DriveAHead – A Large-Scale Driver Head Pose Dataset

Anke Schwarz[*1,2], Monica Haurilet[*1], Manuel Martinez[1], Rainer Stiefelhagen[1]

[1]Institute for Anthropomatics and Robotics, Karlsruhe Institute of Technology
[2]Robert Bosch GmbH

anke.schwarz@bosch.com, {haurilet, manuel.martinez, rainer.stiefelhagen}@kit.edu
https://cvhci.anthropomatik.kit.edu/data/DriveAHead/

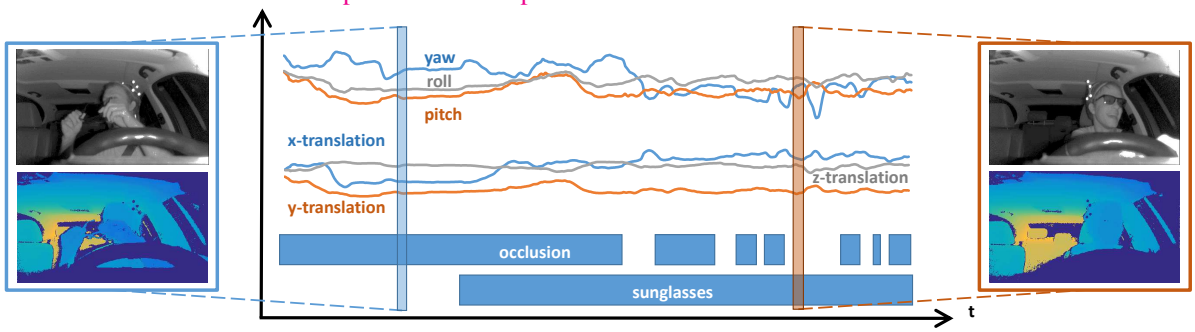

Figure 1: DriveAHead is a new dataset for head pose estimation in driving scenarios with one million depth and infrared images of 20 subjects captured while driving. Each frame is labeled for head position, head orientation and occlusions.

Abstract

Head pose monitoring is an important task for driver assistance systems, since it is a key indicator for human attention and behavior. However, current head pose datasets either lack complexity or do not adequately represent the conditions that occur while driving. Therefore, we introduce DriveAHead, a novel dataset designed to develop and evaluate head pose monitoring algorithms in real driving conditions. We provide frame-by-frame head pose labels obtained from a motion-capture system, as well as annotations about occlusions of the driver's face. To the best of our knowledge, DriveAHead is the largest publicly available driver head pose dataset, and also the only one that provides 2D and 3D data aligned at the pixel level using the Kinect v2. Existing performance metrics are based on the mean error without any consideration of the bias towards one position or another. Here, we suggest a new performance metric, named Balanced Mean Angular Error, that addresses the bias towards the forward looking position existing in driving datasets. Finally, we present the Head Pose Network, a deep learning model that achieves better performance than current state-of-the-art algorithms, and we analyze its performance when using our dataset.

1. Introduction

In the automotive sector, computer vision allows driver assistance systems to monitor the interior of the vehicle and its occupants. To understand the driver's intentions and predict future actions, gaze direction plays a crucial role. However, in car environments eye gaze is difficult to estimate due to occlusions or large head rotations. Thus, the orientation and position of the drivers head is used to approximate the gaze direction. To aid in the task of developing and evaluating head pose algorithms in driving scenarios, we present the DriveAHead Dataset (see Figure 1).

DriveAHead collects data from 20 subjects while driving. The average length of the sequence is around 30 minutes and includes parking maneuvers, driving on the highway and through a small town. Reference pose measurements are recorded with a motion capture system that tracks the orientation and position of the drivers head. We also include frame-by-frame labels for occlusions, glasses, and sunglasses. Typically, images from color cameras exhibit strong variations due to ambient illumination encountered while driving. Images captured using active illumination, as used by the Kinect v2, do not have these variations.

DriveAHead provides 5 times more images than the second largest available driver head pose dataset (Lisa-P [25]), while having at least as many subjects as other available datasets except for Bosphorus [33], which was captured in

[*]Both authors contributed equally to this work.

a laboratory setting.

We use DriveAHead to evaluate the performance of current state-of-the-art head pose algorithms. Furthermore, we train neural network models for head rotation estimation on both depth and infrared images. To tackle the bias towards forward looking while driving, we also provide a new performance metric, the Balanced Mean Angular Error (BMAE). Our training experiments reveal that DriveAHead is complex and large enough to train deep neural networks. In addition, depth frames generally provide better head pose accuracy than 2D data, and when using both we see further improvement of our results.

DriveAHead is the first publicly available real-world driving dataset containing aligned depth and IR images with absolute head position and orientation annotations. The large-scale of the dataset allowed us to create a deep learning model with state-of-the-art performance for the driving scenario.

2. Related work

Since head pose estimation has a wide spectrum of applications, it has received much attention from the research community. Various head pose datasets are available. In the following section, we discuss the most relevant ones to our dataset, see Table 1 for an overview.

2.1. Non-driving head pose datasets

BU [22] is a head pose dataset containing 15k RGB images of 5 subjects. The dataset is split into sequences with uniform lighting conditions and more complex video data where the scenes were exposed to varying lighting conditions. This dataset was recorded in a lab and it provides continuous head orientation and translation measurements using a magnetic sensor attached to the subject's head.

In contrast, the *Pointing'04* [13] gaze direction dataset has three times more subjects, but contains far less number of images. Pointing'04 does not include continuous head pose measurements but is organized into 93 discrete head orientations, totaling 30 video sequences.

Like *Pointing'04*, the *Bosphorus* [33] dataset contains only discrete head orientation labels. With 105 different subjects, the *Bosphorus* dataset has the largest number of subjects. In addition to RGB data, a structured-light based scanner provides 3D data. The in-lab dataset also contains various types of occlusion labels: hair, glasses and self-occlusion.

The *BIWI* [11] dataset provides 15k depth and RGB data including orientation and position head measurements. The dataset is recorded inside a lab environment with annotations estimated using a template based approach [40]. While the template based method provides accurate ground truth for translation and orientation in constrained environments there is a limitation in case of strong occlusions.

Similar to BIWI, *ICT 3dHP* [4] uses the Kinect v1 sensor to record depth and RGB images. The head orientation and position reference measurements are established with a magnetic sensor mounted on the subject's head.

The *GI4E* [3] head pose dataset records the head orientation and position data with a magnetic sensor as well. The dataset provides multiple RGB video sequences of 10 different subjects.

2.2. Driving head pose datasets

The *Lisa-P* [25] dataset provides 200k RGB images obtained from 14 subjects in real driving scenarios. Their head orientation data is obtained from motion capture markers on the back of the driver's head. However, this method requires removing the headrest, which poses a safety risk.

In comparison, *CoHMet* [39] uses an inertial sensor for head pose measurements. Variations due to drift have to be removed manually every 10 seconds. Furthermore, both datasets only include orientation data.

2.3. Head pose estimation algorithms

Next, we compare various methods for head pose estimation and discuss state-of-the-art deep learning algorithms.

Head pose estimation. In the following, we group head pose estimation approaches based on the input data. Most published *2D algorithms* either discretize the head pose space and perform merely a classification task [16, 17] or estimate the head pose from facial points. To find an accurate head pose from facial landmarks, statistical models are applied such as Active Appearance Models [10] or multi-view AAMS [27]. Since one of the key difficulties in these type of approaches are uncontrolled illuminations variations, the authors of [29] propose Asymmetric Appearance Modeling to overcome this issue. Breitenstein *et al*. [9] present a *3D method* to estimate the nose location by applying 3D shape signatures. The head position and orientation are calculated iteratively by minimizing a cost function. In their later work, they extend their approach for low resolution depth data from stereo cameras [8]. Both methods require GPUs to reach real-time performance. With the advent of affordable depth sensors, methods using distance values as an input have increased. In [11] the authors propose decision forests to find the head orientation and position in a frame-by-frame manner. This idea is extended in several works [28, 34]. An efficient method that applies a global linear mapping to local binary features is proposed in [35]. Another idea to reduce the computational cost and obtain accurate results is by performing subject-specific tracking [24, 38]. All these methods have so far been evaluated only in constrained in-lab environments. To close the gap towards evaluation on challenging data, our dataset provides real driving scenarios including occlusions.

	BU [22]	Pointing'04 [13]	Bosphorus [33]	BIWI [11, 12]	ICT 3dHP [4]	Lisa P [25]	CoHMEt [39]	GI4E [3]	DriveAHead
Year	2000	2004	2008	2011	2012	2012	2014	2016	2017
Driving	–	–	–	–	–	✓	✓	–	✓
Publicly available	✓	✓	✓	✓	✓	✓	–	✓	✓
RGB/grayscale	✓	✓	✓	✓	✓	✓	✓	✓	–
Depth	–	–	✓	✓	✓	–	–	–	✓
IR	–	–	–	–	–	–	–	–	✓
Video	✓	✓	–	✓	✓	✓	✓	✓	✓
Resolution	320×240	384×288	1600×1200	640×480	640×480	640×480	640×360	1280×720	512×424
Pixel aligned	N/A[b]	N/A[b]	✓	✓[d]	✓[d]	N/A[b]	N/A[b]	N/A[b]	✓
N° subjects	5	15	105	20	10	14	N/A[c]	10	20
N° images	15k	3k	5k	15k	14k	200k	90k	36k	1M
Female / male	0 / 5	1 / 14	45 / 60	6 / 14	6 / 4	N/A[c]	N/A[c]	4 / 6	4 / 16
N° video sequences	72	30	N/A[a]	24	10	14	N/A[c]	120	21
Glasses labels	–	–	✓	–	–	–	–	–	✓
Sunglasses labels	–	–	–	–	–	–	–	–	✓
Occlusion labels	–	–	✓	–	–	–	–	–	✓
Reference system	magnetic	marker	guided	Faceshift [1]	magnetic	mo-cap	inertial	magnetic	mo-cap
Continuous labels	✓	–	–	✓	✓	✓	✓	✓	✓
Head orientation labels	✓	✓	✓	✓	✓	✓	✓	✓	✓
Head position labels	✓	–	✓	✓	✓	–	–	✓	✓

[a] not applicable since we do not have video
[b] not applicable since we only have a single image modality
[c] information not provided by the authors
[d] transformation between depth and RGB modality available

Table 1: A comparison of various driving and non-driving head pose datasets. This table depicts the characteristics of the recording modalities, the content of the dataset and the properties of the provided reference labels.

Deep learning methods. On the ImageNet Visual Recognition Challenge (ILSVRC), neural networks have shown ground-breaking improvements on the difficult task of classifying images into one of 1000 possible object categories [31]. The first neural network model that won the competition was AlexNet [21]. It outperformed other methods by a significant margin. In subsequent years, other neural networks were able to achieve state-of-the-art results on ImageNet: *e.g.* VGG [36], GoogLeNet [37] and ResNet [15]. Furthermore, Misra *et al.* [26] introduced a novel sharing unit called cross-stitch for multi-task learning. More precisely, the authors showed that, by using two different models that exchange information using the sharing units, they are able to improve the performance on semantic segmentation and surface normal prediction.

So far, only a few works addressed deep learning [2, 30] for head pose estimation. The authors in [7] presented a CNN that can be trained on coarse regression labels and predict continuous head pose on the full range of 360 degrees. In [30] hough networks combine the idea of hough forests and convolutional neural networks. The availability of our large-scale dataset has the potential to apply state-of-the-art deep learning methods for head pose estimation.

3. DriveAHead dataset

We introduce DriveAHead – a head pose dataset captured while driving. We capture our dataset using a Kinect v2 sensor, because it provides us with both infrared and depth images. To measure the head orientation and position we use an accurate 3D motion capture system. This motion capture system tracks the orientation and position of a head target. The target consists of several 3D markers which form a spatial shape (see Figure 3, column 1 and 3). The output of the motion capture system is time synchronized with the Kinect v2 recording device. In total we record 21 sequences of 20 different subjects. 30 minute sequences include parking maneuvers, driving on a highway and a through a small town. During the driving sequence most of the drivers stopped to change between no glasses or glasses to sunglasses. The driving sequences during daytime included different weather conditions sunny, foggy and rainy. In this section, we describe our data annotation method consisting of continuous head pose labels and binary description labels as previously shown in Figure 1. Subsequently, we present a dataset analysis and the evaluation metrics.

3.1. Data annotation

In the following section, we include a detailed description of our reference system providing the orientation and position of the drivers' head.

Head coordinate system. A unique head coordinate system definition is crucial for absolute head position and orientation measurements. We present a facial landmark based definition of the head coordinate system. Existing definitions are either dependent on the complete 3D facial shape with the origin in the center of the face [11] or

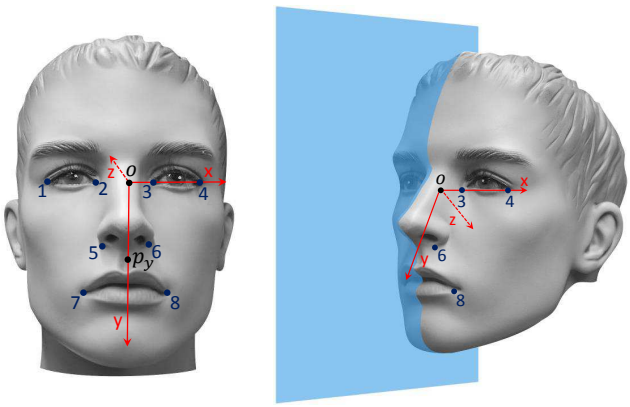

Figure 2: Description of our Head Coordinate System definition based on measured 3D facial landmark positions.

point-to-point mapping of general 3D facial landmark positions [5]. In contrast to a complete 3D facial shape, we propose a definition that is only dependent on facial landmarks, which are visible during frontal views. Compared to the point-to-point mapping, our definition allows the calculation of a unique transformation directly from the individual facial landmarks. Figure 2 shows our Cartesian Coordinate System with the axis and 3D landmark positions $l_{i\in\{1..8\}} \in \mathbb{R}^3$. We define the *head origin* as the center point between the eyes. We calculate it by averaging the four eye corners (see Figure 2):

$$o := \frac{l_1 + l_2 + l_3 + l_4}{4}.$$

Furthermore, the *x-axis* has the origin in o and points in the direction of the vector p_x that spans between the middle point of the two eyes:

$$p_x := \theta_3 + \theta_4 - (\theta_1 + \theta_2),$$

where

$$\theta_i := \frac{l_i - o}{\|l_i - o\|}.$$

Each vector is normalized to achieve a stable definition against error prone measurements.

Similar to the x-axis, we define the *y-axis* using the origin o and span it to a chosen point on the person's head. In this case, we define this point as the middle point of the mouth and nose corners:

$$p_y := \frac{l_5 + l_6 + l_7 + l_8}{4}$$

However, since we define a Cartesian coordinate system, we have to make sure that all axis are pairwise perpendicular and the intersection of them lies in the origin. Therefore, we span a plane perpendicular to the x-axis (see Figure 2) and by choosing a vector on this plane we guarantee that the vector is perpendicular to the x-axis. Thus, we simply pick the vector with the origin in o with the direction to the closest point to our previously defined p_y

Finally, the *z-axis* is the vector, perpendicular to the x and y-axis, pointing towards the face:

$$z := x \times y$$

Head pose reference. Next, we describe the steps to calculate the rotation and translation for each frame i from the camera coordinate system (c) to head coordinate system (h). We define this transformation function with $T_i^{c\to h}$. To calculate this transformation, we make use of two transformation functions that we define next.

First, our reference system measures the orientation and translation of the target fixed on the head. This target consists of several 3D markers which are tracked from the motion capture system. The system provides for each frame i the transformation from the camera coordinate system to the target coordinate system: $T_i^{c\to t}$.

Second, we require the projection of the target coordinate system into the head coordinate system. This handles the issue that the target has a different position depending on the shape of the head. We define this as $T_n^{h\to t}$ for each subject n individually. To calculate $T_n^{t\to h}$, 3D positions of several facial landmarks are accurately measured using a special motion capture target. The measured facial landmarks define the orientation and position of this transformation as previously described (see Figure 2).

Finally, we obtain the ground truth head orientation and position by combining both transformations:

$$T_i^{c\to h} := T_i^{c\to t} \cdot T_n^{t\to h}.$$

Occlusion annotations. In addition to the measured head orientation and position, we provide description labels for each image. Supplemental to the measured head orientation and position, we provide for each image description labels. We asked annotators to provide binary labels for each image which show if the driver is wearing sunglasses or glasses. Furthermore, for each image we add manual annotations about occlusions. We treat a face as occluded if at least one of the 68 facial landmarks defined in [32] is not visible (see right image in Figure 3 for an example). Faces with self occlusions due to large rotations are not marked as occluded. Subjects wearing sunglasses or glasses are annotated with additional labels and we do not count them as 'occlusions'.

Dataset splits. We divide our sequences recorded on 20 different subjects into training, validation and testing sets. The first five subject are used for testing, while the latter are included in the training set. From the training set we use two subjects in our validation set for parameter tuning. To

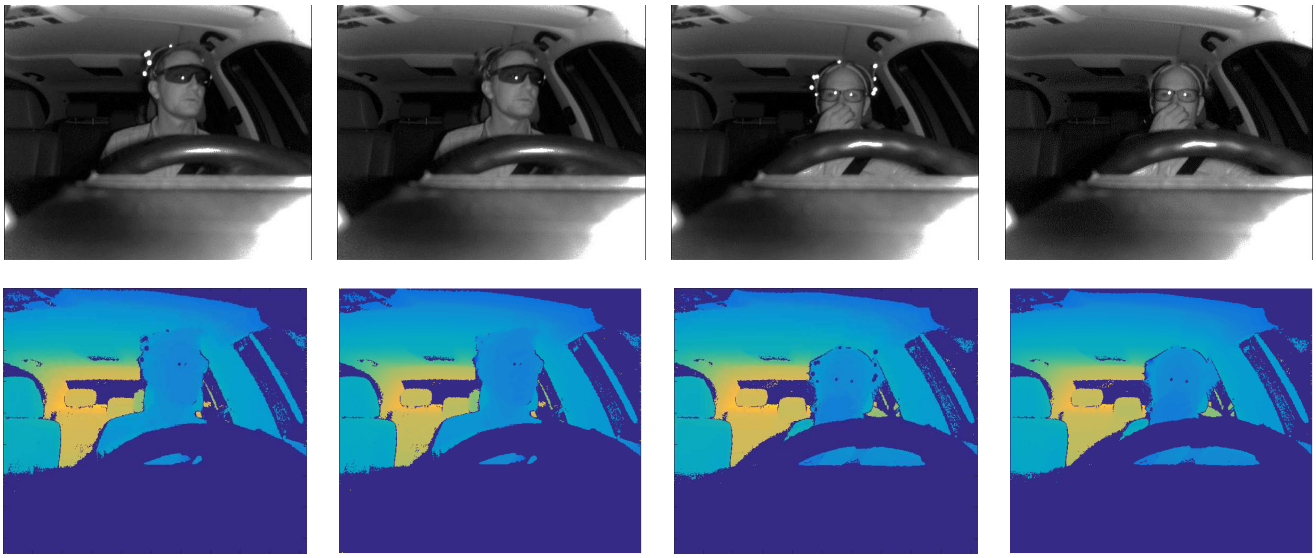

Figure 3: Example images from DriveAHead before (1st and 3rd column) and after preprocessing (2nd and 4th column). Our preprocessing step removes the white markers by interpolating the pixels of the surrounding.

obtain well-adjusted parts, in our splits we include both female and male subjects, as well as subjects wearing glasses and sunglasses.

Preprocessing of our data. To overcome the issue of the visible white markers we preprocess our training data. We eliminate these in the training and validation set by filling these regions with the interpolated values of the surrounding of the markers (see Figure 3). The motion capture system provides us the translation and rotation of the head target for each frame. Hence, we know 2D locations of the markers in the infrared and depth images. The surrounding of the marker locations is interpolated to achieve smooth regions. This ensures that the models are not able to learn the head pose based on the locations of the markers. In case of the test data, we use the raw data to even eliminate methods learning the interpolated areas.

3.2. Dataset analysis

Next, we describe the dataset statistics and discuss other available head pose datasets. Figure 3 shows DriveAHead samples of the aligned depth and infrared images. As shown in Table 1 our dataset consists of 4 female and 16 male subjects. The dataset contains 21 sequences of 20 different subjects, one subject is recorded twice. The resolution of the depth and infrared images is 512×424 with an average inter pupil distance of 35 pixels. Figure 4 shows the distribution of the data amount dependent on yaw, roll and pitch angle.

While the yaw and roll angles are centered at zero degrees, the pitch is slightly shifted. Furthermore, we manually labeled all images into three categories: glasses, sunglasses and other types of occlusion. We call other types of occlusion simply 'occlusion', while with 'None' we mean faces that do not wear neither glasses or sunglasses. Figure 5 shows the number of faces grouped into these categories. Around 26% of faces are occluded by other objects than glasses and sunglasses. In more than one third of the dataset drivers are wearing sunglasses, while around 20% of the faces have glasses.

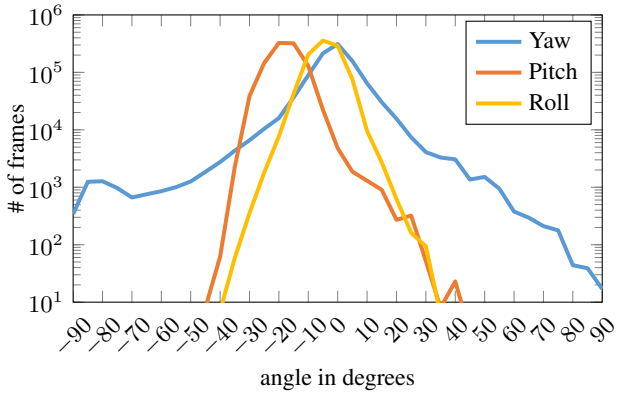

Figure 4: Histogram of DriveAHead frames for yaw, roll and pitch angle.

3.3. Evaluation metrics

In the following, we provide our evaluation metrics to rate the performance of our models on both head rotation and the translation task.

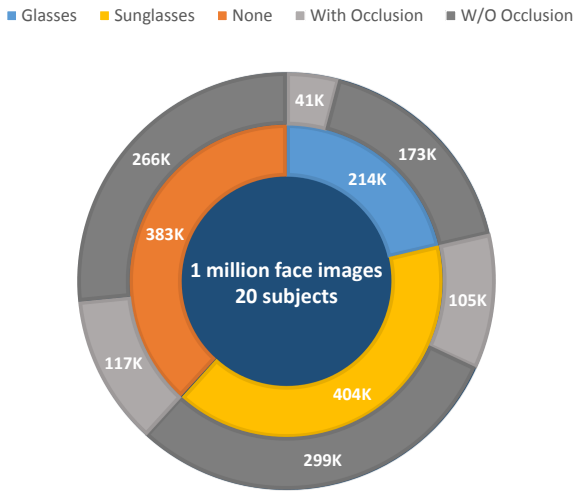

Figure 5: The inner ring of the chart shows the number of faces that are wearing glasses or sunglasses in our dataset. The field 'None' represents the number of images in our dataset with faces that do not wear glass or sunglasses. While, the outer ring represents additional occlusions.

Head rotation task. To evaluate the head rotation, we rely on the angular error between the estimated quaternion q_{est} and the ground truth quaternion q_{gt}:

$$ae(q_{est}, q_{gt}) := 2 \cdot \arccos(< q_{est}, q_{gt} >).$$

The orientations during driving scenarios are biased towards frontal orientation, which leads to an unbalanced amount of different head orientations. To address this, we introduce the Balanced Mean Angular Error (BMAE) metric with:

$$\text{BMAE} := \frac{d}{k} \sum_i \phi_{i,i+d}, i \in d\mathbb{N} \cap [0, k],$$

where $\phi_{i,i+d}$ is the average angular error $ae(q_{est}, q_{gt})$, for all data points where the absolute distance of ground truth angle q_{gt} to the zero quaternion lies between i and $i + d$. During our evaluation, we set the section size to $d = 5$ degrees and k to 75 degrees.

Head translation task. In the case of head translation, we opt to use the euclidean distance to measure the performance of our models. The error is provided in millimeters.

4. Baseline methods

In this section, we describe the head pose methods that we evaluate on our dataset. We categorize these algorithms in conventional methods that were previously applied for head pose estimation, and neural networks. To detect the face regions, we use for all methods the face detector of [19].

4.1. Conventional methods

Additionally to the prior, we use two methods for head pose estimation: a method that uses grayscale data and a depth-based approach.

Prior. To show the difficulty of our data we present the results based on the prior. The prior always predicts the same rotation and translation regardless of the input image. More precisely, this method always outputs the average rotation ($yaw : 0.9°, pitch : -15.3°, roll : -1.8°$) and translation $\vec{t} = (50.7, -143.2, 679.4)$ *mm* calculated from our training set.

Openface [5]. Second, we choose to evaluate a method, which estimates the head orientation and position based on facial landmark tracking from grayscale data. For facial landmark detection Conditional Local Neural Fields (CLNF) are used. The head pose is calculated using point-to-point correspondences of a 3D landmark shape. To perform a fair comparison we transformed the 3D landmark shape into the head coordinate system definition described in section 3.1. The online available approach is trained on various RGB datasets (Multi-PIE [14], LFPW [6]) and Helen [23]).

HeHOP [35]. Furthermore, we evaluate a model that is trained solely on depth data. This approach trains random forests on local areas to obtain binary features. Using these binary features a global linear mapping finds the head orientation and position. We train this method on our training set with the same parameters as proposed in the work of Schwarz *et al.* [35].

4.2. Deep learning methods

Since only few deep models for head rotation estimation are available, we supplementary evaluate deep learning models that were initially proposed for object classification. However, since in our case head pose estimation is a regression task, we make several adjustments to these models.

Data augmentation. First, we rescale all detected faces to a size of 91×70 and obtain the final input by randomly cropping image patches of 88×67. As our datasets includes both depth and infrared images, we will show results using both modalities.

Models. We introduce a novel head pose network (HPN) that uses similar to VGG [36] convolutions of size 3×3. However, our network has a far smaller number of parameters, since we only use half of the number of neural filters with 4 maxpooling layers. Furthermore, we have two fully connected layers each with 2048 neurons and an output layer of size 4. We make use of the fact that we have two input modalities by combining the depth and IR models using cross-stitch units [26] after each maxpooling layer. In comparison to the work of Misra *et al.* [26] that uses these sharing units for multi-task learning, we use them to combine different image modalities. However, we show

that these sharing units can also greatly increase the performance of our HPN model for occluded faces. We compare our model with the N2 network proposed by Ahn *et al*. [2] that we trained in the same way and with the same data as ours.

Loss function. An important aspect of training neural networks is the loss function. Since in our case we have a regression problem, we can not use the cross entropy. An option to define our loss function is to use some distance metric between the Euler angles [2]. However, Euler angles suffer from the gimbal lock problem. Thus, we opt to calculate our loss function using quaternions. We define our loss function between the predicted quaternions $p \in \mathbb{R}^4$ and the ground truth g as:

$$\ell_\gamma(g, p) \coloneqq \|g - n_p\| + \gamma \cdot \alpha(p),$$

where n_p is the normalization of p. In comparison to [18], we use the second term: $\alpha(p) = \|p\|$ for regularization *i.e.* to prevent our predictions to grow too large.

Implementation details. For initializing the weights of our models, we randomly sample from a normal distribution, while we set the biases to zero. We evaluated models both with (*i.e.* $\gamma > 0$) and without any regularization (*i.e.* $\gamma = 0$). However, as we have experienced overflow when removing the regularization, we solely present the results with $\gamma > 0$. As optimizer we use Adam (Adaptive Moment Estimation [20]) with an initial learning rate of 0.001. Finally, we train our models for 600k iterations with mini-batches of size 32.

5. Evaluation

In this section, we show the results of the models we have trained on depth values, IR data and on both modalities. We also discuss how well our models are able to cope with large rotations and occlusions (*i.e.* glasses, sunglasses and other types of occlusion).

5.1. Head rotation estimation.

In this section, we show the results of multiple head rotation models and the impact of occlusions on their performance.

Comparison of different head pose methods. In the first row of Table 2 we show the BMAE of the prior. As mentioned previously, we define as prior the average rotation of the faces in our training set. HeHOP [35] was trained on our depth data and obtained an error far smaller than the prior. In comparison, Openface* [5] that was trained on RGB images from different datasets, shows a BMAE of 20.6 on our IR data. The deep learning models have outperformed the 'conventional' approaches, namely, our HPN model achieves a BMAE of only 16.4 for IR and 14.2 for depth data.

Method	Modality	Fusion	All	Occlusion
Prior	–	–	35.7	37.7
Openface[a][5]	IR	–	20.6	22.7
HeHOP [35]	Depth	–	26.3	30.1
N2 [2]	Depth	–	15.9	18.8
N2 [2]	IR	–	19.2	22.6
N2 [2]	Both	Late F.	16.7	19.7
N2 [2]	Both	Conc.	19.0	22.2
HPN	Depth	–	14.2	16.9
HPN	IR	–	16.4	20.5
HPN	Both	Late F.	**13.4**	17.0
HPN	Both	Conc.	17.4	21.7
HPN	Both	Stitch	13.7	**16.0**

[a] pretrained version from [5].

Table 2: Balanced Mean Angular Error (*BMAE*) of different methods evaluated on our dataset using depth and IR data.

Fusing depth and IR data. Since our datasets include both depth and IR data, a fusion between both modalities have the potential to further increase the performance of our model. Late fusion is one of the possible methods to combine our two modalities. This consists of simply averaging the prediction of a depth and a IR based approach. In contrast, the concatenation is an early fusion modality, where we simply concatenate on the channel dimension the depth with the IR data. Finally, we include our HPN model with cross-stitch units (introduced in [26]).

Table 2 shows the results of our fusion efforts on both HPN and on N2 [2]. Late fusion and the stitch units were able to improve the performance of HPN, while concatenation degrades the performance. As we see in Table 2 the late fusion was able to improve our model to a BMAE of 13.4, while the stitch units have a BMAE of 13.7. In case of the N2 model, the fusion has not performed as well, the concatenation and late fusion even worsen the angular error. Finally, our HPN model using stitch units has outperformed all other methods with a significant margin in the case for faces containing occlusions.

Impact of occlusions on the models. The right most column of Table 2 presents the performance of all methods on occluded faces. First, the BMAE of the prior is higher on occluded faces compared to all faces. The reason behind is that occlusions are more frequently accompanied by large rotations for which our models obtain a higher error. Since HeHOP [35] relies on a linear model we see a large drop of performance in case of occlusion. In comparison, Openface* [5] experience only a small drop in performance for occluded faces. In case of the neural networks, both the depth based and as well the IR models show a higher BMAE. However, the stitch-based HPN are able to outper-

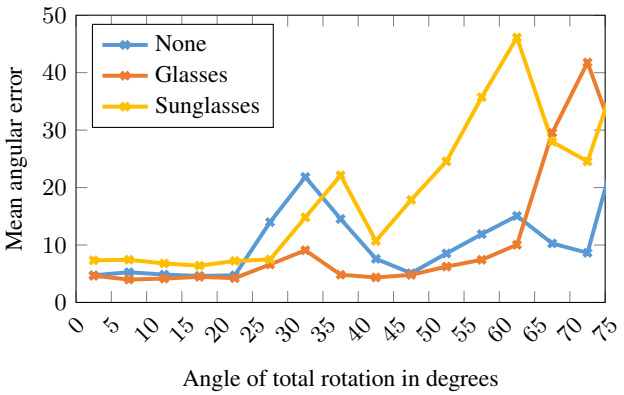

Figure 6: The angular error of HPN with late fusion for drivers wearing glasses, sunglasses and without any accessories. For each interval of 5 degrees the mean angular error is visualized.

form all other models, achieving a BMAE of only 16.0, showing the effectiveness of this fusion modality.

Glasses vs. sunglasses. In Figure 6, we show the performance of our head model HPN with late fusion in regard to different amount of total rotations. We partition the amount of total rotation into sections of 5 degrees and visualize the mean angular error of each section. With this figure we compare the performance of our algorithm for faces wearing sunglasses and glasses, and for all the faces that do not contain these types of occlusion ('none'). While the sunglasses greatly worsen the results of our model, the glasses were even able to improve the head pose estimation in case of small rotations. A reason behind these results is that the glasses only occlude a small portion of the driver's face, and thus both the face and the glasses themselves can be used as a cue to calculate the head orientation. From this it follows that glasses are a highly beneficial feature for our model, leading in an improvement of the performance.

5.2. Head position estimation

The head position is important for precise gaze direction. To evaluate the performance of the head orientation, we split the error in x, y and z direction. In Table 3 we show the results of a method using solely depth data [35] and a model performing solely on IR data [5]. Whereas the depth model is trained on the training part of this dataset, the other method is trained on various other RGB datasets. The table shows that for both approaches, the z-direction is most challenging. Since depth data as an input cue gives direct information about the 3D location of the surface, overall the depth method outperforms the IR method. However, it is worth to notice that the 2D approach performs competitive for the x and y direction.

Method	All			Occlusion		
	x	y	z	x	y	z
Prior	30.9	20.2	36.8	37.8	21.4	51.7
Openface[a] [5]	6.4	7.6	27.8	8.5	6.1	47.4
HeHOP [35]	4.1	3.6	5.3	5.9	4.2	6.7

[a] pretrained version from [5].

Table 3: Euclidean distance in *mm* of the translation predictions and ground truth. In the first column we show the error when using our whole test set, while in the second one we show the results for occluded faces only.

6. Conclusion and outlook

In this work, we present DriveAHead, a large-scale dataset for driver head pose estimation. Our dataset contains more than 10 hours of infrared (IR) and depth images of drivers' head poses taken in real driving situations. Together with precise head pose measurements we provide also manual annotations about a number of occlusion types. It can be used to evaluate head pose estimation approaches under realistic driving conditions. Also, due to its size, it facilitates the training of deep learning based models for head pose estimation.

In our paper, we thoroughly discuss the dataset, its acquisition methods, the calibration and alignment of the data, and its overall statistics. In addition, we evaluate a number of state of the art and baseline head pose estimation methods on the dataset. Moreover, we train several deep-learning based head pose estimation models and show that the here proposed deep pose estimation network HPN provides state-of-the-art results. We evaluate the pose estimation methods on both IR and depth images, and we investigate various fusion methods for the two image modalities. Here, our HPN model using cross-stitch units provides the best results on occluded faces while late fusion performs best on all faces.

While the evaluation shows that generally promising head pose estimation results can be achieved on realistic data, the analysis of the results also reveals that several challenges still remain. In particular for large head rotations and when faces get occluded, which is the case in more than 25 percent of our dataset, head pose estimation models have multiple difficulties. We hope that our dataset will facilitate the development and evaluation of head pose estimation methods addressing a realistic driver scenario.

References

[1] Faceshift. www.faceshift.com. 3

[2] B. Ahn, J. Park, and I. S. Kweon. Real-time head orientation from a monocular camera using deep neural network. In *ACCV*. 2014. 3, 7

[3] M. Ariz, J. J. Bengoechea, A. Villanueva, and R. Cabeza. A novel 2d/3d database with automatic face annotation for head tracking and pose estimation. In *Computer Vision and Image Understanding*, 2016. 2, 3

[4] T. Baltrušaitis, P. Robinson, and L.-P. Morency. 3d constrained local model for rigid and non-rigid facial tracking. In *CVPR*, 2012. 2, 3

[5] T. Baltrušaitis, P. Robinson, and L.-P. Morency. Openface: an open source facial behavior analysis toolkit. In *WACV*, 2016. 4, 6, 7, 8

[6] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using a consensus of exemplars. In *TPAMI*, 2013. 6

[7] L. Beyer, A. Hermans, and B. Leibe. Biternion nets: Continuous head pose regression from discrete training labels. In *GCPR*, 2015. 3

[8] M. D. Breitenstein, J. Jensen, C. Høilund, T. B. Moeslund, and L. Van Gool. Head pose estimation from passive stereo images. In *Scandinavian conference on image analysis*, 2009. 2

[9] M. D. Breitenstein, D. Kuettel, T. Weise, and L. van Gool. Real-time face pose estimation from single range images. In *CVPR*, 2008. 2

[10] T. F. Cootes, G. J. Edwards, C. J. Taylor, et al. Active appearance models. In *TPAMI*, 2001. 2

[11] G. Fanelli, M. Dantone, J. Gall, A. Fossati, and L. Gool. Random forests for real time 3d face analysis. In *IJCV*, 2013. 2, 3

[12] G. Fanelli, J. Gall, and L. van Gool. Real time head pose estimation with random regression forests. In *CVPR*, 2011. 3

[13] N. Gourier, D. Hall, and J. L. Crowley. Estimating face orientation from robust detection of salient facial structures. In *FG Net Workshop on Visual Observation of Deictic Gestures*, 2004. 2, 3

[14] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-pie. *Image and Vision Computing*, 2010. 6

[15] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *ECCV*, 2016. 3

[16] C. Huang, X. Ding, and C. Fang. Head pose estimation based on random forests for multiclass classification. In *ICPR*, 2010. 2

[17] M. Kan, S. Shan, H. Chang, and X. Chen. Stacked progressive auto-encoders (spae) for face recognition across poses. In *CVPR*, 2014. 2

[18] A. Kendall, M. Grimes, and R. Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *ICCV*, 2015. 7

[19] D. E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 2009. 6

[20] D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *arXiv preprint arXiv:1412.6980*, 2014. 7

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 3

[22] M. La Cascia, S. Sclaroff, and V. Athitsos. Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3d models. *TPAMI*, 2000. 2, 3

[23] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang. Interactive facial feature localization. In *ECCV*, 2012. 6

[24] M. Martin, F. Van De Camp, and R. Stiefelhagen. Real time head model creation and head pose estimation on consumer depth cameras. In *3DV*, 2014. 2

[25] S. Martin, A. Tawari, E. Murphy-Chutorian, S. Y. Cheng, and M. Trivedi. On the design and evaluation of robust head pose for visual user interfaces: Algorithms, databases, and comparisons. In *ACM*, 2012. 1, 2, 3

[26] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert. Cross-stitch networks for multi-task learning. In *arXiv preprint arXiv:1604.03539*, 2016. 3, 6, 7

[27] K. Ramnath, S. Koterba, J. Xiao, C. Hu, I. Matthews, S. Baker, J. Cohn, and T. Kanade. Multi-view aam fitting and construction. *IJCV*, 2008. 2

[28] C. Redondo-Cabrera, R. López-Sastre, and T. Tuytelaars. All together now: Simultaneous object detection and continuous pose estimation using a hough forest with probabilistic locally enhanced voting. In *BMVC*, 2014. 2

[29] M. Rezaei and R. Klette. Look at the driver, look at the road: No distraction! no accident! In *CVPR*, 2014. 2

[30] G. Riegler, D. Ferstl, M. Rther, and H. Bischof. Hough networks for head pose estimation and facial feature localization. 2014. 3

[31] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 3

[32] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *ICCVW*, 2013. 4

[33] A. Savran, N. Alyüz, H. Dibeklioğlu, O. Çeliktutan, B. Gökberk, B. Sankur, and L. Akarun. Bosphorus database for 3d face analysis. In *European Workshop on Biometrics and Identity Management*, 2008. 1, 2, 3

[34] S. Schulter, C. Leistner, P. Wohlhart, P. M. Roth, and H. Bischof. Alternating regression forests for object detection and pose estimation. In *ICCV*, 2013. 2

[35] A. Schwarz, Z. Lin, and R. Stiefelhagen. Hehop: Highly efficient head orientation and position estimation. In *WACV*, 2016. 2, 6, 7, 8

[36] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 3, 6

[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 3

[38] D. J. Tan, F. Tombari, and N. Navab. A combined generalized and subject-specific 3d head pose estimation. In *3DV*, 2015. 2

[39] A. Tawari, S. Martin, and M. M. Trivedi. Continuous head movement estimator for driver assistance: Issues, algorithms, and on-road evaluations. In *ITSC*, 2014. 2, 3

[40] T. Weise, S. Bouaziz, H. Li, and M. Pauly. Realtime performance-based facial animation. In *ACM Transactions on Graphics (TOG)*, 2011. 2