SVIRO: Synthetic Vehicle Interior Rear Seat Occupancy Dataset and Benchmark

Steve Dias Da Cruz [1,2,3]
steve.dias-da-cruz@iee.lu

Oliver Wasenmüller [2]
oliver.wasenmueller@dfki.de

Hans-Peter Beise [4]
h.beise@inf.hochschule-trier.de

Thomas Stifter [1]
thomas.stifter@iee.lu

Didier Stricker [2,3]
didier.stricker@dfki.de

[1] IEE S.A. [2] DFKI - German Research Center for Artificial Intelligence
[3] University of Kaiserslautern [4] Trier University of Applied Sciences

Abstract

We release SVIRO, a synthetic dataset for sceneries in the passenger compartment of ten different vehicles, in order to analyze machine learning-based approaches for their generalization capacities and reliability when trained on a limited number of variations (e.g. identical backgrounds and textures, few instances per class). This is in contrast to the intrinsically high variability of common benchmark datasets, which focus on improving the state-of-the-art of general tasks. Our dataset contains bounding boxes for object detection, instance segmentation masks, keypoints for pose estimation and depth images for each synthetic scenery as well as images for each individual seat for classification. The advantage of our use-case is twofold: The proximity to a realistic application to benchmark new approaches under novel circumstances while reducing the complexity to a more tractable environment, such that applications and theoretical questions can be tested on a more challenging dataset as toy problems. The data and evaluation server are available under https://sviro.kl.dfki.de.

1. Introduction

Interior vehicle sensing has gained increased attention in the research community, in particular due to challenges and developments related to automated vehicles [1, 2]. In this work, we focus on rear seat occupant detection and classification using a camera system and different ground truth data, as illustrated in Figure 1. Information about the presence and location of the passengers can be used to help reduce injuries in case of an accident, e.g. by adjusting the strength of airbag deployment [3, 4]. Seat occupancy detection can be used to remind the passengers to fasten their seat-belts or to detect children forgotten in the car [5, 6].

Figure 1. Example scenery of SVIRO together with the provided ground truth data. Left seat: infant seat with an infant. Middle seat: empty. Right seat: adult passenger. a) RGB image with keypoints for human pose estimation. b) Grayscale infrared imitation. c) Position and class based instance segmentation. d) Depth map.

For autonomous driving, it will be of interest to understand the overall scenery in the car interior [7], e.g. for handover situations [8]. For all the aforementioned applications, one has to ensure that trained machine learning models will be capable of classifying new types of child seats correctly while not being mislead by arbitrary everyday objects or through the window background sceneries. However, machine learning-based models, and specifically neural networks, trained in a single environment take non-relevant characteristics of the specific environmental conditions into account in an uncontrolled way [9] and therefore data must be recorded repetitively for different environments. Acquiring images in various (natural) lightning and weather conditions and accounting for different seat textures, car interior features, or even changing camera poses make the data

acquisition even more difficult. While domain adaptation investigates solutions to account for a shift in the source distribution with respect to the target distribution, common approaches still need a large amount of data for the target distribution [10, 11] to work well. Consequently, the means for generating a real training dataset with the corresponding annotations are limited and need to be repeated for each additional new car model and automotive manufacturer. Therefore, theoretically founded means to overcome the limitations of datasets collected for many real world applications have to be developed or advanced.

Common machine learning datasets and benchmarks focus on pushing the state-of-the-art of general tasks like classification [12], segmentation [13], object detection [14], human pose estimation [15] or multiple tasks at once [16, 17, 18, 19]. They do so on sceneries of high variable backgrounds and intra-class variations, or focus on toy examples to investigate theoretical and fundamental research questions [20]. However, none of the available datasets focuses on the application-oriented case when all images are taken on the same, or similar, background. They do not consider classes with only sparse representations, as is common in engineering problems when the available resources are limited. Consequently, available datasets do not provide a framework to evaluate models trained in the above-mentioned challenging conditions for solving identical tasks, but in a new environment. Hence, similar investigations for the rear seat occupancy cannot be performed and there is no publicly available dataset for the vehicle interior.

We release SVIRO to provide a starting point for investigating the aforementioned challenges and overcome some of the shortcomings of common available datasets. For the training set, we used different human models, child and infant seats, backgrounds and textures than for testing. Hence, we can test the generalization and robustness of models trained in one vehicle to a new one, for solving the same task. Our dataset has a higher visual complexity than toy scenarios while being close enough to a realistic application. Consequently, SVIRO can be used to benchmark common machine learning tasks under new circumstances while allowing the investigation of theoretical questions due to its intrinsically more tractable environment. Additional ground truth data for existing sceneries can be generated or new features can be integrated upon request. For an overview, you can also watch our video https://youtu.be/_arwrYIz7Ok

2. Related work

Some previous works have been investigating occupant classification [3, 4], seat-belt detection [21] or skeletal tracking [7] in the passenger compartment, but, as to best of our knowledge, no dataset was made publicly available.

Investigations regarding the tasks and challenges as mentioned in Section 1 could also be performed in a different framework, as long as they reproduce the same limitations. KITTI [17] provides a wide range of different available annotations and benchmarks for vehicle exterior applications. Closely related are the Cityscapes dataset [13] for different segmentation tasks, ECP [14] for person detection in urban traffic scenes and JTA [15] for pedestrian pose estimation and tracking. On the other hand, there is COCO [19], a widely used benchmark for object detection, keypoint detection and panoptic and stuff segmentation as well as PASCAL VOC [16]. Similarly, with Open Images [18], the largest unified dataset for image classification, object detection and instance segmentation was released. Even though these datasets contribute a wide range of images and corresponding annotations, they all have in common that their provided data has intrinsically high background and intra-class variation due to their nature for the exterior application. These datasets can be used to benchmark models for their performance and push the state-of-the-art in specific tasks, as ImageNet [12] did for classification. However, it is not possible to test the generalization to new environments and unseen intra-class variations for a larger range of tasks when only a limited amount of variability is available during training. In particular, those datasets cannot be used to benchmark applications for the (vehicle) interior regarding the challenges discussed in Section 1.

The annual VISDA challenge [22] hosts a benchmark for domain adaptation for different tasks, but it is limited to the transfer from synthetic to real data and solutions to different tasks are not comparable. It includes the Syn2Real [23] dataset for classification and object detection and the transfer from GTA sceneries [24] to Cityscapes [13] for segmentation. Other common datasets for domain adaptation, e.g. Office-Home [25], DomainNet [26] and Open MIC [27], focus on a single task and/or the transfer from non-real to real environments. Some approaches combine two existing datasets to test the generalization from synthetic to real images, e.g. from synthetic traffic signs [28] to real ones [29].

It is believed that scene decomposition into meaningful components can improve the transfer performance on a wide range of tasks [20]. Although datasets like CLEVR [30] and Objects Room [20] exist, they are limited to toy examples and lack increased visual complexity.

Moreover, deep learning-based approaches capture too much relevance between the information contained in the background and the task the models are designed to solve [9]. Consequently, the aforementioned datasets all help to push the state-of-the-art for many computer vision tasks, but lack the possibility to investigate the challenges introduced in Section 1. With our SVIRO dataset and benchmark we are the first to provide the means to analyze the generalization and reliability of machine learning-based approaches for different tasks when only a limited number of variations is available during training. We thereby address an impor-

tant engineering issue. Further, recent studies have shown the importance and applicability of using synthetic data for investigations in the automotive industry [31] possibly in combination with real data [32, 33].

3. Dataset

We created a synthetic dataset to investigate and benchmark machine learning approaches for the application in the passenger compartment regarding the challenges introduced in Section 1 and to overcome some of the shortcomings of common datasets as explained in Section 2.

3.1. Synthetic objects

We used the free and open source 3D computer graphics software Blender 2.79 [34] to construct and render the synthetic 3D sceneries. We used realistic child safety seats or child restraint systems (CRS) to which we will simply refer to as child seats. For our dataset, we selected a subset of available seats on the market, from which we then created a 3D model so that it could be used in our simulation. The 3D models were generated using depth cameras (Kinect v1) and precise structured light scanners (Artec Eva).

We needed to define the reflection properties and visual colors for each 3D object in the scene, so that its perception by the camera under simulated lightning conditions could be calculated. For this, we used textures (Albedo, Normal and Roughness images) from Textures.com [35] (with permission) for all the objects in the scene. The environmental background and lightning were created by means of High Dynamic Range Images (HDRI) from HDRI Haven [36]. The human models (adults, children and babies) and their clothing (additional clothes were downloaded from the community assets [37]), were randomly generated by using the open source 3D graphic software MakeHuman 1.2.0 [37]. The 3D models of the cars were purchased from Hum3D [38] and everyday objects (e.g. backpacks, boxes, pillows) were downloaded from Sketchfab [39].

3.2. Design choices

During the data generation process we tried to simulate the conditions of a realistic application. We decided to partition the available human models, child seats and backgrounds such that one part is only used for the training images (for all the vehicles) and the other part is used for the test images. For each of the ten different vehicle passenger compartments and available child seats, we fixed the texture as if real images had been taken. Consequently, the machine learning models need to generalize to previously unknown variations of humans, child seats and environments. In this setting, we can train models in one or several car environment(s) and test them on a different one. This is an advantage compared to common domain adaptation datasets [23, 25, 26, 28, 29] which usually focus on the transfer from synthetic to real images. Further, the photorealistic rendering and close-to-real models introduce a high visual complexity which makes them more challenging than toy examples [20, 30]. The dataset has an intrinsic dominant background and texture bias: all of the images are taken in a few passenger compartments, but generalization to new, unseen, passenger compartments and child seats should be achieved. This evaluation is currently not possible by state-of-the-art datasets [13, 14, 15, 16, 17, 18, 19].

The human models were generated randomly in MakeHuman. Their facial expression was selected to be neutral and identical. We defined a fixed set of poses for the humans represented by unit quaternions. For every human in each scenery, two poses were selected randomly and a spherical linear interpolation (Slerp) [40] was performed to get an intermediate pose. For each scenery, we randomly selected what kind of object is placed at each position, however, we avoided appearances of the same object for a same scenery. Child and infant seats can be empty and we decided to not allow children to be placed on the rear seat without a child seat. Infant seats were randomly rotated by 180° along the z-axis and an offset from the straight ahead orientation was randomly applied to all child seats. The handle of the infant seat was selected to be up or down. Randomly selected environmental backgrounds were rotated around the vehicle to simulate arbitrary lightning conditions. We placed everyday objects onto the rear seat to make the scenery more versatile. All cameras have the same intrinsic parameters (focal length=$3.4\,\mathrm{mm}$, sensor width: $8.5\,\mathrm{mm}$, f-number= 2.5, skew coefficient= 0, focal length in terms of pixels: $\alpha_x = 514.4208$, $\alpha_y = 514.4208$, principal point: $u_0 = 640$, $v_0 = 480$), however, their pose is different in each car. Example sceneries for training and test data can be found in Figure 2 and in the supplementary material. An overview of the 3D objects are shown in Figure 3.

We also generated a training dataset with randomly selected (partially unrealistic) textures and backgrounds from a large pool of images. When trained on the latter, the increased variations improve the generalization for classification and semantic segmentation on the test set and to new passenger compartments, as shown in Section 4.1 and 4.2. An additional advantage of our approach is the possibility to create images under defined conditions (e.g. same scenery, but under different lightning conditions) so that additional investigations can be performed in future works. Moreover, the difficulty can be gradually increased: one can, for example, train on occupied child and infant seats only, train on infant seats with the handle down (or up) only or removing everyday object completely from training.

3.3. Statistics

Our dataset consists of ten different vehicles: BMW X5, BMW i3, Hyundai Tucson, Tesla Model 3, Lexus GS F,

Figure 2. Example sceneries for training (top) and test (bottom) splits for different cars. Each split uses different objects, seats, environments and humans. Some images appear darker, which is why (also in real applications) it is preferred to use an active infrared camera system.

Figure 3. Representative selection of the assets used for our synthetic dataset. First and third row are assets used for the training while the second and fourth are assets used for testing. Some children and adults for training and 1 environment per split are not shown.

Mercedes A-Class, Renault Zoe, VW Tiguan, Toyota Hilux and Ford Escape. The number of windows varies, which causes different lightning conditions, and some cars have only two rear seats instead of three. The different vehicle interiors are compared in Figure 4. We used the same people and child seats for the training set of each vehicle and the remaining ones for the test sets. This results in two child seats and one infant seat per data split. We did the same for the background: five were selected for the training and five different ones for the test set. For the everyday objects, we used two bags, a card-box and a cup for the training dataset and a different bag, a paper-bag, pillows and a box of bottles for the test set. The number of people and the distribution of the gender, age and ethnicity for the training and test set can be found in Table 1. The number of images generated for each vehicle and each training and test set are identical. In total, this results in 20000 training and 5000 test sceneries. The distribution of the different classes across the vehicles and data splits is summarized in Figure 5. The number and constellation of appearances varies between the vehicles, because all the sceneries were generated randomly.

	Train			Test		
	Adult	Child	Baby	Adult	Child	Baby
African	5	2	1	2	1	1
Asian	5	2	1	2	2	1
Caucasian	4	2	1	4	1	1
Female	9	3	-	5	2	-
Male	5	3	-	3	2	-
Total	14	6	3	8	4	3
Per Car	2000			500		

Table 1. Number of people and distribution of gender, age and ethnicity for the training and test dataset. The same people were used for the training and test set for all the vehicles, respectively, and the same number of images were generated for each car.

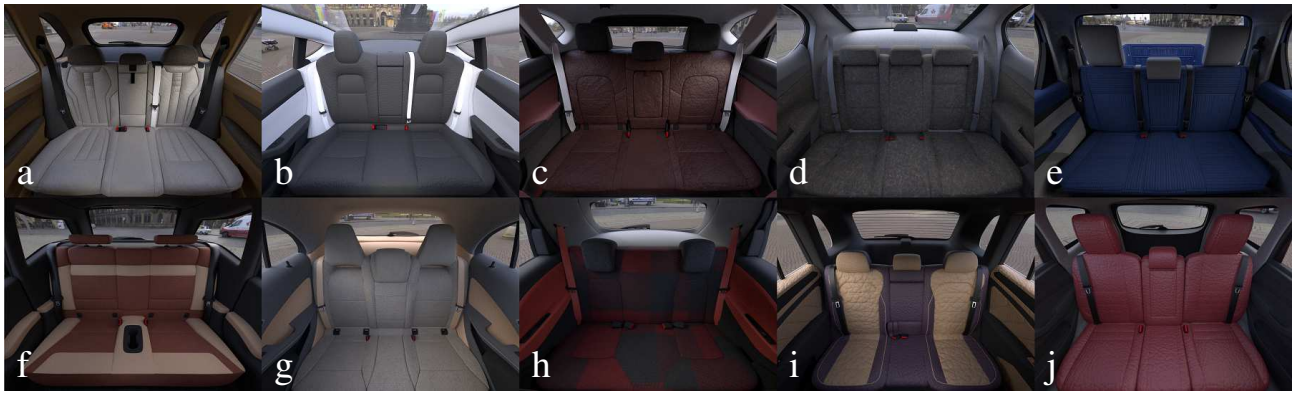

Figure 4. Comparison of the different vehicle interiors. a) BMW X5, b) Tesla Model 3, c) Hyundai Tucson, d) Lexus GS F, e) Toyota Hilux, f) BMW i3, g) Mercedes A-Class, h) Renault Zoe, i) VW Tiguan and j) Ford Escape. The geometry of the rear-seat, the windows, headrest and car features differ between the cars and some cars only have two seats instead of three.

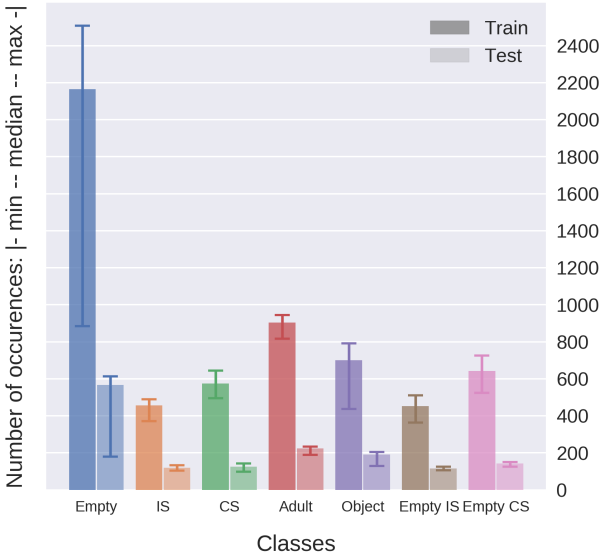

Figure 5. Distribution of the different classes over the vehicles and data splits. As the images were generated randomly, the distribution is different for each split and vehicle. The bar represents the median value for a given class for a given data split over all vehicles. The error bar represents the minimum and maximum number of occurrences along the vehicles for a given split. The dark colors represent the training data and the light ones the test data. We abbreviate infant seat as IS and child seat as CS. The large difference in empty seats is due to vehicles with only two rear seats.

3.4. Rendering

The synthetic images were generated using Blender, its Python API and the Cycles renderer. As many applications in the passenger compartment require an active infrared camera system to work in the dark, we decided to imitate such a system by means of a simple approach: We placed an active red lamp (R=100%, G=0%, B=0%) next to the camera inside of the car illuminating the rear seat, but overlapping with the illumination from the HDR background image. We then took the red channel only from the resulting rendered RGB image. We will refer to these images as grayscale images. This is, however, not a physically accurate simulation of a real active infrared camera system. The simulation of the latter is not trivial, as the perception in the infrared domain not only depends on the object's material properties, but also on the wavelength which is used [41]. We argue that this is of minor importance, because SVIRO is intended to investigate the general applicability of possible machine learning methods. Our approach helps to become less dependent on the environmental lightning and to facilitate the tasks: see Figure 6 for a comparison between a standard RGB image and our grayscale image for a dark scenery, where a lot of information would be lost. More comparisons are available in the supplementary material. Moreover, we report in Section 5 and Figure 10 the evaluation of a model trained on SVIRO on real infrared images and show that it behaves similarly on real data.

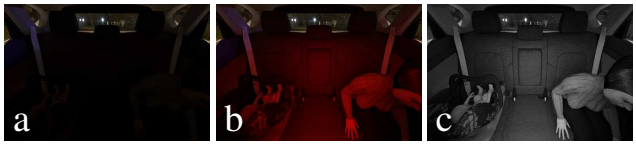

Figure 6. Comparison between a standard RGB image and our simple approach to imitate an active infrared camera system for a dark scenery. a) Standard RGB image in environmental lightning. b) RGB image of the scenery illuminated by an active red light. c) Red channel only of the RGB image of the illuminated scenery (used as infrared imitation in SVIRO).

3.5. Ground truth

For each scenery we provide a set of images and ground truth data: 1) An RGB image of the scenery without an

active red lamp next to the camera, e.g. Figure 2, 2) a grayscale image (red channel only) of the rendered RGB image using an active red lamp next to the camera, e.g. Figure 1 (b), 3) an instance segmentation map, where each object is color-coded depending on its position and class, e.g. Figure 1 (c), 4) Bounding boxes for all the elements in the scenery, 5) Keypoints for all the human poses in the scenery, e.g. Figure 1 (a), 6) a depth map of the scenery, e.g. Figure 1 (d). For classification, we split the images (RGB, grayscale, depth) into three rectangles (one for each seat position) with slight overlap between them. See Figure 7 for an illustration. If a car has only two seats, then we exclude the middle rectangle. Note that objects from neighbouring seats are overlapping to the neighbouring rectangle, which makes classification more difficult. However, this is necessary as people can lean over to the neighbouring seat. Both semantic segmentation and instance segmentation can be performed using the provided segmentation masks. Children on a child seat, as well as babies in an infant seat, are treated as two separate instances. We save the human poses by using keypoints, as used by the COCO dataset [19], but our skeleton is defined using partially different joints. The visibility of the keypoints are set to zero if the keypoint is outside the image, to one if it is occluded by an object or neighbouring human and set to two if it is visible or occluded by the person itself. Keypoints are provided for the babies as well. For each scenery, we provide a .json file containing the 2D pixel coordinates of the keypoints of all people together with the visibility flag, the bone names and their seat position. All the images are provided in .png format. The depth maps are saved in millimetres and as 16-bit .png images. The bounding boxes are given in the format [class, x_1, y_1, x_2, y_2], where (x_1, y_1) is the upper left corner and (x_2, y_2) the lower right corner of the bounding box (coordinates start in the upper left image corner). For classification, the labels are as follows: 0=empty seat, 1=infant in infant seat, 2=child on child seat, 3=adult passenger, 4=everyday object, 5=infant seat without baby, 6=child seat without child. For segmentation and object detection, the labels are: 0=background, 1=infant seat, 2=child seat, 3=person and 4=everyday object. We did not fasten the seat-belt for our models and let them un-attached in all our sceneries.

4. Baseline evaluation

In this baseline evaluation, we will show that SVIRO provides the means to analyze the performance of common machine learning methods under new conditions. We will test some widely used models and approaches for their robustness and reliability, when trained on limited number of variations only. Specifically, we will show that state-of-the-art models cannot generalize well to new environments and textures when trained on the previously discussed challenging, but realistic, conditions. For this evaluation, we limited ourselves to training on the X5 and testing on the Tucson (three seats) and i3 (two seats). For all tasks, we considered two training data versions (for which we used the exact same hyper-parameters): 1) the standard X5 training data with fixed textures and backgrounds (F), 2) half of the standard X5 training data is replaced by randomly textured X5 training data with random backgrounds (F&R).

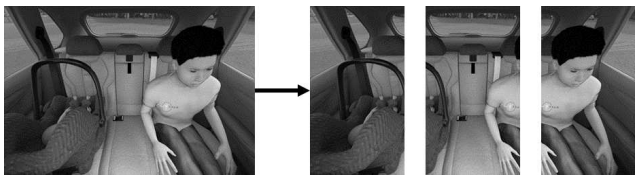

Figure 7. We split each image into three rectangles to use them for classification. The contents of the rectangles overlap slightly, because objects are not limited to their seat position.

We used the grayscale images (infrared imitation) for all the evaluations. For the deep learning-based approaches, we used the pre-defined models implemented in PyTorch 1.2 and Torchvision 0.4.0. For classification, we used pre-trained models on ImageNet. For semantic and instance segmentation, the models were pre-trained on COCO train 2017. The pre-trained models were fine-tuned on the X5 only and then evaluated on the test sets of all three cars. Using this approach, we could test the generalization capacities on two difficulty levels. The training dataset was partitioned randomly according to a 75:25 split for training and evaluation, where the latter was used to perform early stopping when fine-tuning the models. As we consider our F&R dataset as data augmentation, the only additional data augmentation performed was a random horizontal flip.

4.1. Classification

As introduced in Section 3.5, we used the rectangular graycale images for classification with seven different classes. One could decide to classify a seat with an everyday object (and an empty infant/child seat) as empty as well. We trained a single classifier for the three seats, but other setups are possible as well, e.g. train one classifier for each seat. In the following, we will report results on different deep learning models, as they are commonly used for visual classification problems. These results will be compared to a traditional method using a support vector machine (SVM) and handcrafted features. We will show that both methods suffer from the same problems and including the randomized F&R dataset overall improves the results.

4.1.1 CNN

We used the ResNet [42], DenseNet [43], SqueezeNet V1.1 [44] and MobileNet V2 [45] architectures and considered

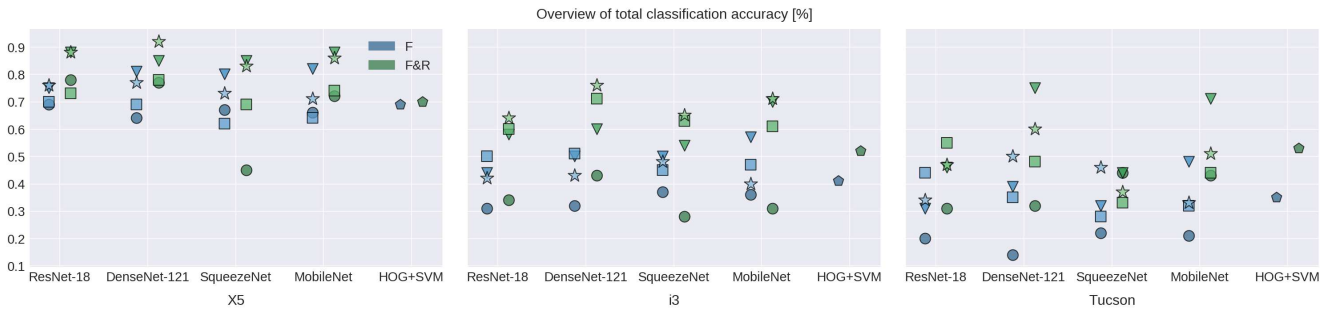

Figure 8. Comparison of different classification results. We trained several models from scratch (■) or fine-tuned pre-trained models, where all the weights (▼), the last block (●) or the last layer (★) were trainable. Further, we trained a SVM using HOG features. We used the standard X5 training data (F, in blue) or replaced half of it with the randomized data (F&R, in green). After training, we retained models with the best total accuracy on the X5 test data and evaluate them on the i3 and Tucson test data. The models have difficulties to generalize to the test data and perform even worse in unknown vehicles, but including the randomized data helps to generalize to unseen objects.

four different training approaches: 1) Training from scratch, 2) only fine-tuning the last fully connected layer, 3) additionally fine-tuning the last residual block, 4) allowing all weights to be trainable. We tried different combinations of weight decay, weighted costs and imbalanced sampling and report results for the best models only. In Figure 8, we compare the results across the different models and training approaches and compare them to the SVM. The deep learning-based approaches have problems to generalize to the test set, especially for new cars. The randomized backgrounds and textures help to improve the accuracy on the same car, which gives hint that models trained on the (F) dataset mostly use the texture as a classification criterion. However, the models can still not generalize well to new vehicle interiors, probably because of the different interior structures (see Figure 4). An exhaustive comparison between the different training approaches and hyper-parameters is available in our supplementary material.

4.1.2 HOG and SVM

For comparison, we also wanted to test at least one traditional machine learning-based approach for the classification task. To this end, we computed the histogram of oriented gradients (HOG) features of all the training images, and their horizontally flipped versions for data augmentation. These features were then used to train a SVM, using the "one vs. rest" approach and balanced class weights. We performed a grid search on different kernels (linear, polynomial and radial basis) and their hyper-parameters and used a 5-fold cross validation for hyper-parameter selection. We used scikit-learn 0.21.2 for the training and scikit-image 0.15.0 for the feature generation. The results for the best hyper-parameters are reported in Figure 8. Overall, the traditional approach has similar problems as the deep learning approach when the standard X5 data is used, and can sometimes even generalize better. However, it cannot exploit the additional information when random textures and backgrounds are included in the training.

Our dataset shows that traditional and deep learning approaches, although commonly used in practice, drastically decrease classification performance when trained in a setting with limited variations without taking additional precautions. No reliability can be guaranteed and both presented approaches do not fully grasp the underlying task, although the environment and the objects are similar. Including randomized images increases the performance, but to be applicable in real world applications further (theoretical) improvements need to be investigated and developed.

4.2. Semantic segmentation

It could be beneficial to take spatial information into account to improve the transfer to new instances and environments. Further, the model might consider overlapping objects from neighbouring seats more efficiently when the entire scene is used. To this end, we evaluated semantic segmentation and considered the five classes as introduced in Section 3.5. The model should separate the child from the child seat and the baby from the infant seat and classify them as a people. We fine-tuned all layers of a Fully Convolutional Network (FCN) with a ResNet-101 backbone and report the results in Figure 9. As for the classification results of the previous section, the model's performance decreases drastically on the child and infant seats on the test set for the same car and it performs even worse in previously unknown cars. Using the F&R training data, the generalization performance largely increased, although the geometry of the child seats of the test sets was never observed during training. It seems that the texture has a larger influence on the performance of classification and semantic segmentation models than the geometry. This observation seems to be in line with recent results by Geirhos *et al*. [46]. However, using SVIRO, we can additionally show that the model

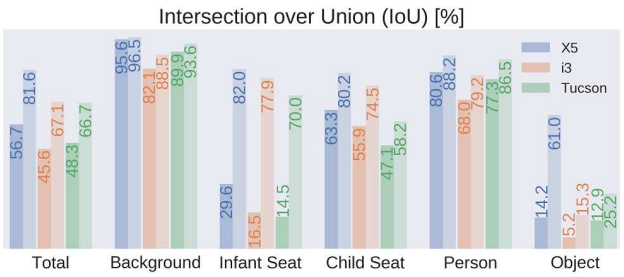

Figure 9. Mean intersection over union (IoU), in percent, for semantic segmentation for a fine-tuned pre-trained FCN. The dark colour represents the model performance when trained on the standard X5 training dataset (F) and the lighter colour when we included the random X5 data (F&R). The models were evaluated on the test dataset for the X5, Tucson and i3. Using the randomized version largely improves the generalization capacities of the model, especially for identifying infant seats and child seats.

cannot perform as good on new environments, even though the textures are randomized and the objects of the different test sets are the same.

5. Comparison with real images

We tested the transferability of a model trained on SVIRO to real infrared images and report results on instance segmentation to illustrate this. We fine-tuned all layers of a pre-trained Mask R-CNN model with a ResNet-50 backbone and considered the same classes as for semantic segmentation. The synthetic images were blurred to be closer to real infrared images. We combined the training images of the i3, Tucson and Model 3 and compare results on synthetic and real images in the X5 in Figure 10. More evaluations on real images are available in the supplementary material. Only bounding boxes and masks with a confidence of at least 0.5 are plotted. The model performs similarly across real and synthetic images and sometimes fails to detect objects. This is expected as the model has only seen a limited amount of variation. However, the similar child seat is detected in the real images, but not in the synthetic ones. We believe that investigations on SVIRO are transferable to real applications as the resulting model behaves similarly on real and synthetic images. Additional realistic effects could be applied to close the synthetic gap even further [47].

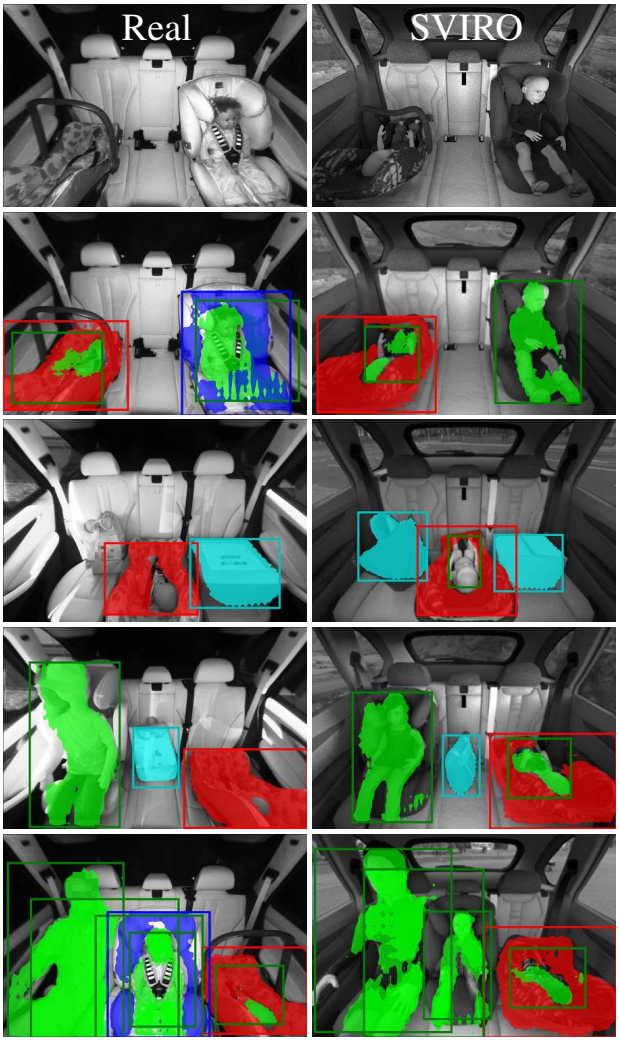

Figure 10. We acquired real active infrared images (first column) in an X5 and reproduced the same sceneries in Blender (second column). The first row compares real and synthetic images. The remaining rows compare instance segmentation mask predictions. The model performs similarly on both setups and the similar child seat is detected in the real images, but not in the synthetic ones.

6. Conclusion

We release SVIRO, a synthetic dataset for sceneries in the passenger compartment of ten different vehicles. Our benchmark addresses real-world engineering obstacles regarding the robustness and generalization of machine learning models. Using SVIRO, we showed in our baseline evaluation that common machine learning models, when trained on limited amount of variability, decrease in performance for solving the same task in a new vehicle interior. Models cannot generalize well to new intra-class variations, even in the car they were trained on. We believe that other research directions, e.g. (disentangled) latent space representation, scene decomposition, domain adaptation and uncertainty estimation, can benefit from our dataset.

Acknowledgement: The first author is supported by the Luxembourg National Research Fund (FNR) under the grant number 13043281. This work was partially funded by the MECO project "Artificial Intelligence for Safety Critical Complex Systems" and the European Union's Horizon 2020 Program in the project VIZTA (826600).

References

[1] E. Ohn-Bar and M. M. Trivedi, "Looking at humans in the age of self-driving and highly automated vehicles," *Transactions on Intelligent Vehicles (T-IV)*, 2016.

[2] L. Fridman, D. E. Brown, M. Glazer, W. Angell, S. Dodd, B. Jenik, J. Terwilliger, J. Kindelsberger, L. Ding, S. Seaman, *et al.*, "Mit autonomous vehicle technology study: Large-scale deep learning based analysis of driver behavior and interaction with automation," *arXiv preprint arXiv:1711.06976*, 2017.

[3] M. E. Farmer and A. K. Jain, "Occupant classification system for automotive airbag suppression," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003.

[4] T. Perrett and M. Mirmehdi, "Cost-based feature transfer for vehicle occupant classification," in *Asian Conference on Computer Vision (ACCV)*, 2016.

[5] S. Dias Da Cruz, H.-P. Beise, U. Schröder, and U. Karahasanovic, "A theoretical investigation of the detection of vital signs in presence of car vibrations and radar-based passenger classification," *Transactions on Vehicular Technology (TVT)*, 2019.

[6] A. R. Diewald, J. Landwehr, D. Tatarinov, P. D. M. Cola, C. Watgen, C. Mica, M. Lu-Dac, P. Larsen, O. Gomez, and T. Goniva, "Rf-based child occupation detection in the vehicle interior," in *International Radar Symposium (IRS)*, 2016.

[7] E. J. L. Pulgarin, G. Herrmann, and U. Leonards, "Drivers' manoeuvre classification for safe hri," in *Conference Towards Autonomous Robotic Systems*, 2017.

[8] R. McCall, F. McGee, A. Meschtscherjakov, N. Louveton, and T. Engel, "Towards a taxonomy of autonomous vehicle handover situations," in *International Conference on Automotive User Interfaces and Interactive Vehicular Applications (AutoUI)*, 2016.

[9] M. Tian, S. Yi, H. Li, S. Li, X. Zhang, J. Shi, J. Yan, and X. Wang, "Eliminating background-bias for robust person re-identification," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[10] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *European Conference on Computer Vision (ECCV)*, 2016.

[11] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[13] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[14] M. Braun, S. Krebs, F. B. Flohr, and D. M. Gavrila, "Eurocity persons: A novel benchmark for person detection in traffic scenes," *Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2019.

[15] M. Fabbri, F. Lanzi, S. Calderara, A. Palazzi, R. Vezzani, and R. Cucchiara, "Learning to detect and track visible and occluded body joints in a virtual world," in *European Conference on Computer Vision (ECCV)*, 2018.

[16] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision (IJCV)*, 2010.

[17] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[18] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, T. Duerig, and V. Ferrari, "The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale," *arXiv preprint arXiv:1811.00982*, 2018.

[19] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European Conference on Computer Vision (ECCV)*, 2014.

[20] C. P. Burgess, L. Matthey, N. Watters, R. Kabra, I. Higgins, M. Botvinick, and A. Lerchner, "Monet: Unsupervised scene decomposition and representation," *arXiv preprint arXiv:1901.11390*, 2019.

[21] M. Baltaxe, R. Mergui, K. Nistel, and G. Kamhi, "Markerless vision-based detection of improper seat belt routing," in *Intelligent Vehicles Symposium (IV)*, 2019.

[22] X. Peng, B. Usman, N. Kaushik, J. Hoffman, D. Wang, and K. Saenko, "Visda: The visual domain adaptation challenge," *arXiv preprint arXiv:1710.06924*, 2017.

[23] X. Peng, B. Usman, K. Saito, N. Kaushik, J. Hoffman, and K. Saenko, "Syn2real: A new benchmark forsynthetic-to-real visual domain adaptation," *arXiv preprint arXiv:1806.09755*, 2018.

[24] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *European Conference on Computer Vision (ECCV)*, 2016.

[25] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[26] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment matching for multi-source domain adaptation," *arXiv preprint arXiv:1812.01754*, 2018.

[27] P. Koniusz, Y. Tas, H. Zhang, M. Harandi, F. Porikli, and R. Zhang, "Museum exhibit identification challenge for domain adaptation and beyond," *arXiv preprint arXiv:1802.01093*, 2018.

[28] B. Moiseev, A. Konev, A. Chigorin, and A. Konushin, "Evaluation of traffic sign recognition methods trained on synthetically generated data," in *Advanced Concepts for Intelligent Vision Systems (ACIVS)*, 2013.

[29] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," *Neural networks*, 2012.

[30] J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. L. Zitnick, and R. Girshick, "Clevr: A diagnostic dataset for compositional language and elementary visual reasoning," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[31] Y. Chen, W. Li, X. Chen, and L. V. Gool, "Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[32] F. E. Nowruzi, P. Kapoor, D. Kolhatkar, F. A. Hassanat, R. Laganiere, and J. Rebut, "How much real data do we actually need: Analyzing object detection performance using synthetic and real data," *arXiv preprint arXiv:1907.07061*, 2019.

[33] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, and S. Birchfield, "Training deep networks with synthetic data: Bridging the reality gap by domain randomization," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[34] "Blender." https://www.blender.org.

[35] "Textures.com." http://www.textures.com.

[36] "Hdri haven." http://www.hdrihaven.com.

[37] Mindfront, punkdunk, MargaretToigo, Sontag78, and Elvaerwyn, "Makehuman." http://www.makehumancommunity.org.

[38] "Hum3d." http://www.hum3d.com.

[39] E. Q. (Backpack), cjohn259 (Bag), costorella (3Dx bag), andree (Mochila), and B. B. O. Bottles), "Sketchfab." http://www.sketchfab.com.

[40] E. B. Dam, M. Koch, and M. Lillholm, *Quaternions, interpolation and animation*, vol. 2. Citeseer, 1998.

[41] H. Piazena, H. Meffert, and R. Uebelhack, "Spectral remittance and transmittance of visible and infrared-a radiation in human skin—comparison between in vivo measurements and model calculations," *Photochemistry and photobiology*, vol. 93, no. 6, pp. 1449–1461, 2017.

[42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[43] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[44] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <0.5 mb model size," *arXiv preprint arXiv:1602.07360*, 2016.

[45] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[46] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness," *arXiv preprint arXiv:1811.12231*, 2018.

[47] A. Ley, R. Hänsch, and O. Hellwich, "Syb3r: A realistic synthetic benchmark for 3d reconstruction from images," in *European Conference on Computer Vision (ECCV)*, 2016.