi-supervised learning. As can be seen, ResNet-50 pretrained with SwAT improves the baseline up to 1.0° when only 10% and 30% of labeled data at the subject level is available. This is of great importance in the gaze estimation context as recruiting fewer subjects saves cost and time.

5.4 Comparison to state of the art

We compare SwAT with state-of-the-art methods for full-face appearance-based gaze estimation on four datasets, namely, ETH-XGaze, Gaze360, MPIIFaceGaze, and MPIIFaceGaze*. We pretrain SwAT with ResNet-50 as encoder on ETH-

Method	TrainTest	ETH-XGaze	Gaze360	MPIIFace	MPIIFace*
Supervised	ETH-XGaze	-	30.0	23.5	17.5
	Gaze360	25.6	-	30.4	21.5
	MPIIFace	32.2	27.4	-	-
	MPIIFace*	35.5	28.9	-	-
SwAT	ETH-XGaze	-	**22.9**	**12.1**	**11.6**
	Gaze360	**19.4**	-	**13.0**	**12.8**
	MPIIFace	**29.5**	**24.9**	-	-
	MPIIFace*	**32.6**	**25.5**	-	-

Table 5.3. Comparison between supervised baseline and SwAT on cross-dataset evaluation. Numbers denote gaze error in degrees. Best results are bolded.

XGaze (without labels) and VGG-Face datasets. Then, we finetune the whole network using the aforementioned datasets. As a baseline, we also train the same encoder (ResNet-50) solely in a supervised fashion. Tab. 5.2 shows the comparison with the state of the art along with the datasets used for pretraining and the type of encoder. As can be seen, the supervised baseline is unable to outperform the state of the art, except on Gaze360. However, the same encoder boosted with SwAT unsupervised pretrained features achieves up to 25%, 5%, 14%, and 19% improvements compared to the supervised baseline on ETH-XGaze, Gaze360, MPIIFaceGaze, and MPIIFaceGaze*, respectively. Furthermore, SwAT pretrained with the VGG-Face dataset outperforms SwAT pretrained on ETH-XGaze (without labels) on all four benchmarks. This suggests that SwAT can effectively make use of uncurated datasets. On ETH-XGaze, SwAT pretrained with VGG-Face outperforms the state of the art that utilizes the pretrained ImageNet supervised weights. In addition, SwAT improves the state of the art up to 9% on Gaze360 while slightly underperforming it on MPIIFaceGaze. However, we can better observe the benefit of SwAT on the unnormalized version of MPIIFaceGaze (MPIIFaceGaze*), where SwAT improves the ETH-XGaze method with no data normalization by 0.2°. These results demonstrate the superior performance of SwAT in unrestricted scenarios.

5.5 Cross-dataset Evaluation

To evaluate the out-of-distribution generalization capability of SwAT, we perform a cross-dataset evaluation, i.e., training on a given dataset and testing on other datasets. We consider four datasets, namely, ETH-XGaze, Gaze360, MPIIFaceGaze, and MPIIFaceGaze*. We use ResNet-50 as encoder, pretrained on VGG-Face using SwAT. We compare our self-supervised approach (SwAT) to a supervised baseline that is solely trained in a supervised fashion. Tab. 5.3 shows the results of cross-dataset evaluation. SwAT improves the supervised baseline by a large amount. In detail, SwAT achieves up to 24% relative improvement on the ETH-XGaze dataset, and outperforms the supervised counterpart by 24% on Gaze360, by 57% on MPIIFaceGaze, and by 41% on MPIIFaceGaze*.

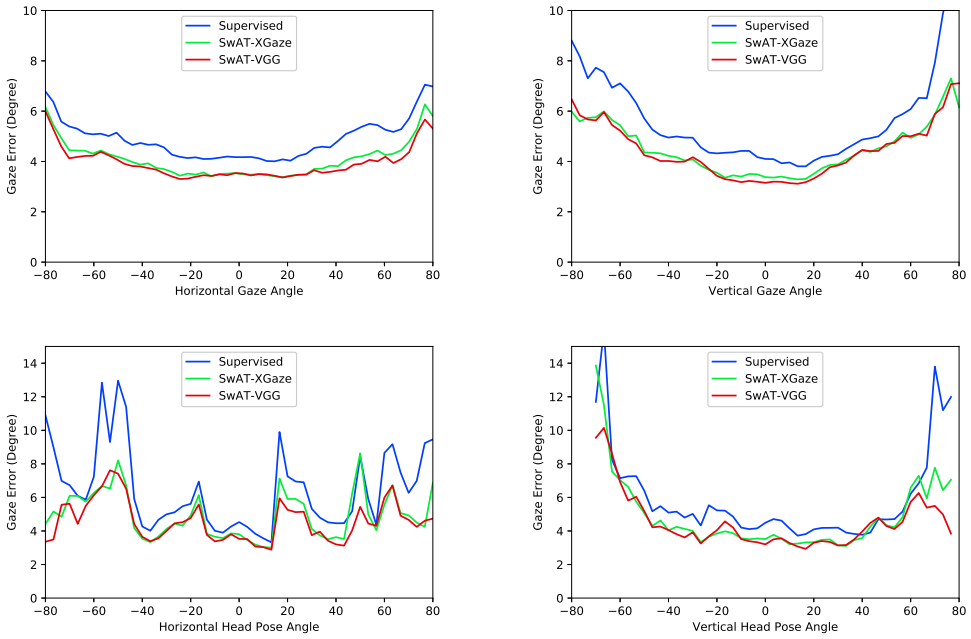

Figure 5.2. Gaze estimation error across horizontal (left) and vertical (right) for gaze and head pose directions in degrees.

5.6 Robustness Analysis

Mean gaze error is not quite an informative indicator of how a method performs within a specific gaze direction and head pose range. Thus, we conduct a robustness analysis to shed light on the performance of our method across gaze and head pose angles. Fig. 5.2 depicts the gaze error in degrees across horizontal and vertical gaze and head pose angles on the test set of ETH-XGaze. We observe that the performance of the supervised baseline substantially decreases as a function of the number of samples in ETH-XGaze (gaze angles follow a Gaussian-like distribution centered at 0, whereas the head pose distribution is multimodal [60]). In contrast, SwAT demonstrates superior robustness across all directions compared to the supervised baseline. However, SwAT pretrained with VGGFace is consistently more stable than SwAT pretrained on ETH-XGaze (without labels), especially in case of extreme gaze and head pose angles. We repeat the same analysis for the semi-supervised setting (Sec. 4.3). As shown in Fig. 5.3, overall, across both horizontal and vertical directions, the performance of SwAT is superior to the supervised (Random Init.) baselines. Nevertheless, the error curves slightly fluctuate for extreme head poses.

5.7 Equivariance Analysis

To evaluate the equivariance capability, we rely on the definition of equivariance (Eq. 3.4) and calculate the following metric ($\mathcal{L}_{equ}$):

$$\mathcal{L}_{equ} = \frac{1}{N}\sum_{i=1}^{N} ||f_\phi(t_I^g(\boldsymbol{x_i})) - t_F^g(f_\phi(\boldsymbol{x_i}))||_2. \tag{5.2}$$

We compare f_ϕ pretrained with SwAV and SwAT on the VGG-Face dataset. As

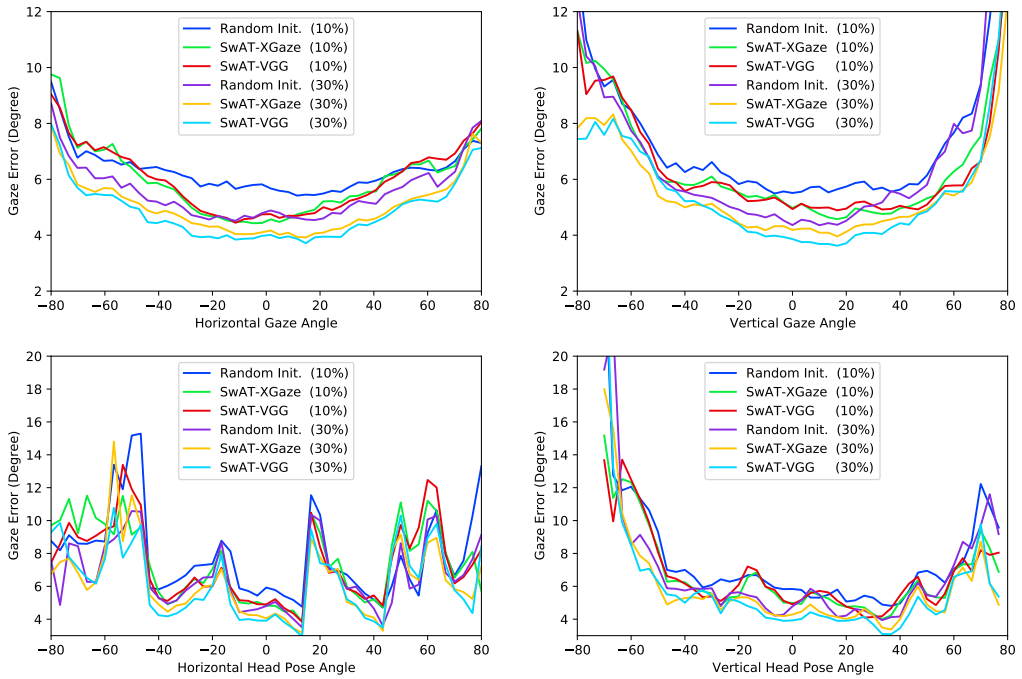

Figure 5.3. Robustness Analysis for Semi-Supervised Setting. Gaze estimation error across horizontal (left) and vertical (right) for gaze and head pose directions in degrees. The percentages show the amount of labeled data used for finetuning.

the evaluation datasets, we specifically focus on unconstrained gaze scenarios and calculate $\mathcal{L}_{equ}$ for Gaze360 and MPIIFaceGaze*. We expect SwAT to achieve lower values, which indicates enforcing equivariance. Fig. 5.4 depicts the results of $\mathcal{L}_{equ}$ on Gaze360 (left) and MPIIFaceGaze* (right), varying rotation degrees. As shown, in both cases SwAT consistently outperforms SwAV in the whole rotation range. More precisely, on average, SwAT achieves 27% and 21% relative improvements compared to SwAV on Gaze360 and MPIIFaceGaze*, respectively. Moreover, we calculate $\mathcal{L}_{equ}$ for horizontal flip and find that SwAT improves SwAV by 26% on Gaze360 and 21% on MPIIFaceGaze*.

5.8 Qualitative Results

Fig. 5.5 shows the estimated gaze direction on the test set of Gaze360. As shown, the supervised model demonstrates a large discrepancy compared to the ground-truth vectors while SwAT estimations better match the ground truth. It can be seen that SwAT is able to better estimate the gaze direction in extreme head-pose conditions. In the last column, we show some failure cases where SwAT and supervised model are not on par with the ground truth. In addition to Gaze360, we also show the estimated gaze direction on MPIIFaceGaze in Fig. 5.6. Note that in the case of MPIIFaceGaze, we performed leave-one-person-out evaluation for two subjects. The visual results in Fig. 5.6 suggest that SwAT achieves higher performance than the supervised baseline in the challenging case of extreme illumination condition. The last column in Fig. 5.6 shows some failure cases where both SwAT and supervised model fail to follow the ground-truth. Nevertheless, as can be seen from the failure

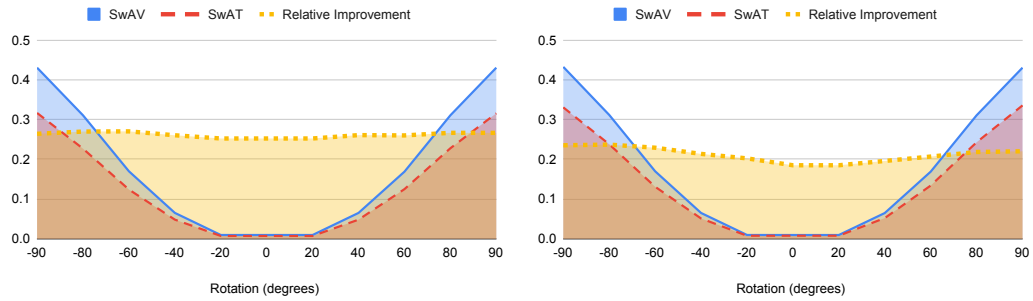

Figure 5.4. Results of calculating $\mathcal{L}_{equ}$ for SwAV and SwAT on Gaze360 (**Left**) and MPIIFaceGaze* (**Right**) datasets. The dotted lines shows the relative improvement achieved by SwAT over SwAV.

Figure 5.5. Visual results of estimated gaze direction on the test set of Gaze360. The green, red, and blue colors are, SwAT (→), Supervised (→), Ground-truth (→), respectively. The last column shows examples of failure cases.

case of closed eyes in both figures (Fig. 5.5 bottom row, Fig. 5.6 top row), despite the fact that the ground truth indicates the theoretical gaze direction, SwAT estimates a downward direction, which is more aligned with the closed eye direction.

5.9 Ablation Studies

In this section, we vary some of the key hyperparameters of SwAT such as the number of prototypes (Sec. 5.9.1), the number of epochs (Sec. 5.9.2), and the dimensionality of the projection head (Sec. 5.9.3). Throughout these experiments, we use ResNet-50 as the backbone encoder and ETH-XGaze (without labels) dataset for pretraining. Then, we freeze the backbone and train a linear regressor on top using the training set of ETH-XGaze, and measure the performance on the validation set. We set the input image size to 112×112. The default values for the number of prototypes, number of epochs, and dimensionality of projection-head are 500, 100, and 128, respectively, unless otherwise specified.

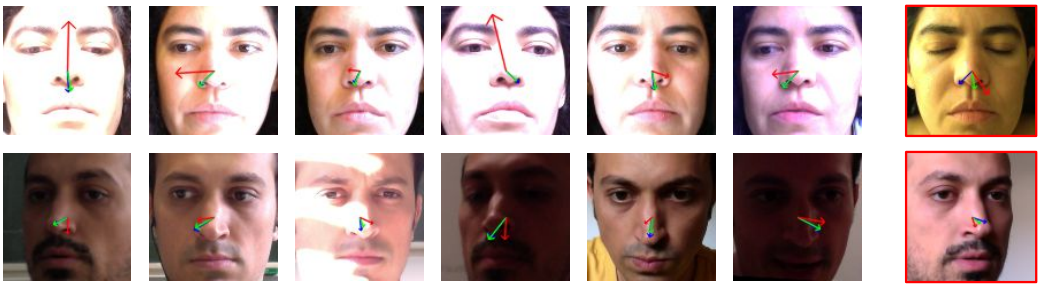

Figure 5.6. Visual results of estimated gaze direction on the test set of MPIIFaceGaze. The green, red, and blue colors are, SwAT (→), Supervised (→), Ground-truth (→), respectively. The last column shows examples of failure cases.

	Number of prototypes				d		epochs		
	500	1500	3000	6000	128-D	256-D	100	200	400
Gaze Error	**25.8**	26.0	25.9	26.0	26.0	**25.7**	**26.0**	26.3	26.4

Table 5.4. Results of ablation study on the number of prototypes, the dimensionality of the projection head (d), and the number of epochs. Numbers denote gaze error in degrees. Best results are bolded.

5.9.1 Number of prototypes

In this experiment, we investigate the effect of the number of prototypes (M) on the performance of SwAT. To achieve that, we consider four candidates, i.e., 500, 1500, 3000, and 6000. As shown in Tab. 5.4, we observe a slight difference in the performance of SwAT with different numbers of prototypes. This shows that the number of prototypes has a negligible impact on the performance of SwAT.

5.9.2 Number of epochs

We aim at increasing the number of epochs for pretraining from 100 to 200 and 400 epochs to assess whether SwAT takes advantage of longer pretraining. Results in Tab. 5.4 suggest that 100 epochs is sufficient and further pretraining leads to worse results.

5.9.3 Dimensionality of projection-head

In this experiment, we increase the dimensionality of the projection head d from 128-D to 256-D. As shown in Tab. 5.4, SwAT achieves a slightly better result with 256-D compared to 128-D.

5.10 Comparison with PeCLR [44]

In this subsection, we shed light on the differences between our equivariance formulation (SwAT) and PeCLR [44], a self-supervised approach for the task of 3D hand pose estimation. To avoid trivial solutions, PeCLR uses a contrastive loss that attracts the positive pairs while repelling the negative pairs. Furthermore, PeCLR achieves equivariance via inverting the image-space affine transformations in feature space which results in having the same affine information for both positive and

negative pairs. Thus, the contrastive loss has to push apart representations with the same affine information in feature space. Additionally, the negative pairs may also contain faces with similar gaze or head poses. In contrast, SwAT equalizes the feature vectors in terms of affine information and does not require negative samples. SwAT learns more geometry-aware representations as throughout training iterations SwAT sees the same image under various transformation information in feature space. Thus, the same image can have different cluster assignments depending on the randomly sampled transformation. Whereas, throughout training, PeCLR observes the same image with the same transformation information in feature space.

We compare both methods in the same setting and we use rotation as the only affine transformation. We evaluate the quality of the features by linear evaluation where we freeze the backbone and train a linear regressor on top. The results show that SwAT achieves better performance (29.2°) compared to PeCLR (29.6°).

Chapter 6

Conclusion

In this thesis, we explored the effectiveness of a self-supervised method in the context of gaze estimation, and proposed a novel approach (SwAT) to learn an equivariant representation for geometric transformations, i.e., rotations and horizontal flip. Our approach is task-agnostic and can be applied to any joint embedding-based self-supervised approach. We showed that SwAT learns more informative representations than other pretraining schemes for the task of gaze estimation. We also showed that our approach fueled by a large-scale uncurated dataset achieves more generalizable and consistent results, outperforming the supervised baselines and state-of-the-art approaches for both within- and cross-dataset settings. We also showed that our method achieves superior performance with fewer subjects. Thus, our approach can be leveraged to boost the performance of current gaze estimation systems in the real world via leveraging large-scale freely available face images on the Internet.

Bibliography

[1] Pulkit Agrawal, João Carreira, and Jitendra Malik. Learning to see by moving. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 37–45, 2015. 4

[2] Adrien Bardes, Jean Ponce, and Yann LeCun. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations*, 2022. 1, 2, 4

[3] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 1, 2, 4, 6, 12

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 1, 2, 4, 14

[5] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *arXiv preprint arXiv:2011.10566*, 2020. 1, 4

[6] Zhaokang Chen and Bertram E. Shi. Appearance-based gaze estimation using dilated-convolutions. In *ACCV*, 2018. 1, 4, 17

[7] Zhaokang Chen and Bertram E. Shi. Offset calibration for appearance-based gaze estimation via gaze decomposition. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 259–268, 2020. 5

[8] Yihua Cheng, Yiwei Bao, and Feng Lu. Puregaze: Purifying gaze feature for generalizable gaze estimation, 2021. 5

[9] Yihua Cheng, Shiyao Huang, Fei Wang, Chen Qian, and Feng Lu. A coarse-to-fine adaptive network for appearance-based gaze estimation. In *AAAI*, 2020. 4

[10] 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. 1

[11] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. 7, 12

[12] Murthy L R D and Pradipta Biswas. Appearance-based gaze estimation using attention and difference mechanism. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 3143–3152, June 2021. 4

[13] Rumen Dangovski, Li Jing, Charlotte Loh, Seung-Jun Han, Akash Srivastava, Brian Cheung, Pulkit Agrawal, and Marin Soljaić. Equivariant contrastive learning. In *ICLR*, 2022. 4

[14] Tobias Fischer, Hyung Jin Chang, and Y. Demiris. Rt-gene: Real-time eye gaze estimation in natural environments. In *ECCV*, 2018. 17

[15] Shreya Ghosh, Munawar Hayat, Abhinav Dhall, and Jarrod Knibbe. Mtgls: Multi-task gaze estimation with limited supervision. *ArXiv*, abs/2110.12100, 2021. 5, 17

[16] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*, 2018. 4

[17] Jean-Bastien Grill, Florian Strub, Florent Altch'e, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. *ArXiv*, abs/2006.07733, 2020. 1, 2, 4

[18] E. D. Guestrin and M. Eizenman. General theory of remote gaze estimation using the pupil center and corneal reflections. *IEEE Transactions on Biomedical Engineering*, 53:1124–1133, 2006. 1

[19] 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. *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 1149–1158, 2019. 5

[20] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9726–9735, 2020. 2, 4

[21] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 11

[22] Ashish Jaiswal, Ashwin Ramesh Babu, Mohammad Zaki Zadeh, Debapriya Banerjee, and Fillia Makedon. A survey on contrastive self-supervised learning. *ArXiv*, abs/2011.00362, 2020. 1

[23] Petr Kellnhofer, Adrià Recasens, Simon Stent, W. Matusik, and A. Torralba. Gaze360: Physically unconstrained gaze estimation in the wild. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6911–6920, 2019. 1, 4, 11, 13, 17

[24] Rakshit Kothari, Shalini De Mello, Umar Iqbal, Wonmin Byeon, Seonwook Park, and Jan Kautz. Weakly-supervised physically unconstrained gaze estimation. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9975–9984, 2021. 5

[25] K. Krafka, A. Khosla, Petr Kellnhofer, Harini Kannan, S. Bhandarkar, W. Matusik, and A. Torralba. Eye tracking for everyone. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2176–2184, 2016. 1, 4, 5

[26] Peng Li, Xuebin Hou, Xingguang Duan, Hiu Man Yip, Guoli Song, and Yunhui Liu. Appearance-based gaze estimator for natural interaction control of surgical robots. *IEEE Access*, 7:25095–25110, 2019. 1

[27] Dongze Lian, Lina Hu, Weixin Luo, Yanyu Xu, Lixin Duan, Jingyi Yu, and Shenghua Gao. Multiview multitask gaze estimation with deep convolutional neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 30:3010–3023, 2019. 1, 4

[28] E. Linden, J. Sjostrand, and A. Proutiere. Learning to personalize in appearance-based gaze tracking. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 1140–1148, oct 2019. 5

[29] Gang Liu, Yuechen Yu, Kenneth Alberto Funes Mora, and Jean-Marc Odobez. A differential approach for gaze estimation with calibration. In *BMVC*, 2018. 5

[30] Xiao Liu, Fanjin Zhang, Zhenyu Hou, Li Mian, Zhaoyu Wang, Jing Zhang, and Jie Tang. Self-supervised learning: Generative or contrastive. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–1, 2021. 1

[31] Yunfei Liu, Ruicong Liu, Haofei Wang, and Feng Lu. Generalizing gaze estimation with outlier-guided collaborative adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 5

[32] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv: Learning*, 2017. 11

[33] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6706–6716, 2020. 4

[34] Kenneth Alberto Funes Mora and J. Odobez. Geometric generative gaze estimation (g3e) for remote rgb-d cameras. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1773–1780, 2014. 1

[35] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016. 4

[36] Cristina Palmero, Javier Selva, Mohammad Ali Bagheri, and Sergio Escalera. Recurrent CNN for 3d gaze estimation using appearance and shape cues. In *British Machine Vision Conference 2018, BMVC 2018, Newcastle, UK, September 3-6, 2018*, page 251. BMVA Press, 2018. 1

[37] Cristina Palmero, Javier Selva, Mohammad Ali Bagheri, and Sergio Escalera. Recurrent cnn for 3d gaze estimation using appearance and shape cues. In *BMVC*, 2018. 5

[38] Seonwook Park, Shalini De Mello, Pavlo Molchanov, Umar Iqbal, Otmar Hilliges, and Jan Kautz. Few-shot adaptive gaze estimation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9367–9376, 2019. 5

[39] Seonwook Park, Adrian Spurr, and Otmar Hilliges. Deep pictorial gaze estimation. In *ECCV*, 2018. 1, 4

[40] Omkar M. Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. In *British Machine Vision Conference*, 2015. 11

[41] Deepak Pathak, Philipp Krähenbühl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2536–2544, 2016. 4

[42] Akshay Rangesh, Bowen Zhang, and Mohan M. Trivedi. Driver gaze estimation in the real world: Overcoming the eyeglass challenge. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 1054–1059, 2020. 1

[43] Roberto Rodríguez-Labrada, Yaimeé Vázquez-Mojena, and Luis Velázquez-Pérez. Eye movement abnormalities in neurodegenerative diseases. *Eye Motility*, 2019. 1

[44] Adrian Spurr, Aneesh Dahiya, Xi Wang, Xucong Zhang, and Otmar Hilliges. Self-supervised 3d hand pose estimation from monocular rgb via contrastive learning. In *International Conference on Computer Vision (ICCV)*, 2021. iv, 1, 4, 15, 16, 22, 23

[45] Yusuke Sugano, Y. Matsushita, and Yoichi Sato. Appearance-based gaze estimation using visual saliency. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35:329–341, 2013. 1

[46] Yunjia Sun, Jiabei Zeng, Shiguang Shan, and Xilin Chen. Cross-encoder for unsupervised gaze representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3702–3711, October 2021. 5

[47] K. Tan, D. Kriegman, and N. Ahuja. Appearance-based eye gaze estimation. *Sixth IEEE Workshop on Applications of Computer Vision, 2002. (WACV 2002). Proceedings.*, pages 191–195, 2002. 1

[48] Ling Tao, Quan Wang, Ding Liu, J. Wang, Zi qing Zhu, and L. Feng. Eye tracking metrics to screen and assess cognitive impairment in patients with neurological disorders. *Neurological Sciences*, 41:1697–1704, 2020. 1

[49] Kang Wang, S. Wang, and Q. Ji. Deep eye fixation map learning for calibration-free eye gaze tracking. *Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications*, 2016. 1

[50] Kang Wang, Rui Zhao, and Q. Ji. Human computer interaction with head pose, eye gaze and body gestures. *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 789–789, 2018. 1

[51] Kang Wang, Rui Zhao, Hui Su, and Qiang Ji. Generalizing eye tracking with bayesian adversarial learning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11899–11908, 2019. 5

[52] Daniel E. Worrall, Stephan J. Garbin, Daniyar Turmukhambetov, and Gabriel J. Brostow. Interpretable transformations with encoder-decoder networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5737–5746, 2017. 8

[53] Yong Wu, Gongyang Li, Zhi Liu, Mengke Huang, and Yang Wang. Gaze estimation via modulation-based adaptive network with auxiliary self-learning. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1–1, 2022. 5

[54] Enze Xie, Jian Ding, Wenhai Wang, Xiaohang Zhan, Hang Xu, Peize Sun, Zhenguo Li, and Ping Luo. Detco: Unsupervised contrastive learning for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8392–8401, 2021. 1

[55] Yuyang Xie, Jianhong Wen, Kin Wai Lau, Yasar Abbas Ur Rehman, and Jiajun Shen. What should be equivariant in self-supervised learning. In *CVPR Workshops*, 2022. 4

[56] Yang You, Igor Gitman, and Boris Ginsburg. Large batch training of convolutional networks. *arXiv: Computer Vision and Pattern Recognition*, 2017. 11

[57] Yuechen Yu, Gang Liu, and Jean-Marc Odobez. Improving few-shot user-specific gaze adaptation via gaze redirection synthesis. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11929–11938, 2019. 5

[58] Yuechen Yu and Jean-Marc Odobez. Unsupervised representation learning for gaze estimation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7312–7322, 2020. 5

[59] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016. 4

[60] 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 *European Conference on Computer Vision (ECCV)*, 2020. 1, 4, 11, 12, 17, 19

[61] Xucong Zhang, Yusuke Sugano, and Andreas Bulling. Revisiting data normalization for appearance-based gaze estimation. *Proceedings of the 2018 ACM Symposium on Eye Tracking Research & Applications*, 2018. 5, 12

[62] Xucong Zhang, Yusuke Sugano, and Andreas Bulling. Evaluation of appearance-based methods and implications for gaze-based applications. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019. 1

[63] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. Appearance-based gaze estimation in the wild. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4511–4520, 2015. 1, 11

[64] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. It's written all over your face: Full-face appearance-based gaze estimation. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pages 2299–2308. IEEE, 2017. 5, 11, 17

[65] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. Mpiigaze: Real-world dataset and deep appearance-based gaze estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41:162–175, 2019. 5

[66] Zhiwei Zhu and Q. Ji. Novel eye gaze tracking techniques under natural head movement. *IEEE Transactions on Biomedical Engineering*, 54:2246–2260, 2007. 1