2014 IEEE Intelligent Vehicles Symposium (IV)
June 8-11, 2014. Dearborn, Michigan, USA

Semivirtual Simulations for the Evaluation of Vision-based ADAS

Marc René Zofka, Ralf Kohlhaas, Thomas Schamm and J. Marius Zöllner[1]

***Abstract*— The design and development process of advanced driver assistance systems (ADAS) is divided into different phases, where the algorithms are implemented as a model, then as software and finally as hardware. Since it is unfeasable to simulate all possible driving situations for environmental perception and interpretation algorithms, there is still a need for expensive and time-consuming real test drives of thousands of kilometers. Therefore we present a novel approach for testing and evaluation of vision-based ADAS, where reliable simulations are fused with recorded data from test drives to provide a task-specific reference model. This approach provides ground truth with much higher reliability and reproducability than real test drives and authenticity than using pure simulations and can be applied already in early steps of the design process. We illustrate the effectiveness of our approach by testing a vision-based collision mitigation system on recordings of a german highway.**

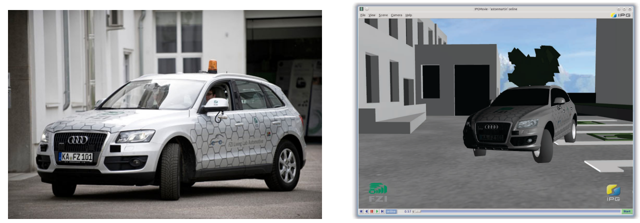
I. INTRODUCTION

Since the DARPA's Urban Challenge in 2007, research for highly automated driving has been pushed from academic to industrial research. Beside typical highly automated cars of academic institutions [1], [2], [3], Daimler recently has passed the Bertha-Benz Memorial Route in the south of germany with a prototype of an highly automated car. This car has been equipped with close-to-production sensors and algorithms. There is increasing research in developing more and more advanced driver assistance systems (ADAS), which will facilitate the driver in his driving task.

A typical ADAS consists of several information processing steps, as shown in figure 2. In the first step the particular environment is sensed with different sensor principles. This sampled information is enhanced using perception and interpretation algorithms. Based on the current situation assessment an appropriate action for the ego vehicle is planned and executed.

Current research concerns with facilitating and easing of the development and verification of safety functions, so that the public acceptance is raised. Although in some american states autonomous driving is permitted, it is not the matter in other countries like germany. So the challenge is to create frameworks and tools which will enable the systems engineer to make statements about some algorithms performance. These statements have to be more reliable than using simulations or test drives on their own. This is necessary to bring an increasing number of complex advanced driver assistance systems to production and to reduce the severity of accidents, to decrease traffic congestions and fuel consumption.

[1]The authors are with Research Center for Information Technology (FZI), 76131 Karlsruhe, Germany {zofka, kohlhaas, schamm, zoellner}@fzi.de



(a) CoCar, the Cognitive Car from the FZI Research Center for Information Technology[2].

(b) CoCars' physical and sensory mapping for simulation.

Fig. 1. Fusing the two worlds of simulation and real test drives.

A typical development process model in automotive systems engineering is the V-Model, as it is proposed in the ISO 26262 standard of functional safety in automotive electronics [4]. According to this, algorithms are usually implemented on different platforms and evaluated by simulating the respective environment. This methods are called model-in-the-loop (MiL), software-in-the-loop (SiL) or hardware-in-the-loop (HiL). Thereby the respective response of the algorithm can be evaluated with the expected one. Appropriate ground truth can be provided for evaluation, by using simulations as reference systems. Because of the increasing complexity and integration of different ADAS also the amount of influences on a certain driving situation increases intractable, like variations on weather, on pedestrian or vehicle movement. Finally, at the end of the development process the real test drive helps to assure that no mistakes have been made in the specification phase and unexpected reactions of the system will not occur. So thousands of kilometers are passed to make reliable statements of the performance of an ADAS. This is an expensive and time-consuming testing method and lacks of reproducability of critical scenes.

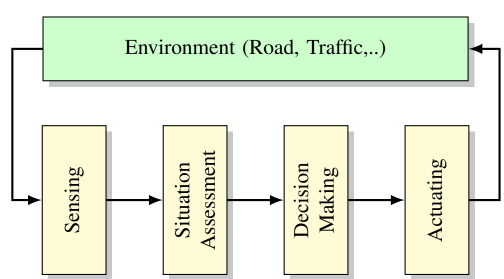


Fig. 2. Information processing in an ADAS for highly automated driving. The vehicle composes a feedback loop with the environment.

[2]For more information on CoCar, please visit http://www.fzi.de/en/forschung/projekte/cocar/

This paper presents a novel approach for fusing simulations and real test drives to develop and evaluate vision-based environmental algorithms from an early step in the information processing chain. We demonstrate our approach using an exemplary vehicle detection and tracking framework with real test drive data from our highly automated car. Finally we discuss our results and illustrate future work.

II. RELATED WORK

More and more research is undertaken to simplify the evaluation and verification of ADAS systems. One branch of current research tries to minimize the simulated test cases by structuring situations and constructing appropriate equivalence classes of test cases, as shown in [5]. In [6] an exemplary description language is presented to generalize and structure test cases for simulations. Other research examines how to simplify the cost-intensive and time-consuming real test drives. For example in [7] an approach is presented to monitor incidences with the underlying environmental situations and vehicle states during the real test drive, to derive ADAS-specific information, that can be used to reduce the necessary kilometers.

Another branch tries to combine the advantages of virtual and real test drives, called vehicle-in-the-loop (ViL) [8]. Thereby a collision mitigation system is tested by driving an experimental vehicle on a free testing ground, while the sensors' response is simulated from a virtual test drive and delivered to the subsequent steps of the algorithmic processing pipeline. So the real vehicle dynamics can be used with simulated sensor outputs to verify the functionality of an adaptive cruise control (ACC) [8].

Based on an exemplary localization reference system in [9], the authors in [10] carve out basical requirements for any environmental perception system. In [11] real test data is used to derive comparable virtual test drives and simulation models with respect to environmental influences towards optical sensors. Both, real camera recordings and simulated cameras are used to create a reference system for the image processing algorithmic parts of a lane and vehicle detection algorithm. In [12] augmented data sequences are proposed to evaluate a pedestrian detector, but the approach is limited to offline rendering and does not make use of additional sensor data, like gps information.

To demonstrate our approach, we utilize a vision-based collision warning system, as visualized exemplary in figure 3. The underlying vehicle detection and tracking system is based on a mono camera, which senses the environment like the road or traffic in front of the ego vehicle. The exemplary vehicle detection algorithm we used, is based on the work of Viola and Jones [13] and HOG-based SVM classification, similar to [14]. Thereby search windows of different sizes are shifted over each camera image, wherein Haar-Wavelet-based features are determined. Those features are evaluated in a classifier cascade, where each layer represents an own weak classifier. Although the detection phase is realtime-capable, the learning phase is time-consuming, because the classifier cascade has to be trained with labeled training data. This statistical learning algorithm is called AdaBoost.

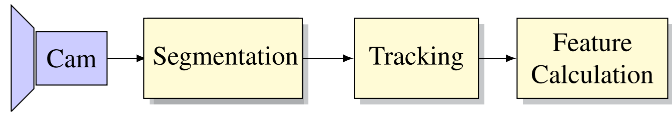


Fig. 3. Block diagram of an exemplary vision-based ADAS, following the tracking-by-detection framework. First the images from the sensor are obtained, then segmentation is performed. Those extracted features are tracked over space and time and facilitate the calculation of features, which describe the units in the observed environment.

After the detection phase the vehicles found in an image are tracked over time. This step is performed using common filter techniques like extended kalman filters or particle filters, for example. They usually consist of two basic steps: In the first step an initial state of the traffic vehicle is propagated to the next timestep with a dynamic system model. Then, sensor observations are used to update and correct the propagated state of the vehicle. Thereby outliers can be rejected and additional features of the environmental objects can be determined, like speed, acceleration or movement direction of traffic participants.

III. CONCEPT

In this section we present the concept of augmenting recorded camera images from real test drives with synthetic, simulated camera views as input for the subsequent environmental perception algorithmic components.

Figure 4 demonstrates the proposed process. Our concept is based on three steps. In the first step the test vehicle is used to record video streams of different driving scenes with high precision localization. In the second step we use those recorded situations to derive simulation scenarios, based on the same setting we have encountered during the test drive. In the last step we extract segmented environmental objects from the synthetic view and overlay the recorded camera stream with them. Finally we can use the ground truth from the simulation to verify the subsequent algorithmic parts. With the ground truth from the simulation, we can even verify the calculation of features of the traffic participants, like speed. Because of its analogy, one can denote this as augmented reality testing in the sense of Milgram [15], or semivirtual simulation.

A. Real test drive

In a first step the test vehicle is equipped with the optical camera sensor system. Furthermore, a high-precision localization system is integrated, so that the passed trajectory can be read-out and reconstructed later on. While driving the test track, the sensor outputs are collected on CAN or network busses, depending on the interfaces of the particular ECUs. The GPS stream has to be synchronized with the video data stream of the optical sensor system. Therefore we assign global timestamps. Finally, this data is stored in a stream-oriented database.

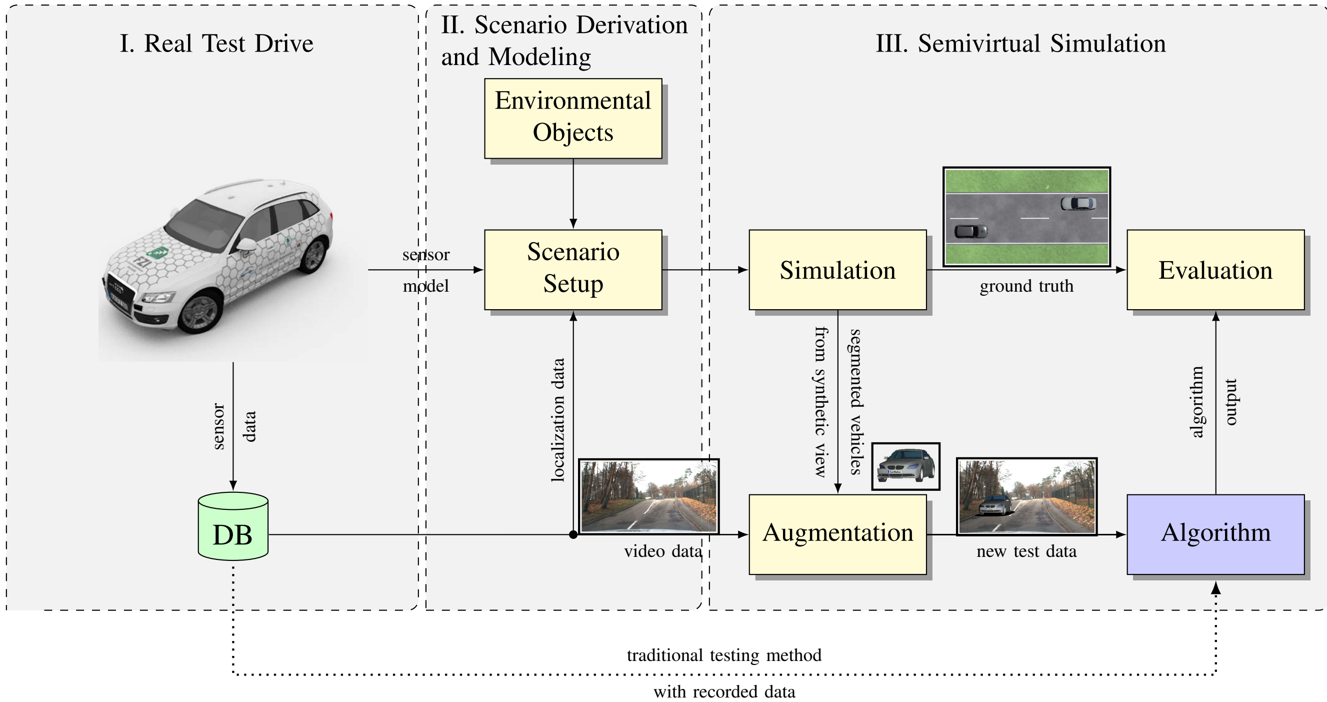


Fig. 4. Our approach to evaluate a vision-based ADAS consists of three basic steps: First, a real test drive is passed to obtain recorded sensor data streams. Then in laboratory the recordings are used to derive and model the appropriate driving situation in a simulation framework. Finally, we can extract exactly those environmental objects from a synthetic camera view, the algorithmic tries to detect in the recorded video streams. The great advantage is, that we have an exact ground truth for the objects from the simulation.

B. Derivation and Modeling of Test Scenario

We use the collected data to derive a certain virtual test drive model, which is consistent with the recorded driving scenes. Thus we extract the test track with the course of the passed trajectory as a sequence of GPS points with longitude, latitude and altitude. To generate a complete road model additional information of the geometric configuration of the roads is necessary, like the amount of lanes or their width. During the real test drive the GPS can be affected by multipath effects caused by street canyons or high trees. These outliers will also affect the road model. Beside a road model we also have to provide an appropriate geometric model of the test vehicle itself. Here the geometric positioning of the optical sensor setting relative to the vehicle plays the major role. At least the optical sensor setting with correct extrinsic and intrinsic parameters have to be modeled to create an appropriate synthetic view.

Depending on what kind of detector is tested, we can add dynamic objects like vehicles and pedestrians with different movement patterns. The road model also can be augmented with static objects, like traffic signs, for which the underlying simulation provides ground truth then. This virtual test drive now can be replayed in simulation.

C. Semivirtual Simulation

Using an appropriate simulation is a crucial task for our approach. Many simulation frameworks in the domain of SiL and HiL-testing are available, for example see TASSPrescan[3], IPG CarMaker[4], CarSim[5] or the open-source alternative OpenDS[6]. The simulation tool has to fulfill the following requirements, so that we can use it for semivirtual simulations: It has to offer the possibility to define a road model using local or global coordinates, so that the simulation traffic can be registered with the real test run. Then, during simulation it is necessary to move the ego vehicle using a velocity profile or even move it according to the recorded timestamped GPS position w. r. t. the relative simulation time. If the road model has been created from a smoothed GPS trajectory, the smoothing also has to be applied for moving the simulated vehicle. Another capability is that the simulation framework offers the possibility to define synthetic camera views, which can be mounted on a geometric vehicle model. Futhermore it would be a desirable function to exclude specific environmental objects from the rendering pipeline, so that we only obtain objects, the detection algorithm is trained for. Nevertheless, image processing algorithms for segmentation can be used to extract these important components.

The augmentation process is carried out as follows: For each recorded camera image, we move the virtual ego vehicle according to a chosen velocity profile or the real GPS position. Then, the scene is rendered and we receive a synthetic camera view. This image is segmented and only the traffic objects are obtained. Now the recorded video data

[3]https://www.tassinternational.com/prescan
[4]http://www.ipg.de
[5]http://carsim.com
[6]http://www.opends.eu

stream is augmented with the extracted vehicle objects. This augmented camera stream is used with the recorded GPS stream as input for the subsequent algorithmic parts. With the knowledge about the underlying scenario from simulation, we have a complete new testbed for evaluation on recorded camera streams with synthetic overlays, see figure 5.

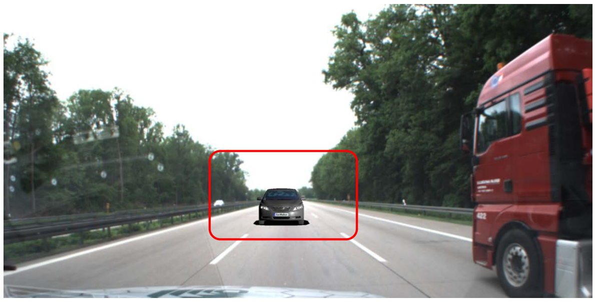


Fig. 5. Concept of semivirtual simulations: Objects of interests are segmented from a synthetic camera view and real test drives augmented with them.

This technique, as visualized in figure 4, can be applied in SiL and HiL, but is limited to open-loop testing. This means that no feedback control is modeled, and we are not able to evalute the actuation layer of an active ADAS, as shown in figure 2.

D. Towards Vehicle-in-the-loop Testing

Our proposed testing method is also applicable for vehicle-in-the-loop-testing, so that an ADAS can be evaluated as a whole in closed loop. Therefore our concept has to be adapted, as visualized in figure 6.

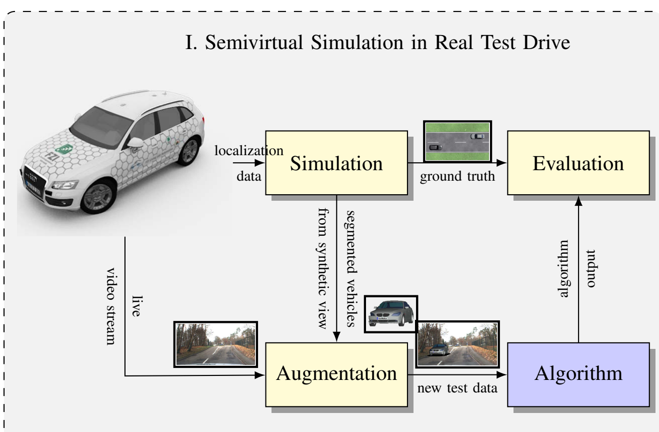


Fig. 6. Concept of semivirtual simulations for vehicle-in-the-loop testing of vision-based ADAS. The scenario model has to be predefined.

The simulated environment with its road model has to be registered within the global coordinate system of the free testing ground, we want to drive on. Then traffic participants can be placed on the virtual road model. The recorded video stream is now replaced by the real video stream of the optical sensor. Then while driving the position of the ego vehicle in the virtual environment is updated permanently. Then we can augment the live video stream with the segmented traffic objects and visualize this to the test driver or pass it also to the subsequent vision-based ADAS systems, like an collision mitigation system, for example. So vision-based ADAS can be tested in a closed-loop manner.

IV. IMPLEMENTATION

We have implemented this approach based on our own data synchronisation framework. Thereby we can record data from different sensors, like cameras, LIDAR or GPS sensors with global timestamps, so that we can replay them synchronously to evaluate a real test drive in laboratory.

A. Simulation framework

For simulation we took a commonly used hardware-in-the-loop tool, the IPG CarMaker, which implements an own rendering engine for simulating the environment. It has also the capability to define simple road networks and sensor models. For our application we have to modify the rendering pipeline to extract only the objects-of-interest, the traffic vehicles. Therefore we define a single-colored skybox and set the roard texture to invisible. Then we use a color-based segmentation algorithm to discard all skybox and environment colours to obtain an image mask, which contains the resulting traffic objects. These steps are visualized in figure 7.

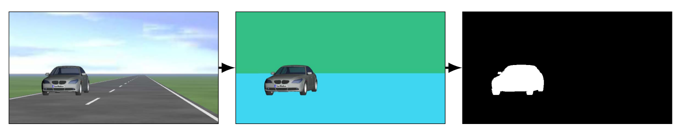


Fig. 7. Using CarMaker for traffic scene simulation (left). We have to modify the textures for road and gras and to create a customized skybox (middle), so that we can segment the traffic objects via a color-based image processing algorithm. Finally, we use the resulting image as a mask (right) to augment the original scene with the traffic objects.

B. Testing method

In the preprocessing step the GPS stream of the recorded test drive is parsed, smoothed and exported as a road model to the CarMaker. Due to CarMaker restrictions the passed trajectory is modelled as single road lane. Additionally we include a-priori knowledge by defining additional lanes and their specific width. Futhermore, the geometric vehicle model including the positioning and orientation of the optical sensor system is defined in the simulation. While replaying the recorded sensor data streams and the virtual test drive simultaneously, the traffic is segmented color-based and augmented on the camera image. The resulting video stream is used as input for the collision warning system and the simulation setting is used as ground truth to evaluate the algorithm. The algorithm then detects simulated and real vehicles.

V. EVALUATION

We demonstrate the effectiveness and applicability of our testing method creating a critical driving scenario. Then we show the scientific challenges we have to consider, when creating a semivirtual simulation framework.

We demonstrate our approach using recorded data from the FZIs experimental vehicle CoCar. It's a modified Audi Q5, which is equipped with different environmental sensors and gas, brake and steering actuators. Thereby it is able to drive fully automated, as presented in [1]. Beside a mono and stereo optical sensor system, it is also equipped with an array of IBEO laserscanner. A high precision D-GPS/IMU coupled system OxTS RT3003 provides a high detailed localization. For the authentic generation of synthetic camera images we build a geometric model of CoCar in the simulation, including the appropriate sensor mounting, see figure 1.

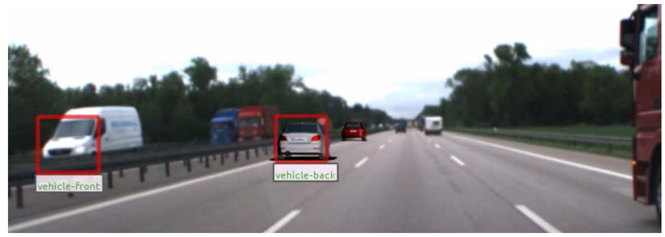
With real test drives it is almost impossible to trigger and produce critical situations without the danger of human risks. Therefore we passed a test drive on a German highway to collect synchronized GPS and video streams. Then afterwards, we augment the recordings with a dangerous situation to evaluate our vision-based ADAS: A traffic participant performs an emergency brake and the successive driver has to avoid him by changing his lane. The results of the vehicle detection algorithm are visualized in figure 8. Thereby both simulated and real vehicles are detected by the same detector, if they fall below a certain distance range.

This kind of simulation is applicable in early phases in the development process of an ADAS. Using the ground truth from the simulation, different features of the traffic objects can be compared (TTC, relative distance, . . .) or even the input signals for the actuating components can be tested. Also in the vehicle-in-the-loop approach we can setup critical driving situations, if a wide testing ground or long test road is available. Then a virtual highway scenario can be put onto it and the collision mitigation system can be tested with its decision making and vehicle actuating components, see figure 2.

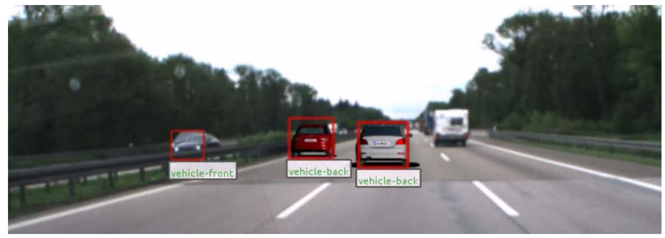
A very basic requirement for this method to work is that the augmentation process retains the consistency of the features. Therefore the features, the vehicle detector is based on, have to outlast the augmentation process. Since our detector is based on Haar-Wavelets, we have to consider the effect of anti-aliasing, which could smoothes important edges of virtual objects. In figure 9 the impact of additional anti-aliasing is visualized. We recognized in our scenario, that the results are almost the same and anti-aliasing only improves the detectability of vehicles with a larger distance to the ego vehicle. So we can conclude, that for the sake of computing time anti-aliasing is negligible, especially if we want to use it online in vehicle-in-the-loop testing.

We also recognized, that shadows under the simulated vehicles are a crucial feature for the success of our vehicle detector. In figure 10 we displayed an exemplary comparison of the same frames, but with different vehicle models.

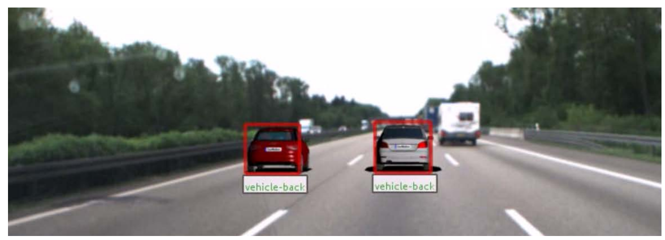
Thus we don't need photorealistic rendering of synthetic camera views for the application of our testing approach.



(a)



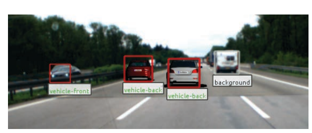
(b)



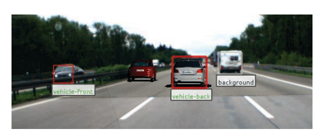
(c)

(d)

Fig. 8. Simulation of a critical situation on the german highway A5. We have augmented two simulated vehicles onto a certain driving scenario on the german highway A5 (a). The red Audi on the left lane performs an emergency brake, so that the successive car has to react (b) and changes to our lane, as shown in (c). Finally we approach it in figure (d). If the ADAS calculates the time-to-collision or considers the relative velocity of the ego vehicle, this could be an exemplary decision for us to brake.



(a)



(b)

Fig. 9. Comparision on the impact of additional anti-aliasing (a) and without anti-aliasing (b).

Rather it is important that the features, on which the detector is based on, are reproduced and can be extracted from the simulation. So we do not need to re-train the classifier, which usually takes a long time from days to weeks. If a real vehicle is not detected by the classifier (true negative), we can conclude from the synthetic true positives, what kind

(a) (b)

Fig. 10. Comparision on the vehicle detection of an ordinary vehicle model and the same model with an additional shadow plane under it.

of characteristic vehicle features might be missing in the training data and vice-versa. This is a simple way to gain insight into the quality of a classifiers training dataset.

There are also several limitations, which have to be considered. The first concept is only valid, if testing the ADAS in an open-loop manner fulfils the requirements of the specific development phase. An intervention of the ADAS to the simulated vehicle, like an emergency brake, would cause a drift and inconsistency between simulation and recordings. While an ordinary GPS sensor can be used for open-loop testing, for closed-loop testing in vehicle-in-the-loop a high-precision IMU is needed. Then the exact position can be passed to the ego vehicle in the simulation within an uncertainty of few centimeters.

VI. CONCLUSION

Typically simulation gives higher confidence in system reliability before going on the road. In contrast real data is much more reliable through authentic influences but lacks of reproducability. We combine both approaches by presenting the concept of semivirtual simulation of vision-based sensor systems. This holds several advantages. In an early step of the development and design process the need for many test drive kilometers can be reduced. Especially if the underlying perception interpretation algorithms are changed, an exact re-run of the driven scenario can be performed and the previous and current outputs can be compared. Also dangerous scenarios can be tested reliable in an open-loop manner. If the sensor setup is changed, it is only a little effort to collect new data streams and to adapt the sensor model in the simulation accordingly.

Developing and implementing this concept, we have found some strong statements about the requirements on an environmental simulation framework for semivirtual simulations. Instead of photorealistic rendering, we only need to extract the same features the classifier is based on. So we do not need to perform a new time-consuming re-training of the classifier.

We also have shown how this approach can be used for software or hardware-in-the-loop, and also for vehicle-in-the-loop. This has the advantage, that the full information processing pipeline can be tested in a closed-loop way. With this approach critical scenes can be generated, which can be replayed on a free testing ground without any danger.

VII. OUTLOOK

We have shown the applicability of our testing concept for SiL and HiL testing. For ViL applications we will implement this approach on the FZIs experimental vehicle CoCar, as disposed in the concept section III-D. Using vehicle-in-the-loop we will be able to evaluate and verify the whole information processing pipeline of a vision-based ADAS. Also we want to extend the open-loop testing by registering traffic object information, we have encountered during the real test drives. The simulated vehicles then can integrate themselfes into the recorded flow of traffic. Future work will also consider applying this concept to other kind of sensor simulation, like LIDAR for example. Also it is important to check wether semivirtual simulations can be extended to verify sensor data fusion algorithms.

ACKNOWLEDGMENT

This work was partially supported by the project ReTivu under the Federal Ministry for Economic Affairs and Energy (BMWi) grant KF2076929SS3.

REFERENCES

[1] R. Kohlhaas, T. Schamm, D. Lenk, and J. M. Zöllner, "Towards driving autonomously: Autonomous cruise control in urban environments," in *Intelligent Vehicles Symposium*, 2013, pp. 116–121.

[2] M. Goebl, M. Althoff, M. Buss, G. Farber, F. Hecker, B. Heissing, S. Kraus, R. Nagel, F. P. Leon, F. Rattei, M. Russ, M. Schweitzer, M. Thuy, C. Wang, and H. J. Wuensche, "Design and capabilities of the Munich Cognitive Automobile," in *Intelligent Vehicles Symposium, 2008 IEEE*, 2008, pp. 1101–1107.

[3] A. Broggi, M. Buzzoni, S. Debattisti, P. Grisleri, M. C. Laghi, P. Medici, and P. Versari, "Extensive Tests of Autonomous Driving Technologies." *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1403–1415, 2013.

[4] ISO, "Iso 26262: Road vehicles - functional safety," International Organization for Standardization, Tech. Rep., 2010.

[5] F. Schuldt, F. Saust, B. Lichte, and M. Maurer, "Effiziente systematische Testgenerierung für Fahrerassistenzsysteme in virtuellen Umgebungen," in *AAET2013 - Automatisierungssysteme, Assistenzsysteme und eingebettete Systeme für Transportmittel, Braunschweig*, 2013.

[6] C. Berger, "Design considerations for a cyber-physical testing language on the example of autonomous driving," in *Proceedings of the 2012 workshop on Domain-specific modeling - DSM '12*. New York, New York, USA: ACM Press, Oct. 2012, p. 49.

[7] P. D. Glauner, A. D. Blumenstock, and M. D. Haueis, "Effiziente Felderprobung von Fahrerassistenzsystemen," in *8. Workshop Fahrerassistenzsysteme FAS2012*, 2012, pp. 5–14.

[8] T. Bock, M. Maurer, and G. Farber, "Validation of the Vehicle in the Loop (VIL); A milestone for the simulation of driver assistance systems," in *2007 IEEE Intelligent Vehicles Symposium*. IEEE, June 2007, pp. 612–617.

[9] B. Strasser, "Vernetzung von Test- und Simulationsmethoden für die Entwicklung von Fahrerassistenzsystemen," Ph.D. Thesis, Technische Universität München, 2012.

[10] M. Brahmi, K.-H. Siedersberger, Siegel A., and Maurer M., "Reference Systems for Environmental Perception: Requirements, Validation and Metric-based Evaluation," in *6. Tagung Fahrerassistenzsysteme*, München, 2013.

[11] M. Nentwig and M. Stamminger, "Hardware-in-the-loop testing of computer vision based driver assistance systems," in *2011 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, June 2011, pp. 339–344.

[12] J. Nilsson, A. Ödblom, J. Fredriksson, and A. Zafar, "Using augmentation techniques for performance evaluation in automotive safety," in *Handbook of Augmented Reality*. Springer, 2011, pp. 631–649.

[13] P. Viola and M. Jones, "Robust real-time object detection," in *International Journal of Computer Vision*, 2001.

[14] F. Han, Y. Shan, R. Cekander, H. S. Sawhney, and R. Kumar, "A two-stage approach to people and vehicle detection with hog-based svm," in *In Performance Metrics for Intelligent Systems 2006 Workshop*.

[15] P. Milgram, H. Takemura, A. Utsumi, and F. Kishino, "Augmented Reality: A Class of Displays on the Reality-Virtuality Continuum," 1994, pp. 282–292.