TICaM: A Time-of-flight In-car Cabin Monitoring Dataset

Jigyasa Singh Katrolia
jigyasa.katrolia@dfki.de

Ahmed Elsherif
ahmad.elsherif27@gmail.com

Hartmut Feld
hfeld@posteo.de

Bruno Mirbach
bruno.mirbach@dfki.de

Jason Rambach
jason.rambach@dfki.de

Didier Stricker
didier.stricker@dfki.de

DFKI
German Research Center for
Artificial Intelligence
Kaiserslautern, Germany

Abstract

We present TICaM, a Time-of-flight In-car Cabin Monitoring dataset for vehicle interior monitoring using a single wide-angle depth camera. Our dataset goes beyond currently available in-car cabin datasets in terms of the ambit of labeled classes, recorded scenarios and annotations provided; all at the same time. We recorded an exhaustive list of actions performed while driving and provide for them multi-modal labeled images (depth, RGB and IR), with complete annotations for 2D and 3D object detection, instance and semantic segmentation as well as activity annotations for RGB frames. In addition to real recordings, we provide a synthetic dataset of in-car cabin images with the same multi-modality of images and annotations, contributing a unique and extremely beneficial combination of synthetic and real data for effectively training cabin monitoring systems and also evaluating domain adaptation approaches. We provide baseline evaluation for object detection, segmentation and transfer learning tasks on our dataset. The dataset is available here.

1 Introduction

With the advent of autonomous and driver-less vehicles, it is imperative to monitor the entire in-car cabin scene in order to realize active and passive safety functions, as well as comfort functions and advanced human-vehicle interfaces. Such car cabin monitoring systems typically involve a camera fitted in the overhead module of a car and a suite of algorithms to monitor the environment within a vehicle [7, 9]. Ever growing performance of deep learning based computer vision methods has made it possible to monitor dynamic scenarios inside a car with high accuracy. To aid these monitoring systems, several in-car datasets exist to

Dataset	TICaM (Ours)	SVIRO[13]	AUC[10]	Brain4Cars[20]	HEH[25]	Drive&Act[23]
Year	2021	2020	2018	2016	2014	2019
#Frames	>118K	25K	17K	2M	11K	**>9.6M**
#Subjects	13	N/A	31	10	8	**15**
#Views	1	1	1	2	2	**6**
Synthetic/Real	**Both**	Synthetic	Real	Real	Real	Real
Data	**Depth/RGB/IR**	**Depth/RGB/IR**	RGB	RGB	RGB/Depth	**Depth/RGB/IR**
Annotation	**2D+3D box, 3D segmentation mask, activity, 2D keypoints**	**classification labels, 2D box and mask 2D keypoints**	activity	activity	activity	activity, 3D skeleton
#Activity classes	19	N/A	10	5	3	**83**
Scenarios	**Driver Driver+Passenger Driver+Object Driver+Child seat**	**Driver Driver+Passenger Driver+Object Driver+Child seat**	Driver	Driver	Driver	Driver

Table 1: Overview of existing in-car cabin monitoring datasets.

train deep learning methods for solving problems like driver distraction monitoring, occupant detection or activity recognition [10, 20, 23, 25]. In the same vein, we present TICaM, an in-car cabin dataset of 6.7K real time-of-flight depth images and 3.3k synthetic images with ground truth annotations for 2D and 3D object detection, and semantic and instance segmentation. In addition, we provide RGB video stream of driver and passenger activity along with activity annotations, totalling 118K RGB frames. Our intention is to provide a comprehensive in-car cabin depth image dataset that addresses the deficiencies of currently available such datasets in terms of the ambit of labeled classes, recorded scenarios and provided annotations; all at the same time.

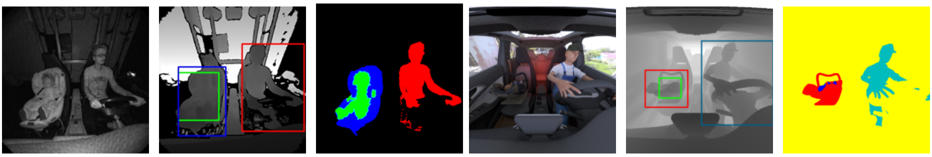

Figure 1: **From left to right** Real IR image, real depth image with 2D bounding box annotation, real image segmentation mask, synthetic RGB image, synthetic depth image with 2D bounding box annotation, synthetic image segmentation mask.

An immediately apparent limitation of contemporary car cabin datasets is the lack of representation of some commonly occurring driving scenarios. For example, scenarios involving passengers, everyday objects, and children and infants in forward facing and rearward facing child seats respectively are missing in these datasets. We take care to record a wide range of driving scenarios that occur in everyday life so that our dataset can be used for critical automotive safety applications like airbag adjustment. Using the same airbag setting for a child as for an adult can be fatal for the child. Therefore, it is important to know the occupant class (person, child, infant, object or empty) of each car seat and to determine the configuration of child seat (forward facing FF or rearward facing RF) [18]. Our dataset also contains annotations of both driver and passenger activities. Activity recognition is not only crucial for innovative contact-less human machine interfaces but can moreover be fused with other modalities like driver gaze monitoring to obtain a robust estimation of the driver state, activity, awareness and distraction, which is crucial for hand-over maneuvers in conditional or highly automated driving [12, 17].

Another key feature missing from popular in-car cabin datasets is multi-modality [10,

[20]. We provide depth, RGB and infrared images with focus on 3D data annotations. The images have been captured inside a driving simulator using a Kinect Azure [1] fixed in the front near the rear-view mirror, providing a more practical viewpoint compared to other datasets [10] and a mounting position that can be realistically replicated inside cars and which allows to monitor the entire front space of the vehicle cabin including driver and passenger seat.

We use time-of-flight depth modality because it is associated with some unique benefits. Depth images preserve privacy as subjects cannot be identified, they are more robust to illumination and color variations, and allow natural background removal. Additionally, it is easier to generate realistic synthetic depth data for training machine learning systems than it is to generate synthetic RGB data, which is also something we provide as part of our dataset. For depth and infrared images we provide 2D and 3D bounding boxes, and class and instance masks. These masks and 2D boxes can be used as they are for training on infrared images and can also be mapped to RGB images since the relative pose between depth and RGB cameras is known and provided. Lastly, we also provide annotations for activity recognition task making our dataset truly comprehensive in terms of input modalities and ground truth annotations. Such multi-modal data and annotations can be of interest for holistic driver monitoring methods like [14] for autonomous driving handover cases and monitoring driver attention.

Synthetic datasets are extremely useful for training systems that generalize well in practice when sufficient real training data is not available. Recognizing this fact has led us to also create a synthetic front in-car cabin dataset of over 3.3K depth, RGB and infrared images with annotations for detection, segmentation and keypoint estimation tasks. We believe this addition makes our dataset uniquely useful for training and testing domain adaptation methods.

To summarize, the main contributions of this paper are:

- **TICaM**, a comprehensive ToF dataset for in-car cabin monitoring, consisting of a total of 6668 images annotated with 2D and 3D bounding boxes, segmentation masks, and over 118K RGB images with activity labels.

- Additional to real images, **TICaM** contains 3306 synthetic images (Depth, RGB and IR imitation) with annotations for 2D and 3D object detection, instance segmentation as well as 2D keypoints, allowing for evaluation of domain adaptation methods.

- **TICaM** provides a new public benchmark opportunity for in-car cabin monitoring with high quality annotations, larger amounts of data and novel, more practical scenarios compared to existing datasets.

- The data in **TICaM** are recorded with a wide-angle RGB-D camera monitoring the entire front space of the cabin, capturing both driver and passenger seats.

2 Related Work

Surveying existing car and driving datasets indicates that the most common application of such datasets is driver monitoring. Many datasets exist for tasks like driver distraction recognition [10], driver behavior recognition [23], driver gaze detection [28, 29], driver activity recognition [25] and driver intention prediction [20]. All these datasets provide color images of driver from front view for predicting driver activity or intention, with the exception of HEH [25] and Drive&Act [23] which provide depth images as well. Brain4Cars [20] dataset is used to classify driver intention into 5 categories for maneuver prediction. HEH [25] captures additionally images of drivers' hands for predicting driver activity. The AUC Distracted Driver dataset [10] offers color images of drivers from side-view and can be used for detecting when the driver is distracted. All these datasets are limited in the number of image modalities they provide, with only one dataset providing depth images and none providing IR images. Moreover they are limited in the number of activity classes and image viewpoints they provide. Drive&Act [23] contributed a large-scale driver activity dataset with 6 views and multi-modal image data (RGB, Depth and IR) with 83 activity classes. However, their dataset provides only activity and 3D skeleton annotations and is limited in terms of recorded scenarios. In contrast, TICaM provides multi-modal images and annotations for a wider range of driving scenarios with a single wide-angle front view that effectively captures the entire car cabin.

As mentioned in section 1, occupant classification is a safety-critical task for safe deployment of airbags. Many past works have addressed this problem but no dataset has been made publicly available [15, 19, 22, 26]. Nowruzi *et al.* [24] released a dataset of thermal images for occupant classification, however, their dataset lacks images captured with child seats and children/infants in the scene. SVIRO [13] released a dataset of exclusively synthetic RGB, IR and depth images of the rear bench interior of ten different vehicles for occupant classification into four classes: person (also including child and infant), child seat, infant seat and object. We borrow methods and materials from SVIRO to build the synthetic imageset of our dataset. To the best of our knowledge, no publicly released real dataset for occupant classification provides scenes with passengers, children and child seats. We refer readers to Table 1 for a summary of the differences between TICaM and other popular in-car cabin datasets.

3 TICaM

We describe here in detail the dataset recording and annotation process for TICaM. The data capturing setup is described in section 3.1, the acquisition and rendering process of real and synthetic data is elaborated on in sections 3.2 and 3.3 respectively, the data format and the ground truth annotation format are described in sections 3.4 and 3.5 respectively, and a summary of training and testing splits is provided in section 3.6.

3.1 Data Capturing Setup

For data recording, we use an in-car cabin test platform developed by Feld et al.[16], shown in Figure 2. It consists of a realistic in-car cabin mock-up located in our indoor lab, equipped with a wide-angle projection system for a realistic driving experience. A Kinect Azure camera with a wide field of view is mounted at the rear-view mirror position for 2D and 3D data recording. The camera is set to record at 30fps with 2×2 binning. The captured data consists of RGB, depth and IR amplitude images. To ensure a wide range of variability in the dataset, we adjust the seat positioning of the driver and passenger seats in the driving simulator via a CAN bus system for every recorded sequence.

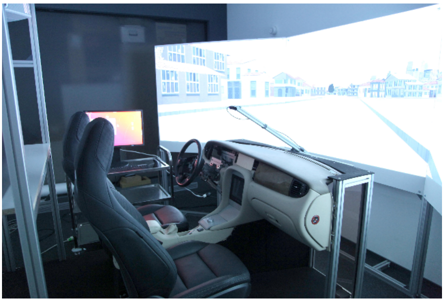

Figure 2: Our data capturing setup equipped with a wide-angle projection system, car front seats and a Kinect Azure camera in the front.

3.2 Data Acquisition

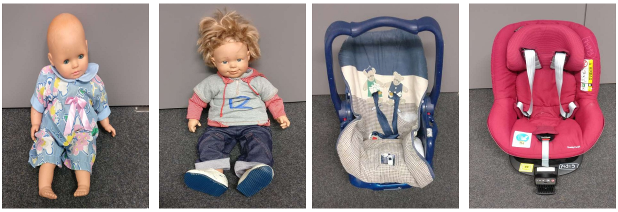

Figure 3: Example human dolls and child seats used for recording scenarios with children and infants on the passenger seat.

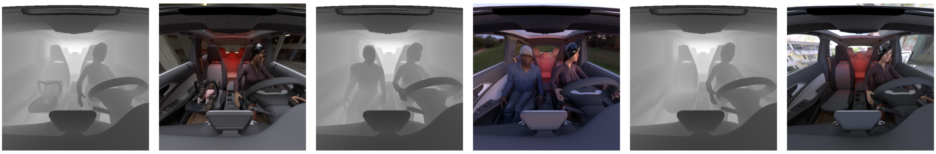

Figure 4: Different scenarios in the synthetic dataset.

We record following scenarios with 13 participants, 4 female and 9 male: 1) only driver in scene, 2) driver and passenger in scene, 3) driver and an object in scene, 4) driver and an empty Forward Facing Child Seat (FF) in scene, 5) driver and an empty Rearward Facing Infant Seat (RF) in scene, 6) driver and an occupied Forward Facing Child Seat (FF) in scene, 7) driver and an occupied Rearward Facing Infant Seat (RF) in scene, and 8) only an object in scene (please see supplementary material for illustration). We define a choreography each for the driver and the passenger which we share with the participants before recording of a driving sequence. For example, drivers performed actions like sitting, driving normally, looking left while turning wheel, turning right and so on. On the other hand, passengers performed actions like talking to the driver, grabbing something from the dashboard, etc. In total we have 19 actions for both driver and passenger (see Table 3). For each participant or pair of participants (in case of both driver and passenger), we record several sequences with varying positions of car seats. We also vary the appearance of the people through different clothing accessories like jackets and hats. Example images from the dataset are provided in the supplementary material.

For practical reasons we use human dolls as substitute for children and infants to record scenarios where one person is driving with an FF or RF seat securely placed on the passenger seat. Along with the dolls, we use 3 FF and 3 RF seats in different orientations like sun shade up/down or handle up/down. The human dolls and example RF and FF seats are shown in Figure 3.

3.3 Synthetic Data Generation

We refer to the materials and methods described in SVIRO [13] and vary the body poses of human models to adapt them for realistic driving scenarios to create our synthetic in-car cabin dataset. The difference between SVIRO and TICaM being that SVIRO provides car rear-cabin data while we provide front-cabin data. We render synthetic images using Blender 2.81 [2]. The 3D model of Mercedes A Class vehicle is from Hum3D [4], the everyday objects were downloaded from Sketchfab [6] and the human models were generated via MakeHuman [5]. In addition, High Dynamic Range Images (HDRI) [3] was used to get different environmental backgrounds and illumination, and finally, in order to define the reflection properties and colors for the 3D objects, textures from Textures.com [8] were obtained for each object.

Since our goal was to generate simulated driving scenarios, we had the class occupying the driving seat to always be an adult driver in a driving pose while for the passenger seat, we have randomly selected the occupant between the rest of the classes, that is child seat (empty and child-occupied), everyday objects, and adult passengers. We selected some of the same objects as the everyday objects in real driving scenarios like handbag, backpack, bottle, etc. but we group them under the same class ’object’. We recreated the actions the participants were asked to perform during the recording of the real dataset. Since the driver poses are always restricted by the car elements they are interacting with, and in order to avoid intersection, we created some fixed poses for the hand positions that are possible in real driving scenarios, while allowing movement of other body parts up to some threshold. On the other hand, there were no such restrictions for the passenger poses, and thus we replicated the poses used in SVIRO by randomly selecting body poses within the physical constraints of the car seats. Figure 4 shows a few images from the synthetic dataset showcasing the different scenarios present in this imageset.

3.4 Data Format

Both the real and synthetic data are delivered in the same format for image data and annotations with a few differences explained below. The camera information is as follows:

- **Depth Z-image**. The depth image is undistorted with a pixel resolution of 512×512 pixels and captures a $105^\circ \times 105^\circ$ FOV. The depth values are normalized to $[1mm]$ resolution and clipped to a range of $[0, 2550mm]$. Invalid depth values are coded as ‘0’. Images are stored in 16bit PNG-format.

- **IR amplitude Image**. Undistorted IR images from the depth sensor are provided in the same format as the depth image above.

- **RGB image**. Undistorted color images are saved in PNG-format in 24bit depth. While the synthetic RGB images have the same resolution and field of view as the corresponding depth images (512 × 512), the real recorded RGB images have a higher resolution of 1280 × 720 pixels, but a lower field of view of 90° × 59° FOV.

- **RGB video**. The color video is provided as is recorded by the camera for sequences where a person is present in the scene.
- **Camera intrinsics**. Along with the images, the camera intrinsics are also provided for both RGB and depth sensors in Kinect Azure as well as their relative rotation and translation. The camera intrinsics of the virtual camera in Blender are also provided.
- **Camera pose**. The different camera poses used for real recorded sequences are also provided as .yml files.

3.5 Data Annotation

The data annotation is done using the SALT 3D-annotation tool [30]. We annotate every twentieth frame of the recorded sequences. For synthetic data, the ground truth is generated using scripts provided by authors of SVIRO [13]. We provide the following annotations:

- **2D bounding boxes**. For each real depth image, the 2D boxes are defined by the top-left and bottom-right corners of the box, its class label and a flag ’low remission’ which is set to 1 for objects which are either black or very reflective or both, and therefore are barely visible in the depth image. Each 2D box in the synthetic dataset is represented by its class ID, the top-left and the bottom-right corners.
- **3D bounding boxes**. Each 3D box in the real dataset is represented by the coordinates (cx, cy, cz) of the box center, its dimensions (width, height, depth), its orientation along x-, y- and z- axes with respect to the world coordinate system, its class label and the ’low remission’ flag. Whereas, each 3D box for the synthetic images is represented by the coordinates (cx, cy, cz) of the box center in camera coordinate system, its dimensions (width, height, depth) and its class ID.
- **Pixel segmentation masks**. For each real depth image two corresponding masks are generated: instance mask and class mask. The pixel intensities in these masks correspond to the class ID in the class mask and the instance ID for a certain class in the instance mask. For synthetic images we provide a single mask in the format used by SVIRO [13].
- **Activity annotations**. For all sequences with people in the scene, we provide a .csv file describing the activities performed throughout the sequence. Each .csv contains the activity ID, activity name, person ID, a label either ’Driver’ or ’Passenger’ to specify if the action is performed by the driver or the passenger, the starting frame number of the action, the ending frame and the duration of that action in number of frames.
- **Skeleton annotations**. We provide 2D keypoints for all persons including children and infants in the synthetic images in the same format as in SVIRO [13].

Table 2: Number of frames in real and synthetic imagesets for training and testing for different tasks.

	Activity recognition	**2D detection** **3D detection** **Segmentation**	
	real	***real***	***synthetic***
train	86K	4.7K	3.3K
test	32K	2.0K	-

3.6 Data Statistics

TICaM is a combined dataset of 6.7K real time-of-flight depth and IR images, video recordings equivalent to over 118K real RGB frames, and 3.3K synthetic depth, IR and RGB images. We use synthetic dataset only for training purpose in order to evaluate domain adaptation benefits when evaluating on the real dataset. We split the real dataset into training and testing sets such that a participant, RF, FF or an object instance belongs to either training or testing set. Table 2 summarizes the number of images present in each imageset. An overview of the object and the activity classes annotated in the dataset for detection, segmentation and activity recognition tasks is provided Table 3.

Table 3: Overview of the object and activity classes present in TICaM.

Data type	**Annotation type**	**Class names**	**No. of Classes**
Real	Object classes	Person, Backpack, WinterJacket, Box, WaterBottle, MobilePhone, Blanket, Accessory, Book, Laptop, LaptopBag, Infant, Handbag, FF, RF, Child	16
	Activity Classes	drive, look left while turning wheel , look right while turning wheel, touch screen, open glove compartment, touch head or face, lean forward, turn left, turn right, turn backwards while reversing, adjust sun visor, turn backwards, talk, take something from dashboard, sitting normally, read paper or book, bending down, using phone, using laptop	19
Synthetic	Object Classes	Person, FF, RF, Object, Child, Infant	6

4 Baseline Evaluation

In this section we provide baselines for different tasks in TICaM using CNN-based methods. Detailed results, training hyperparameter settings and architecture details for all experiments are provided in the supplementary material.

4.1 Semantic Segmentation

We choose YOLACT [□] for object detection and segmentation due to its light-weight architecture that is suitable for real-time car cabin monitoring applications. We train YOLACT

Table 4: Detection and segmentation performance (in terms of Average Precision AP at IoU=0.5) of YOLACT for each class, when trained on *train_real* imageset for all 16 classes and evaluated on *test_real* imageset.

Class	Person	Backpack	Jacket	Box	Bottle	Phone	Blanket	Accessory	Book	Laptop	Laptop bag	Handbag	Infant	FF	RF	Child
Box AP	98.90	95.21	92.49	20.92	54.45	15.04	2.09	3.82	26.72	6.92	55.46	14.72	34.44	99.75	91.61	100.0
Mask AP	99.0	95.21	97.49	31.37	54.45	0.0	0.0	3.82	26.72	5.54	54.37	16.07	34.44	67.26	91.61	100.0

on *train_real* imageset on all 16 classes labelled in TICaM. As noted in Table 4, YOLACT has great difficulty in detecting highly reflective objects like 'Laptop' and 'MobilePhone'. Beside these, class 'Blanket' and 'Accessory' also perform poorly.

We then combine all object classes (for example laptop, box, bottle, etc.) into a single 'object' class and train YOLACT again on *train_real* and *train_syn* imagesets for resulting 6 object classes. We do this because in most occupant classification systems only a broad classification of objects is required. For all further experiments, only these 6 object classes are used. Table 5 (first three rows) shows the mean Average Precision mAP of the predicted masks and boxes on *test_real* dataset for each of the three training strategies.

Table 5: Detection and segmentation performance (in terms of mean Average Precision mAP at IoU=0.5) of YOLACT with different training strategies.

Method	**Training Split**	**Box mAP**	**Mask mAP**
YOLACT [11]	*train_real* with 16 classes	50.79	48.59
	train_real	**91.11**	85.78
	train_syn	27.67	18.99
	train_syn; finetuned on *train_real*	87.12	**86.25**
Faster R-CNN [27]	*train_real*	**87.1**	na
	train_syn	16.4	na
	train_syn; finetuned on *train_real*	84.8	na

4.2 Object Detection

We train and evaluate a Faster R-CNN [27] network on our dataset for object detection. We train on both *train_real* and *train_syn* imagesets before evaluating on *test_real*. Detection mean Average Precision is presented in Table 5 (rows five and six) and qualitative results are presented in the supplementary. We can note that Faster R-CNN performs slightly worse than YOLACT for both experiments.

4.3 Transfer Learning from Synthetic to Real Data

The results in table 5 (rows three and five) show that training on synthetic data without any adaptation results in rather poor results on real test data, therefore some form of domain adaptation is clearly required and methods therefore can be evaluated with our dataset. As one possible approach we train both YOLACT and Faster R-CNN on *train_syn* imageset and then further finetune later layers of these networks on *train_real* imageset. Thus, we are able to also analyze the benefit of transfer learning from synthetic to real data. Table 5 (rows four and seven) presents the mean Average Precision achieved by YOLACT and Faster R-CNN in these experiments. We can see that the performance obtained in this manner is slightly below than when trained on only real images for both the networks. We can attribute this loss of performance to mainly two classes, namely 'object' and 'FF' (please see supplementary material for more discussion). We have also observed in our experiments, that using a smaller subset of the synthetic imageset for pretraining leads to slightly higher performance than when the entire imageset is used. We believe this is due to the pretrained classifier overfitting to the synthetic imageset. Nevertheless, we believe TICaM offers a possibility to test more sophisticated domain adaptation approaches on an interesting and challenging domain.

We can also observe in table 5 that the box mAP of YOLACT is consistently better than the mask mAP for every training strategy. This is in accordance with the results reported in the YOLACT paper [11] and we refer readers to the same for a detailed discussion on the mAP gap between boxes and masks. According to the authors their novel Fast NMS version of the usual NMS method removes slightly too many boxes, and since YOLACT uses predicted boxes to crop the masks, this affects the segmentation performance.

We note that although SVIRO in itself presents a synthetic source domain dataset that could have been used to test the transfer learning approach, we do not demonstrate it here because SVIRO is a dataset of car rear bench that contains images from a different viewpoint. We refer readers to our previous work [21] where we showed usage of unsupervised image translation using CycleGAN for domain adaptation from SVIRO to real in-car cabin depth images.

5 Conclusion

We present TICaM, a unique time-of-flight in-car cabin monitoring dataset consisting multi-modal images and multi-purpose annotations. We capture the entire front car cabin space with a single camera and record a comprehensive set of real life driving scenarios that are missing from contemporary driving datasets. We provide both real and synthetic imagesets, and annotate them for 2D and 3D detection, instance segmentation and activity recognition tasks. Our dataset can be used for training cabin monitoring systems to provide both safety-critical functionalities like safe deployment of airbags, driver behaviour monitoring as well as comfort functions. Furthermore, the similarity of the real and the synthetic image sets renders TICaM suitable for the testing of domain adaptation approaches. We show through baseline evaluation that TICaM is a challenging dataset to test in-car cabin monitoring systems and domain adaptation methods.

Acknowledgement. This work was partially funded within the Electronic Components and Systems for European Leadership (ECSEL) Joint Undertaking in collaboration with the European Union's H2020 Framework Program and Federal Ministry of Education and Research of the Federal Republic of Germany (BMBF), under grant agreement 16ESE0424 / GA826600 (VIZTA).

The authors would also like to thank Steve Dias Da Cruz from IEE S.A. for providing the source code used in the creation of the synthetic imageset and IEE S.A. for providing the child seats and dolls used while recording.

References

[1] Azure kinect. https://azure.microsoft.com/en-us/services/kinect-dk/.

[2] Blender. https://www.blender.org.

[3] Hdri haven. http://www.hdrihaven.com.

[4] Hum3d haven. http://www.hum3d.com.

[5] Makehuman. http://www.makehumancommunity.org.

[6] Sketchfab. http://www.sketchfab.com.

[7] Driver monitoring system interior sensing for vehicleintegration. https://smarteye.se/automotive-solutions/.

[8] Textures.com. http://www.textures.com.

[9] Valeo driver monitoring. https://www.valeo.com/en/driver-monitoring/.

[10] Yehya Abouelnaga, Hesham M. Eraqi, and Mohamed N. Moustafa. Real-time distracted driver posture classification, 2018.

[11] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9156–9165, 2019. doi: 10.1109/ICCV.2019.00925.

[12] C. Braunagel, W. Rosenstiel, and E. Kasneci. Ready for take-over? a new driver assistance system for an automated classification of driver take-over readiness. *IEEE Intelligent Transportation Systems Magazine*, 9(4):10–22, 2017. doi: 10.1109/MITS.2017.2743165.

[13] Steve Dias Da Cruz, Oliver Wasenmuller, Hans-Peter Beise, Thomas Stifter, and Didier Stricker. Sviro: Synthetic vehicle interior rear seat occupancy dataset and benchmark. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 973–982, 2020.

[14] Isha Dua, Akshay Uttama Nambi, C.V. Jawahar, and Venkat Padmanabhan. Autorate: How attentive is the driver? In *2019 14th IEEE International Conference on Automatic Face Gesture Recognition (FG 2019)*, pages 1–8, 2019. doi: 10.1109/FG.2019.8756620.

[15] M. E. Farmer and A. K. Jain. Occupant classification system for automotive airbag suppression. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 1, pages I–I, 2003. doi: 10.1109/CVPR.2003.1211429.

[16] Hartmut Feld, Bruno Mirbach, Jigyasa Singh Katrolia, Mohamed Selim, Oliver Wasenmüller, and Didier Stricker. Dfki cabin simulator: A test platform for visual in-cabin monitoring functions. In *Proceedings of the 6th Commercial Vehicle Technology Symposium - CVT*, 2020.

[17] Lex Fridman, Philipp Langhans, Joonbum Lee, and Bryan Reimer. Driver gaze estimation without using eye movement. 07 2015.

[18] John D Graham, Sue J Goldie, Maria Segui-Gomez, Kimberly M Thompson, Toben Nelson, Roberta Glass, Ashley Simpson, and Leo G Woerner. Reducing risks to children in vehicles with passenger airbags. *Pediatrics*, 102(1):e3–e3, 1998.

[19] Shih-Shinh Huang and Pei-Yung Hsiao. Occupant classification for smart airbag using bayesian filtering. pages 660 – 665, 07 2010. doi: 10.1109/ICGCS.2010.5542979.

[20] Ashesh Jain, Hema S. Koppula, Bharad Raghavan, Shane Soh, and Ashutosh Saxena. Car that knows before you do: Anticipating maneuvers via learning temporal driving models. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[21] Jigyasa Katrolia, Lars Krämer, Jason Rambach, B. Mirbach, and Didier Stricker. An adversarial training based framework for depth domain adaptation. In *VISAPP 2021 - 16th International Conference on Computer Vision Theory and Applications*, 2021.

[22] Jooyoung Lee, Jihye Byun, Jaedeok Lim, and Jaeyun Lee. A framework for detecting vehicle occupancy based on the occupant labeling method. pages 660 – 665, 12 2020. doi: 10.1155/2020/8870211.

[23] Manuel Martin, Alina Roitberg, Monica Haurilet, Matthias Horne, Simon Reiß, Michael Voit, and Rainer Stiefelhagen. Drive&act: A multi-modal dataset for fine-grained driver behavior recognition in autonomous vehicles. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2019.

[24] Farzan Erlik Nowruzi, Wassim El Ahmar, Robert Laganiere, and Amir Ghods. In-vehicle occupancy detection with convolutional networks on thermal images. 06 2019. doi: 10.1109/CVPRW.2019.00124.

[25] E. Ohn-Bar, S. Martin, A. Tawari, and M. M. Trivedi. Head, eye, and hand patterns for driver activity recognition. In *2014 22nd International Conference on Pattern Recognition*, pages 660–665, 2014. doi: 10.1109/ICPR.2014.124.

[26] Toby Perrett, Majid Mirmehdi, and Eduardo Dias. Cost-based feature transfer for vehicle occupant classification. 12 2015.

[27] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper/2015/file/14bfa6bb14875e45bba028a21ed38046-Paper.pdf.

[28] R. F. Ribeiro and P. Costa. Driver gaze zone dataset with depth data. *2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019)*, pages 1–5, 2019.

[29] Mohamed Selim, Ahmet Firintepe, Alain Pagani, and Didier Stricker. Autopose: Large-scale automotive driver head pose and gaze dataset with deep head orientation baseline. In *VISIGRAPP (4: VISAPP)*, pages 599–606, 2020.

[30] Dennis Stumpf, Stephan Krauß, Gerd Reis, Oliver Wasenmüller, and Didier Stricker. Salt: A semi-automatic labeling tool for rgb-d video sequences. In *VISAPP 2021 - 16th International Conference on Computer Vision Theory and Applications*, 2021.