Driver Glance Classification In-the-wild: Towards Generalization Across Domains and Subjects

Sandipan Banerjee[1], Ajjen Joshi[1], Jay Turcot[1], Bryan Reimer[2], and Taniya Mishra*[3]
[1] Smart Eye, [2] Massachusetts Institute of Technology, [3] SureStart
{firstname.lastname}@smarteye.ai, reimer@mit.edu, taniya.mishra@mysurestart.com

***Abstract*— Distracted drivers are dangerous drivers. Equipping advanced driver assistance systems (ADAS) with the ability to detect driver distraction can help prevent accidents and improve driver safety. In order to detect driver distraction, an ADAS must be able to monitor their visual attention. We propose a model that takes as input a patch of the driver's face along with a crop of the eye-region and classifies their glance into 6 coarse regions-of-interest (ROIs) in the vehicle. We demonstrate that an hourglass network, trained with an additional reconstruction loss, allows the model to learn stronger contextual feature representations than a traditional encoder-only classification module. To make the system robust to subject-specific variations in appearance and behavior, we design a personalized hourglass model tuned with an auxiliary input representing the driver's baseline glance behavior. Finally, we present a weakly supervised multi-domain training regimen that enables the hourglass to jointly learn representations from different domains (varying in camera type, angle), utilizing unlabeled samples and thereby reducing annotation cost.**

I. Introduction

Driver distraction has been shown to be a leading cause of vehicular accidents [16]. Anything that competes for a driver's attention, such as talking or texting on the phone, using the car's navigation system or eating, can be a cause of distraction. A distracted individual often directs their visual attention away from driving, which has been shown to increase accident risk [39]. According to the NHTSA, a large percentage of crashes and near-crashes occur when the driver looks away from the street [35].Therefore, driver glance behavior can be an important signal in determining their level of distraction. A system that can accurately detect where the driver is looking can then be used to alert drivers when their attention shifts away from the road. Such systems can also monitor driver attention to manage and motivate improved awareness [12]. For example, the system can decide whether a driver's attention needs to be cued back to the road prior to safely handing them back the control.

A real-time system that can classify driver attention into a set of ROIs can be used to infer their overall attentiveness and offer predictive indication of attention failures associated with crashes and near-crashes [61]. Real-time tracking of driver gaze from video is attractive because of the low equipment cost but challenging due to variations in illumination, eye occlusions caused by eyeglasses/sunglasses and poor video quality due to vehicular movements and sensor noise. In this paper, we propose a model that can predict driver glance ROI, given a patch of the driver's face along with a crop of their eye-region. We show that an hourglass network [56], [50], composed of encoder-decoder modules, trained with a reconstruction loss on top of the classification task, performs better than a vanilla CNN. The reconstruction task serves as a regularizer [49], helping the model learn robust representations of the input by implicitly leveraging useful information around its context [55], [52].

However, a model that makes predictions based on only a single static frame may struggle to deal with variations in subject characteristics not well represented in the training set (*e.g.* a shorter or taller-than average driver may have different appearances for the default on-the-road driving behavior). To address this challenge, we add an auxiliary input stream representing the subject's baseline glance behavior, yielding improved performance over a rigid network.

Another challenge associated with an end-to-end glance classification system is the variation in camera type (RGB/NIR) and placement (on the steering wheel or rearview mirror). Due to variations in cabin configuration, it is impossible to place the camera in the same location with a consistent view of the car interior and the driver. Therefore, a model trained on driver head-poses associated with a specific camera-view may not generalize. To overcome this domain-mismatch challenge, we present a framework to jointly train models in the presence of data from multiple domains (camera types and views). Leveraging our backbone hourglass' reconstruction objective, this framework can utilize unlabeled samples from multiple domains along with weak supervision to jointly learn stronger domain-invariant representations while effectively reducing labeling cost.

In summary, we make the following contributions: (1) we propose an hourglass architecture that can predict driver glance ROI from static images, illustrating the utility of adding a reconstruction loss to learn robust representations even for classification tasks; (2) we design a personalized version of our hourglass model, that additionally learns residuals in feature space from the driver's default 'eyes-on-the-road' behavior, to better tune output mappings wrt the subject's default; (3) we formulate a weakly supervised multi-domain training approach that utilizes unlabeled samples for classification and allows for model adaptation to novel camera types and angles, while reducing the associated labeling cost.

II. Related Work

Computer-vision based driver monitoring systems [9] have been used to estimate a driver's state of fatigue [32], cogni-

*Work done while at Smart Eye

978-1-6654-3176-7/21/$31.00 ©2021 IEEE

Fig. 1: Sample frames from the datasets used in our experiments (MIT2013 (left), AVT (middle) and In-house (right). For each dataset, we present an example raw frame captured by the camera, and an example each of a driver's cropped face for each driver glance region-of-interest class.

tive load [19] or the driver's focus on the road [72].

Gaze estimation: The problem of tracking gaze from video has been studied extensively [23], [6]. Professional gaze tracking systems do exist (*e.g.* Tobii[1]), however they typically require user or session-specific calibration to achieve good performance. Appearance-based, calibration-free gaze estimation has numerous applications in computer vision, from gaze-based human-computer interaction to analysis of visual behavior. Researchers have utilized both real [81] and synthetic data [77], [76] to model gaze behavior, with generative approaches used to bridge the gap between synthetic and real distributions, so that models trained on one domain work well on another [64], [33].

Glance Classification: In the case of driver distraction, classifying where the driver is looking from an estimated gaze vector involves finding the intersection between the gaze vector and the 3D car geometry. A simpler alternative is to directly classify the driver image into a set of car ROIs using head pose [30], as well as eye region appearance[18]. Rangesh et al. focused on estimating driver gaze in the presence of eye-occluding glasses to synthetically remove eyeglasses from input images before feeding them to a classification network [54]. Ghosh et al. recently introduced the Driver Gaze in the Wild (DGW) dataset to further encourage research in this area [22].

Personalization: Personalized training has been applied to other domains (e.g. facial action unit [11] and gesture recognition [78], [31]) but not yet on vehicular glance classification. In the context of eye tracking, personalization is usually achieved through apriori user calibration. [37] reported results for unconstrained (calibration-free) eye tracking from mobile devices and showed calibration to significantly improve performance. For personalizing gaze models latent representation for each eye has been used [40], for utilizing saliency information in visual content [7] or adapting a generic example with few training samples [80].

Domain Invariance: Domain adaptation has been used in a variety of applications, *e.g.* object recognition [59]. Researchers have trained shared networks with samples from different domains, regularized via an adaptation loss between their embeddings [21], [70], or trained models with domain confusion to learn domain-agnostic embeddings [71], implemented by reducing distance between the domain embeddings [69], [51] or reversing the gradient specific to domain classification during backpropagation [20]. Another popular approach towards domain adaptation is to selectively fine-tune task specific models from pre-trained weights [27], [36] by freezing pre-trained weights that are tuned to specific tasks or domains [43], [44] or selectively pruning weights [5] to prudently adapt to new domains [48]. Specific to head pose, lighting and expression agnostic face recognition, approaches like feature normalization using class centers [75], [82] and class separation using angular margins [41], [74], [14] have been proposed. Such recognition tasks have also benefitted from mixing samples from different domains, like real and synthetic [45], [4].

While most research on gaze estimation proposes models that predict gaze vectors, our glance classification model directly predicts the actual ROI of the driver's gaze inside the vehicle. Unlike previous work, our multi-domain training approach tunes the model's ROI prediction to jointly work on multiple domains (*e.g.* car interiors), varying in camera type, angle and lighting, while requiring very little labeled data. Our model can be personalized for continual tuning based on the driver's behavior and anatomy as well.

III. Dataset Description and Data Analysis

MIT-2013: The dataset was extracted from a corpus of driver-facing videos, which were collected as part of large driving study that took place on a local interstate highway [46]. For each participant in the study, videos of the drivers were collected either in a 2013 Chevrolet Equinox or a Volvo XC60. The participants performed a number of tasks, such as using the voice interface to enter addresses or combining it with manual controls to select phone numbers, while driving. Frames with the frontal face of the drivers were then annotated to the following ROIs: 'road', 'center stack', 'instrument cluster', 'rearview mirror', 'left', 'right', 'left blindspot', 'right blindspot', 'passenger', 'uncodable', and 'other'. The data of interest was independently coded by two evaluators and mediated according to standards described by [66]. Following Fridman et al. [18], frames labeled 'left' and 'left blindspot' were given a single generic 'left' label and frames labeled 'right', 'right blindspot' and 'passenger' were given a generic 'right' label, while frames labeled 'uncodable', and 'other' were ignored. We used a subset of the data with 97 unique subjects, which was split into 60

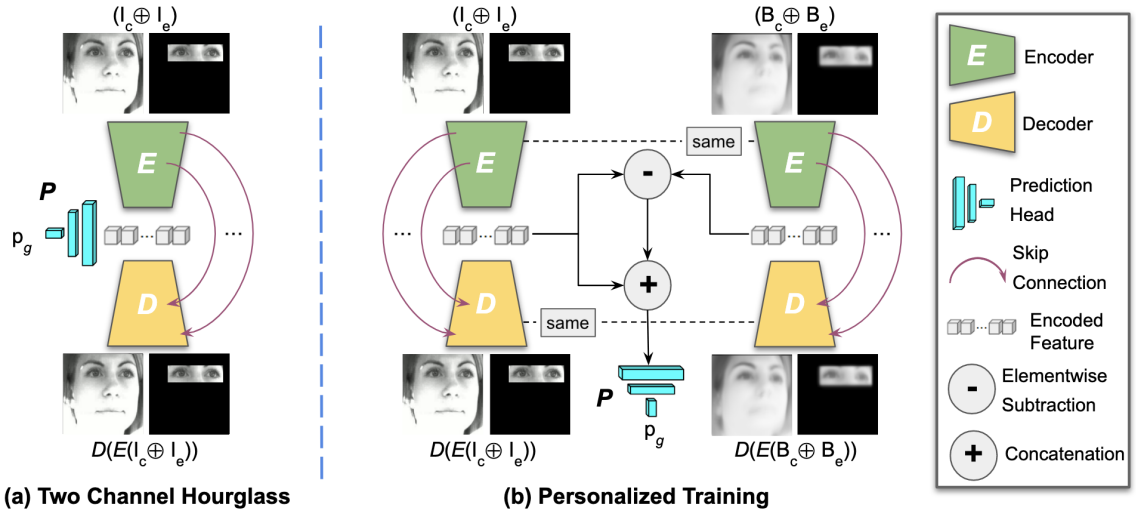

Fig. 2: Illustration of our (a) two channel hourglass and (b) multi-stream personalization models described in Sections IV-A and IV-B respectively.

train, 17 validation and 20 test subjects.

AVT: This dataset contains driver-initiated, non-critical disengagement events of Tesla Autopilot in naturalistic driving [47] and was extracted from a large corpus of naturalistic driving data, collected from an instrumented fleet of 29 vehicles, each of which record the IMU, GPS, CAN messages, and video streams of the driver face, the vehicle cabin and the forward roadway [17]. The MIT Advanced Vehicle Technology (MIT-AVT) study was designed to collect large-scale naturalistic driving data for better understanding of how drivers interact with modern cars to aid better design and interfaces as vehicles transition into increasingly automated systems. Each video was processed by a single coder with inter-rater reliability assessments as detailed in [47].

In-house: This dataset was collected to train machine learning models to estimate gaze from the RGB and NIR camera types and a challenging camera angle. A camera, with a wide-angle lens, was placed under the rear-view mirror for this collection, the focus of which was to capture data from a position where the entire cabin was visible. Participants followed instructions from a protocol inside a static/parked car, where they glanced at various ROIs using 3 behavior types: 'owl', 'lizard' and 'natural'[18]. In our experiments, we used samples from 85 participants - 50 for training, 18 for validation and 17 for testing. Videos of each participant was manually annotated by 3 human labelers. Example frames from all three datasets are shown in Figure 1.

We do not use the recently released DGW dataset [22] in our experiments as the annotations are provided for a different set of regions, with the driver seated on the right-hand side, which makes it difficult to integrate into our multi-domain training pipeline.

IV. Proposed Models

A. Two-channel Hourglass

While a standalone classification (*i.e.* encoder with prediction head) or reconstruction module (*i.e.* encoder-decoder) can produce high performance numbers for recognition or semantic segmentation or super-resolution tasks, combining them together has been shown to further boost model performance [15], [42], [49], [24], [62]. The auxiliary module's (prediction or reconstruction) loss acts as a regularizer [49] and boosts model performance on the primary task. For our specific task of driver glance classification, adding a reconstruction element can tune the model weights to implicitly pay close attention to contextual pixels while making a decision. Thus, instead of using a feed forward neural network, as traditionally done for classification tasks [38], [65], [25], [58], we use an hourglass structure consisting of encoder (E) and decoder (D) modules [56].

In our model, E takes as input the cropped face and eye patch images I_c and I_e respectively, concatenated together as a two-channel tensor $(\mathrm{I}_c \oplus \mathrm{I}_e)$ and produces a feature vector (*i.e.* $E(\mathrm{I}_c \oplus \mathrm{I}_e)$) as its encoded representation. This feature vector is then passed through a prediction head P to extract the estimated glance vector $\mathbf{p}_g$, before being sent to D to generate the face and eye patch reconstructions $D(E(\mathrm{I}_c \oplus \mathrm{I}_e))$, as shown in Figure 2.a. E is composed of a dilated convolution layer [79] followed by a set of n downsampling residual blocks [25] and a dense layer for encoding. D takes $E(\mathrm{I}_c \oplus \mathrm{I}_e)$ and passes it through n upsampling pixel shuffling blocks [63] followed by a convolution layer with *tanh* activation for image reconstruction [53], [60]. For better signal propagation, we add skip connections [56] between corresponding layers in E and D [3]. The encoded feature is also passed through the prediction head P, composed of two densely connected layers followed by softmax activation to produce the glance prediction vector $\mathbf{p}_g$.

The hourglass model is trained using a categorical cross entropy based classification loss L_{cls} between the ground truth glance vector $\mathbf{c}_g$ and the predicted glance vector $P(E(\mathrm{I}_c \oplus \mathrm{I}_e))$ (*i.e.* $\mathbf{p}_g$), and a pixelwise reconstruction loss L_{rec} between the input tensor $(\mathrm{I}_c \oplus \mathrm{I}_e)$ and its reconstruction $D(E(\mathrm{I}_c \oplus \mathrm{I}_e))$. For a given training batch $\mathcal{N}$ and the ground truth classes $\mathcal{C}$, they can be represented as:

$$L_{cls} = -\frac{1}{|\mathcal{N}|}\sum_{1}^{\mathcal{N}}\sum_{i}^{\mathcal{C}} \mathrm{c}_{gi} \log P(E(\mathrm{I}_c \oplus \mathrm{I}_e))_i \qquad (1)$$

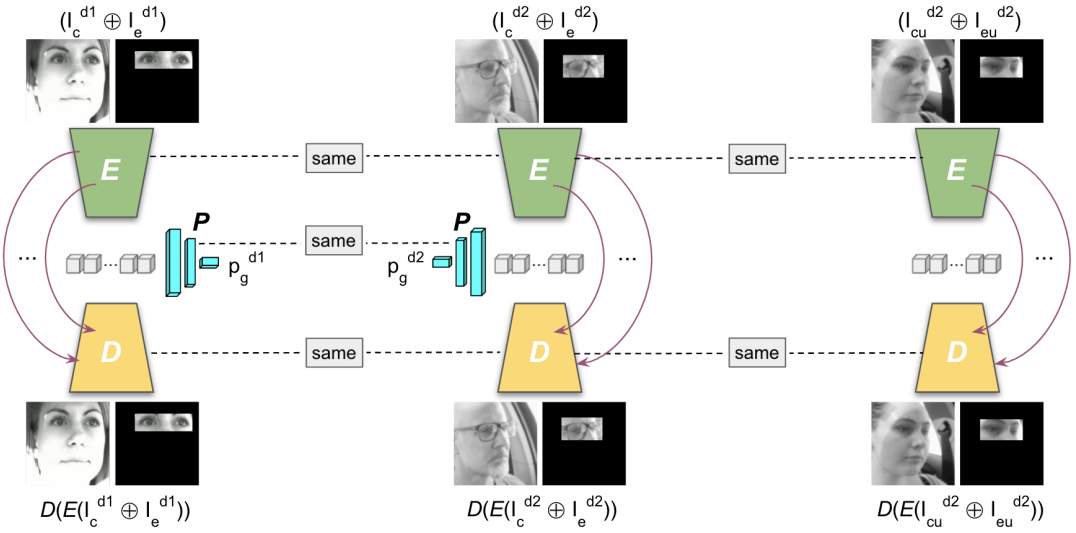

Fig. 3: Our multi-domain training pipeline: For every iteration, the model is trained with mini-batches consisting of labeled input samples from $d1$ ($(\mathrm{I}_c^{d1} \oplus \mathrm{I}_e^{d1})$) and $d2$ ($(\mathrm{I}_c^{d2} \oplus \mathrm{I}_e^{d2})$), and unlabeled input from $d2$ ($(\mathrm{I}_{cu}^{d2} \oplus \mathrm{I}_{eu}^{d2})$). The model weights are updated based on the overall loss accumulated over the mini-batches. It is to be noted that all three subjects in this figure are looking at the road, but appear very different due to different camera angles.

$$L_{rec} = \frac{1}{|\mathcal{N}|} \sum_{1}^{\mathcal{N}} |(\mathrm{I}_c \oplus \mathrm{I}_e) - D(E(\mathrm{I}_c \oplus \mathrm{I}_e))| \tag{2}$$

where $\mathcal{N}$ is the training set in a batch and $\mathcal{C}$ the ground truth classes. The overall objective L is defined as:

$$L = L_{cls} + \lambda_1 L_{rec} \tag{3}$$

B. Personalized Training

As mentioned earlier, introducing an auxiliary channel of baseline information can better tune the classification model to specific driver anatomy and behaviors. To this end, we propose a personalized version of our hourglass framework, composed of the same encoder (E) and decoder (D) modules. For each driver in the training dataset, we extract their baseline face crop B_e and eye patch B_e, where $\mathrm{B}_e = \frac{1}{m}\sum_{i=1}^{m} \mathrm{I}_e$ and $\mathrm{B}_c = \frac{1}{m}\sum_{i=1}^{m} \mathrm{I}_c$, for all cases where the driver is looking forward at the road. The baseline face crop and eye patch images are calculated prior to training.

During training, we extract the representation of the current frame $E(\mathrm{I}_c \oplus \mathrm{I}_e)$ by passing the face crop I_c and eye patch I_e images through E. Additionally, the baseline representation of the driver $E(\mathrm{B}_c \oplus \mathrm{B}_e)$ is computed by utilizing the baseline images. The residual between these tensors is computed in the representation space using encoded features as $E(\mathrm{I}_c \oplus \mathrm{I}_e) - E(\mathrm{B}_c \oplus \mathrm{B}_e)$. This residual acts as a measure of variance of the driver's glance behavior from looking forward, and is concatenated with the current frame representation $E(\mathrm{I}_c \oplus \mathrm{I}_e)$. This concatenated tensor is then passed through the prediction head P to get the glance prediction $\mathbf{p}_g$. Two streams each for E and D are deployed during training that share weights, as depicted in Figure 2.b.

The classification loss L_{cls}^p is then calculated as:

$$L_{cls}^p = -\frac{1}{|\mathcal{N}|} \sum_{1}^{\mathcal{N}} \sum_{i}^{\mathcal{C}} \mathrm{c}_{gi} \log P((E(\mathrm{I}_c \oplus \mathrm{I}_e) - E(\mathrm{B}_c \oplus \mathrm{B}_e)) \oplus E(\mathrm{I}_c \oplus \mathrm{I}_e))_i \tag{4}$$

where $\mathcal{N}$ is training batch and $\mathcal{C}$ the ground truth classes.

The reconstruction loss L_{rec}^p is calculated for both the current frame and baseline tensors as:

$$L_{rec}^p = \frac{1}{|\mathcal{N}|} \sum_{1}^{\mathcal{N}} |(\mathrm{I}_c \oplus \mathrm{I}_e) - D(E(\mathrm{I}_c \oplus \mathrm{I}_e))| + \frac{1}{|\mathcal{N}|} \sum_{1}^{\mathcal{N}} |(\mathrm{B}_c \oplus \mathrm{B}_e) - D(E(\mathrm{B}_c \oplus \mathrm{B}_e))| \tag{5}$$

The overall objective L^p is a weighted sum of these two losses, calculated as:

$$L^p = L_{cls}^p + \lambda_2 L_{rec}^p \tag{6}$$

C. Domain Invariance

As can be seen in Figure 1, driver glance can look significantly different when the camera type (RGB or NIR), its placement (steering wheel or rear-view mirror) and car interior changes. Such a domain mismatch can result in considerable decrease in performance when the classification model is trained on one dataset and tested on another, as experimentally shown in Section V. To mitigate this domain inconsistency problem, we propose a multi-domain training regime for our two-channel hourglass model. This regime leverages a rich set of labeled training images from one domain to learn domain invariant features for glance estimation from training samples from a second domain, only some of which are labeled. The hourglass structure of our model provides an advantage as the unlabeled samples from the second domain can also be utilized during training using D's reconstruction error.

Our multi-domain training starts with three input tensors:
(1) $(\mathrm{I}_c^{d1} \oplus \mathrm{I}_e^{d1})$ - the labeled face crop and eye patch images from the richly labeled domain $d1$,
(2) $(\mathrm{I}_c^{d2} \oplus \mathrm{I}_e^{d2})$ - the labeled face crop and eye patch images from the sparsely labeled second domain $d2$,
(3) $(\mathrm{I}_{cu}^{d2} \oplus \mathrm{I}_{eu}^{d2})$ - the unlabeled face crop and eye patch images from the second domain $d2$.

Each tensor is passed through E to generate their embedding, which are then passed through D to reconstruct the

input. For the input tensors with glance labels (*i.e.* $(\mathrm{I}_c^{d1} \oplus \mathrm{I}_e^{d1})$ and $(\mathrm{I}_c^{d2} \oplus \mathrm{I}_e^{d2})$), the encoded feature is also passed through P to get the glance predictions $\mathbf{p}_g^{d1}$ and $\mathbf{p}_g^{d2}$ respectively. We set shareable weights across the multi-streams of E, D and P during training, as shown in Figure 3.

The classification loss L_{cls}^{md} for the multi-domain training is set as:

$$L_{cls}^{md} = -\frac{1}{|\mathcal{N}^{d1}|}\sum_{1}^{\mathcal{N}^{d1}}\sum_{i}^{\mathcal{C}} \mathrm{c}_{gi}^{d1} \log P(E(\mathrm{I}_c^{d1} \oplus \mathrm{I}_e^{d1}))_i - \frac{1}{|\mathcal{N}^{d2}|}\sum_{1}^{\mathcal{N}^{d2}}\sum_{i}^{\mathcal{C}} \mathrm{c}_{gi}^{d2} \log P(E(\mathrm{I}_c^{d2} \oplus \mathrm{I}_e^{d2}))_i \quad (7)$$

where $\mathcal{N}^{d1}$ and $\mathcal{N}^{d2}$ are the labeled training batches, and c_g^{d1} and c_g^{d2} are the ground truth glance labels from domains $d1$ and $d2$ respectively.

Similarly, the reconstruction error L_{rec}^{md} is calculated as:

$$L_{rec}^{md} = \frac{1}{|\mathcal{N}^{d1}|}\sum_{1}^{\mathcal{N}^{d1}} \left|(\mathrm{I}_c^{d1} \oplus \mathrm{I}_e^{d1}) - D(E(\mathrm{I}_c^{d1} \oplus \mathrm{I}_e^{d1}))\right| + \frac{1}{|\mathcal{N}^{d2}|}\sum_{1}^{\mathcal{N}^{d2}} \left|(\mathrm{I}_c^{d2} \oplus \mathrm{I}_e^{d2}) - D(E(\mathrm{I}_c^{d2} \oplus \mathrm{I}_e^{d2}))\right| + \frac{1}{|\mathcal{N}_u^{d2}|}\sum_{1}^{\mathcal{N}_u^{d2}} \left|(\mathrm{I}_{cu}^{d2} \oplus \mathrm{I}_{eu}^{d2}) - D(E(\mathrm{I}_{cu}^{d2} \oplus \mathrm{I}_{eu}^{d2}))\right| \quad (8)$$

where $\mathcal{N}_u^{d2}$ is the unlabeled training batch from domain $d2$.

The full multi-domain loss L^{md} is calculated as:

$$L^{md} = L_{cls}^{md} + \lambda_3 L_{rec}^{md} \quad (9)$$

The weighing scalars λ_1 (3), λ_3 (6) and λ_3 (9) are hyper-parameters that are tuned experimentally.

V. Experiments

A. Training Details

To train our models we use ∼235K video frames from the MIT2013 dataset, and ∼153K and ∼163K for validation and testing respectively. The videos were split offline to assign into training, validation and testing buckets. Due to the large amount of labeled samples, we also use this dataset to represent the richly labeled domain (*i.e.* $d1$) for our domain invariant experiments, while using the AVT or In-house datasets as the second domain $d2$ (check Section IV-C). We randomly sample ∼204K frames (Training: 162K, Validation: 22K, Testing: 20K) from the AVT and ∼377K video frames (Training: 240K, Validation: 65K, Testing: 72K) from the In-house datasets for these experiments. All frames were downsampled to 96×96×1 to generate the facial image and the eye patch was cropped out (also 96×96×1 in size) using the eye-landmarks extracted using [29]. Frames with undetected faces were removed.

During training, we use the Adam optimizer [34] with the base learning rate set as 10^{-4} with a Dropout [67] layer (rate=0.7) between the dense layers of the prediction head in the two-channel hourglass network (Section IV-A). The weighing scalars λ_1, λ_2 and λ_3 are empirically set as 1, 1 and 10 respectively. We train all models using Tensorflow [2] coupled with Keras [10] on a single NVIDIA Tesla V100 card with the batch size set as 8. For the personalized model however, we find it optimal to train with a batch size of 16 and learning rate of 10^{-3}. To reduce computation cost and further prevent overfitting, we stop model training once the validation loss plateaus across three epochs and save the model snapshot for testing. To compute statistical significance between the performance of various methods, we train each model 5 times using random seeds for initialization. We only use the trained encoder and prediction head during inference.

For training the personalization framework, we prepare multiple mini-batches for every iteration with the current frame (I_c, I_e) and baseline frame (B_c, B_e) inputs. For the domain invariant regimen, the mini-batches are prepared with labeled $d1$ $((\mathrm{I}_c^{d1} \oplus \mathrm{I}_e^{d1}))$, labeled $d2$ $((\mathrm{I}_c^{d2} \oplus \mathrm{I}_e^{d2}))$ and unlabeled $d2$ inputs $((\mathrm{I}_{cu}^{d2} \oplus \mathrm{I}_{eu}^{d2}))$. The overall loss is computed from the mini-batches before updating model weights.

Computation Overhead: In terms of model size, the encoder E and prediction head P together consist of 24M parameters while adding the decoder D for reconstruction increases the number to 54M. While D does add computational load during training, only E and P together are required for inference. Thus, turning the typical classifier into an hourglass does not introduce additional overhead when deployed in production. The personalized version of the model has the same number of trainable parameters but does require an additional stream of baseline driver information.

B. Performance on the MIT2013 Dataset

Post training, we test our two-channel hourglass and personalization models for glance estimation on the test frames from the MIT2013 dataset[46]. To gauge of their effectiveness, we compare our model with the following:

(1) **Landmarks + MLP**. Following [18], we train a baseline MLP model with 3 dense layers on a flattened representation of facial landmarks extracted using [29].

(2) **Dense Eye-landmarks + Headpose**. This recently-proposed lightweight MLP model is trained on a set of dense landmarks of the eyes, as well as head pose estimates [13].

(3) **Baseline CNN**. We also train a baseline CNN with 4 convolutional and max pooling layers followed by 3 dense layers, similar to AlexNet [38]. The baseline CNN takes as input the 96×96×1 cropped face image.

(4) **Upperface Squeezenet**. Following the best performing configuration in [73], we train a SqueezeNet model [28] on the upper half of the driver's face.

(5) **One-Channel Hourglass**. This model only receives the cropped face image I_c without the eye-patch channel I_e. The hyper-parameters and losses however remain the same.

(6) **Upperface Squeezenet w/ decoder**. To test whether a secondary reconstruction task helps the encoder learn more discriminable representations, we also add a decoder (same architecture as (3)), to the SqueezeNet configuration (3).

As can be seen in Table I, increasing the input quality (*e.g.* landmarks vs. actual pixels) and model complexity (*e.g.* baseline CNN vs. residual encoder) also improves classification

TABLE I: Performance (ROC-AUC) of the different glance classification models on the MIT2013 dataset. The best two results are highlighted.

Model	Macro Average
Landmarks + MLP [18]	0.898 ± 0.001
Dense Eye Landmarks + Headpose [13]	0.800 ± 0.020
Baseline CNN [38]	0.953 ± 0.001
Upperface SqueezeNet [73]	0.960 ± 0.002
One-Channel Hourglass	0.961 ± 0.001
Upperface SqueezeNet [73] w/ Decoder	0.964 ± 0.001
Two-Channel Hourglass (ours)	**0.966 ± 0.001**
Personalized Hourglass (ours)	**0.967 ± 0.001**

performance, with both the personalization multi-stream and hourglass models outperforming the other approaches and the latter producing the best macro average ROC-AUC. This suggests providing the model with an additional stream of subject-specific information (*i.e.* personalization) can better tune the model with respect to movement of the driver head. Note that the model trained on dense eye landmarks performs poorly because the landmarks aren't localized accurately due to low input face resolution. Alternatively, adding an auxiliary reconstruction task can also boost the classification accuracy by learning useful contextual information while requiring no extra data stream, further underpinned by adding a decoder to the Squeezenet [28] architecture from [73]. Additionally, our model also outperform two recent approaches [73], [13] on gaze region estimation.

C. Domain Invariance

For the domain invariance task, as described in Section IV-C, we assign the MIT2013 dataset as the richly labeled domain $d1$ as it has a large number of video frames with human-annotated glance labels and use the AVT and our In-house datasets interchangeably as the new domain $d2$. To evaluate its effectiveness, we compare our multi-domain training approach with following regimes while keeping the backbone network (two-channel hourglass) the same:

(1) **Mixed Training**. Only labeled data from $d1$ and $d2$ are pooled together based on their glance labels for training.

(2) **Fine-tuning** [8]. We train the model on labeled data from $d1$ and then fine-tune the saved snapshot on labeled data from $d2$, a strategy similar to [8].

(3) **Gradient Reversal** [20]. We add a domain classification block on top of the encoder output to predict the domain of each input. However, its gradient is reversed during back-propagation to confuse the model and shift its representations towards a common manifold, similar to [20] (we use the implementation from [68]).

(4) **Tri-training** [83], [57]. We split the labeled data from $d1$ and $d2$ into three disjoint sets and train two model instances in a supervised manner independently with the first two splits. Then we use these two trained instances to predict the glance state for the samples from the remaining set. If the two predictions are in agreement, we assign it as the proxy-label to the sample and use it to train the third model. This model is finally used for inference.

(5) **Distillation** [26]. We split the data from both domains into two sets and train a teacher model on the first split in a supervised fashion. While we use labels for the $d1$ split to train a student model, we use the teacher's prediction on the unused split from $d2$ to regress the student to its output distribution. The trained student is used for inference.

TABLE II: Multi-domain performance (ROC-AUC) of our hourglass model, trained using different regimes, on the MIT2013 and AVT datasets. The best two results are highlighted.

Model	Macro Average
Mixed Training	**0.963 ± 0.001**, 0.918 ± 0.005
Fine-tuning [8]	0.875 ± 0.001, **0.920 ± 0.001**
Gradient Reversal [20]	0.961 ± 0.001, 0.912 ± 0.001
Tri-training [57]	0.956 ± 0.002, **0.929 ± 0.004**
Distillation [26]	0.961 ± 0.001, 0.895 ± 0.002
Ours	**0.964 ± 0.001**, 0.919 ± 0.001

TABLE III: Multi-domain performance (ROC-AUC) of our hourglass model, trained using different regimes, on the MIT2013 and the In-house dataset. The best two results are highlighted.

Model	Macro Average
Mixed Training	**0.956 ± 0.002**, **0.882 ± 0.002**
Fine-tuning [8]	0.807 ± 0.001, 0.858 ± 0.001
Gradient Reversal [20]	0.939 ± 0.002, 0.849 ± 0.003
Tri-training [57]	0.953 ± 0.001, 0.840 ± 0.005
Distillation [26]	0.945 ± 0.001, 0.830 ± 0.001
Ours	**0.962 ± 0.001**, **0.877 ± 0.003**

Although our multi-domain training approach can utilize the unlabeled samples from $d2$, for our first experiment we use 100% of the annotated images from both $d1$ and $d2$ to level the playing field. The same model snapshot is used for testing on both the MIT2013 dataset ($d1$) and the AVT or In-house datasets ($d2$). The results can be seen in Tables II and III respectively. In both cases, the fine-tuning approach fails to generalize to both domains, essentially "forgetting" details of the initial task (*i.e.* $d1$). Adding the gradient reversal head, does generate a boost over fine-tuning, however it overfits slightly on the training set and takes almost twice as the other approaches to converge. The tri-training and distillation based models generate competitive scores but fail to glean the full information from both domains due to the noise in the proxy labels. The mixed training and our multi-domain approaches perform competitively and generate the best two ROC-AUC numbers overall.

However, using all labeled data from the new domain does not fairly evaluate the full potential of our approach. Unlike the other approaches, our training regimen can utilize the unlabeled data (*i.e.* $(\mathrm{I}_{cu}^{d2} \oplus \mathrm{I}_{eu}^{d2})$) via the reconstruction loss, as proposed in Section IV-C. To put this functionality into effect, we use different amount of labeled samples (50%, 10% and 1%) from $d2$ during training the hourglass model with mixed training and multi-domain regimes. As shown in Table IV, our approach significantly outperforms mixed training as the amount of labeled data in the new domain diminishes. Interestingly, our multi-domain hourglass trained with 50% labeled data generalizes better than when trained with 100% labeled data suggesting more generalizable global features are learned when an unsupervised component is added to a classification task. Thus, this technique can gauge the amount of labeling required when adapting models to new domains and consequently reduce annotation cost.

TABLE IV: Performance (ROC-AUC) of our two-channel hourglass model with mixed training and our multi-domain regime, on the In-house dataset with different amount of labeled samples. The utility of unlabeled samples paired with reconstruction loss is evident as the percentage of labeled data from the second domain decreases.

Model (labeled data)	Centerstack	Instrument Cluster	Left	Rearview Mirror	Right	Road	Macro Average
Mixed Training (50%)	0.876	0.785	0.920	0.936	0.922	0.828	0.877
Ours (50%)	**0.897**	**0.818**	**0.944**	**0.943**	**0.938**	**0.842**	**0.897**
Mixed Training (10%)	0.821	0.734	0.882	0.867	**0.924**	0.798	0.838
Ours (10%)	**0.881**	**0.814**	**0.900**	**0.898**	0.893	**0.845**	**0.872**
Mixed Training (1%)	0.776	0.704	0.859	0.775	**0.854**	0.696	0.777
Ours (1%)	**0.830**	**0.790**	**0.904**	**0.847**	0.852	**0.777**	**0.833**

TABLE V: Performance (ROC-AUC) of our two-channel hourglass model with different components ablated on the MIT2013 dataset.

Model	Macro Average
w/ MSE	0.963 ± 0.001
w/o skip connections	0.961 ± 0.002
w/o L_{rec}	0.959 ± 0.001
w/o L_{cls} [8]	0.962 ± 0.001
Full Model	$\mathbf{0.966 \pm 0.001}$

D. Ablation Studies

To check the contribution of each component, we train the following variations of our hourglass model:
(1) **w/ MSE**. Instead of mean absolute error, the reconstruction loss is computed with mean squared error.
(2) **w/o skip connections**. We remove skip connections between the encoder and decoder layers.
(3) **w/o** L_{rec}. Reconstruction loss is removed, essentially making the model a classification based residual network.
(4) **w/o** L_{cls}. Taking inspiration from [8], we first train the hourglass solely with the reconstruction task (*i.e.* no L_{cls}) and then use the encoder module as a feature extractor to train the prediction block. For all the model variations, we keep everything else the same for consistency.

As presented in Table V, ablating the different components generates slightly different results. Due to the pixel normalization between $[-1, 1]$ before training, using MSE based reconstruction slightly dampens the error due to squaring. The skip connections help in propagating stronger signals across the network [56], hence removing them negatively affects model performance. Removing L_{rec} altogether deteriorates model performance as contextual information gets overlooked. Surprisingly, unsupervised pre-training performs quite well, suggesting reconstruction can teach the model features useful for classification. This reconstruction element helps the full model achieve the best overall performance.

VI. Conclusion

In this work, we proposed a model that takes as input a patch of the driver's face along with a crop of the eye-region and provides a classification into 6 coarse ROIs in the vehicle. We demonstrated that an hourglass network consisting of encoder-decoder modules, trained with a secondary reconstruction loss, allows the model to learn strong feature representations and perform better in the primary glance classification task. In order to make the system more robust to subject-specific variations in appearance and driving behavior, we proposed a multi-stream model that takes a representation of a driver's baseline glance behavior as an auxiliary input for learning residuals. Results indicate such personalized training to improve model performance for multiple glance ROIs over rigid models.

Finally, we designed a multi-domain training regime to jointly train our hourglass model on data collected from multiple camera views. Leveraging the hourglass' auxiliary reconstruction objective, this approach can learn domain invariant representations from very little labeled data in a weakly supervised manner, and consequently reduce labeling cost. As a future work, we plan to use our hourglass model as a proxy for annotating unlabeled data from new domains and actively learn from high confidence samples.

Acknowledgements: The AVT and MIT 2013 dataset used in this study were drawn from work supported by the Advanced Vehicle Technologies (AVT) Consortium at MIT (http://agelab.mit.edu/avt) and the Insurance Institute for Highway Safety (IIHS) respectively.

References

[1] Tobii. https://www.tobii.com/.
[2] M. Abadi et al. Tensorflow: A system for large-scale machine learning. In *OSDI Symposia*, 2016.
[3] S. Banerjee et al. On hallucinating context and background pixels from a face mask using multi-scale gans. In *WACV*, 2020.
[4] S. Banerjee, W. Scheirer, K. Bowyer, and P. Flynn. Fast face image synthesis with minimal training. In *WACV*, 2019.
[5] D. Blalock, J. J. Gonzalez Ortiz, J. Frankle, and J. Guttag. What is the state of neural network pruning? In *MLSys*, 2020.
[6] D. Cazzato, M. Leo, C. Distante, and H. Voos. When i look into your eyes: A survey on computer vision contributions for human gaze estimation and tracking. *Sensors*, 20(13):3739, 2020.
[7] Z. Chang et al. Salgaze: Personalizing gaze estimation using visual saliency. In *ICCV Workshops*, 2019.
[8] M. Chen, A. Radford, R. Child, J. Wu, H. Jun, D. Luan, and I. Sutskever. Generative pretraining from pixels. In *ICML*, 2020.
[9] R. Chhabra, S. Verma, and C.R. Krishna. A survey on driver behavior detection techniques for intelligent transportation systems. In *Confluence*, 2017.
[10] F. Chollet et al. Keras. https://github.com/fchollet/keras, 2015.
[11] W.S. Chu, F. De la Torre, and J. Cohn. Selective transfer machine for personalized facial action unit detection. In *CVPR*, 2013.
[12] J. Coughlin, B. Reimer, and B. Mehler. Monitoring, managing, and motivating driver safety and well-being. *Pervasive Computing*, 10(3):14–21, 2011.
[13] S. Dari, N. Kadrileev, and E. Hüllermeier. A neural network-based driver gaze classification system with vehicle signals. In *IJCNN*, 2020.
[14] J. Deng, J. Guo, N. Xue, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, 2019.
[15] X. Dong, S. Yu, X. Weng, S. Wei, Y. Yang, and Y. Sheikh. Supervision-by-registration: An unsupervised approach to improve the precision of facial landmark detectors. In *CVPR*, 2018.
[16] G. Fitch et al. The impact of hand-held and hands-free cell phone use on driving performance and safety-critical event risk. Technical report, 2013.
[17] L. Fridman et al. Mit advanced vehicle technology study: Large-scale naturalistic driving study of driver behavior and interaction with automation. *IEEE Access*, 7:102021–102038, 2019.

[18] L. Fridman, J. Lee, B. Reimer, and T. Victor. 'owl'and 'lizard': Patterns of head pose and eye pose in driver gaze classification. *IET Computer Vision*, 10(4):308–314, 2016.

[19] L. Fridman, B. Reimer, B. Mehler, and W. Freeman. Cognitive load estimation in the wild. In *CHI*, 2018.

[20] Y. Ganin et al. Domain-adversarial training of neural networks. *JMLR*, 17(59):1–35, 2016.

[21] T. Gebru, J. Hoffman, and F-F. Li. Fine-grained recognition in the wild: A multi-task domain adaptation approach. In *ICCV*, 2017.

[22] S. Ghosh, A. Dhall, G. Sharma, S. Gupta, and N. Sebe. Speak2label: Using domain knowledge for creating a large scale driver gaze zone estimation dataset. *arXiv:2004.05973*.

[23] D.W. Hansen and Q. Ji. In the eye of the beholder: A survey of models for eyes and gaze. *T-PAMI*, 32(3):478–500, 2009.

[24] M. Haris, G. Shakhnarovich, and N. Ukita. Task-driven super resolution: Object detection in low-resolution images. *arXiv:1803.11316*.

[25] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[26] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv:1503.02531*.

[27] M. Huh, P. Agarwal, and AA. Efros. What makes imagenet good for transfer learning? *arXiv:1608.08614*.

[28] FN. Iandola et al. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and ¡0.5mb model size. In *ICLR*, 2017.

[29] A. S. Jackson, A. Bulat, V. Argyriou, and G. Tzimiropoulos. Large pose 3d face reconstruction from a single image via direct volumetric cnn regression. *ICCV*, 2017.

[30] S. Jha and C. Busso. Probabilistic estimation of the gaze region of the driver using dense classification. In *ITSC*, pages 697–702, 2018.

[31] A. Joshi, S. Ghosh, M. Betke, S. Sclaroff, and H. Pfister. Personalizing gesture recognition using hierarchical bayesian neural networks. In *CVPR*, 2017.

[32] A. Joshi, S. Kyal, S. Banerjee, and T. Mishra. In-the-wild drowsiness detection from facial expressions. In *IV*, 2020.

[33] J. Kim, M. Stengel, A. Majercik, S. De Mello, D. Dunn, S. Laine, M. McGuire, and D. Luebke. Nvgaze: An anatomically-informed dataset for low-latency, near-eye gaze estimation. In *CHI*, 2019.

[34] D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[35] SG. Klauer et al. The impact of driver inattention on near-crash/crash risk: An analysis using the 100-car naturalistic driving study data. 2006.

[36] S. Kornblith, J. Shlens, and QV. Le. Do better imagenet models transfer better? In *CVPR*, 2019.

[37] K. Krafka, A. Khosla, P. Kellnhofer, H. Kannan, S. Bhandarkar, W. Matusik, and A. Torralba. Eye tracking for everyone. In *CVPR*, 2016.

[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012.

[39] Y. Liang, J. Lee, and L. Yekhshatyan. How dangerous is looking away from the road? algorithms predict crash risk from glance patterns in naturalistic driving. *Human Factors*, 54(6):1104–1116, 2012.

[40] E. Lindén, J. Sjostrand, and A. Proutiere. Learning to personalize in appearance-based gaze tracking. In *ICCV Workshops*, 2019.

[41] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and S. Le. Sphereface: Deep hypersphere embedding for face recognition. In *CVPR*, 2017.

[42] A. Makhzani, J. Shlens, N. Jaitly, and I. Goodfellow. Adversarial autoencoders. In *ICLR*, 2016.

[43] A. Mallya, D. Davis, and S. Lazebnik. Piggyback: Adapting a single network to multiple tasks by learning to mask weights. In *ECCV*, 2018.

[44] A. Mallya and S. Lazebnik. Packnet: Adding multiple tasks to a single network by iterative pruning. In *CVPR*, 2018.

[45] I. Masi, A. T. Tran, J. T. Leksut, T. Hassner, and G. Medioni. Do we really need to collect millions of faces for effective face recognition? In *ECCV*, 2016.

[46] B. Mehler et al. Multi-modal assessment of on-road demand of voice and manual phone calling and voice navigation entry across two embedded vehicle systems. *Ergonomics*, 59(3):344–367, 2016.

[47] A. Morando, P. Gershon, B. Mehler, and B. Reimer. Driver-initiated tesla autopilot disengagements in naturalistic driving. In *AutomotiveUI*, 2020.

[48] P. Morgado and N. Vasconcelos. Nettailor: Tuning the architecture, not just the weights. In *CVPR*, 2019.

[49] M. Mostajabi, M. Maire, and G. Shakhnarovich. Regularizing deep networks by modeling and predicting label structure. In *CVPR*, 2018.

[50] A. Newell, K. Yang, and J. Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016.

[51] X. Peng, Z. Huang, X. Sun, and K. Saenko. Domain agnostic learning with disentangled representations. In *ICML*, 2019.

[52] P.J. Phillips. A cross benchmark assessment of a deep convolutional neural network for face recognition. In *FG*, 2017.

[53] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*, 2016.

[54] A. Rangesh, B. Zhang, and M. Trivedi. Driver gaze estimation in the real world: Overcoming the eyeglass challenge. *arXiv:2002.02077*.

[55] A. Rice, P.J. Phillips, V. Natu, X. An, and A.J. O'Toole. Unaware person recognition from the body when face identification fails. *Psychological Science*, 24:2235–2243, 2013.

[56] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[57] S. Ruder and B. Plank. Strong baselines for neural semi-supervised learning under domain shift. In *ACL*, 2018.

[58] O. Russakovsky et al. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 115(3):211–252, 2015.

[59] K. Saenko, B. Kulis, M. Fritz, , and T. Darrell. Adapting visual category models to new domains. In *ECCV*, 2010.

[60] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training gans. In *NeurIPS*, 2016.

[61] B.D. Seppelt, S. Seaman, J. Lee, L.S. Angell, B. Mehler, and B. Reimer. Glass half-full: On-road glance metrics differentiate crashes from near-crashes in the 100-car data. *Accident Analysis & Prevention*, 107:48–62, 2017.

[62] V. Sharma, A. Diba, D. Neven, MS. Brown, L. Van Gool, and R. Stiefelhagen. Classification-driven dynamic image enhancement. In *CVPR*, 2018.

[63] W. Shi et al. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *CVPR*, 2016.

[64] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. Learning from simulated and unsupervised images through adversarial training. In *CVPR*, 2017.

[65] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[66] D.L. Smith et al. Methodology for capturing driver eye glance behavior during in-vehicle secondary tasks. *Transportation Research Record*, 1937(1):61–65, 2005.

[67] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, 15(56):1929–1958, 2014.

[68] M. Tonutti, E. Ruffaldi, A. Cattaneo, and CA. Avizzano. Robust and subject-independent driving manoeuvre anticipation through domain-adversarial recurrent neural networks. *Robotics and Autonomous Systems*, 115:162–173, 2019.

[69] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko. Simultaneous deep transfer across domains and tasks. In *ICCV*, 2015.

[70] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *CVPR*, 2017.

[71] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell. Deep domain confusion: Maximizing for domain invariance. *arXiv:1412.3474*.

[72] F. Vicente, Z. Huang, X. Xiong, F. De la Torre, W. Zhang, and D. Levi. Driver gaze tracking and eyes off the road detection system. *T-ITS*, 16(4):2014–2027, 2015.

[73] S. Vora, A. Rangesh, and M. Trivedi. Driver gaze zone estimation using convolutional neural networks: A general framework and ablative analysis. *T-IV*, 3(3):254–265, 2018.

[74] H. Wang et al. Cosface: Large margin cosine loss for deep face recognition. In *CVPR*, 2018.

[75] Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, 2016.

[76] E. Wood, T. Baltrušaitis, L.P. Morency, P. Robinson, and A. Bulling. Learning an appearance-based gaze estimator from one million synthesised images. In *ETRA*, 2016.

[77] E. Wood, T. Baltrusaitis, X. Zhang, Y. Sugano, P. Robinson, and A. Bulling. Rendering of eyes for eye-shape registration and gaze estimation. In *ICCV*, 2015.

[78] A. Yao, L. Van Gool, and P. Kohli. Gesture recognition portfolios for personalization. In *CVPR*, 2014.

[79] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016.

[80] Y. Yu, G. Liu, and J.M. Odobez. Improving few-shot user-specific gaze adaptation via gaze redirection synthesis. In *CVPR*, 2019.

[81] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling. Mpiigaze: Real-world dataset and deep appearance-based gaze estimation. *T-PAMI*, 41(1):162–175, 2017.

[82] Y. Zheng, DK. Pal, and M. Savvides. Ring loss: Convex feature normalization for face recognition. In *CVPR*, 2018.

[83] ZH. Zhou and M. Li. Tri-training: Exploiting unlabled data using three classifiers. *T-KDE.*, 17(11):1529–1541, 2005.