IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 7 October 2022, accepted 2 November 2022, date of publication 9 November 2022, date of current version 14 November 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3220904

RESEARCH ARTICLE

The Inadequacy of Discrete Scenarios in Assessing Deep Neural Networks

KEN T. MORI, XU LIANG, LUKAS ELSTER, AND STEVEN PETERS
Institute of Automotive Engineering, Technical University of Darmstadt, 64289 Darmstadt, Germany

Corresponding author: Ken T. Mori (ken.mori@tu-darmstadt.de)

This work was supported by Virtual Validation Toolchain for Automated and Connected Driving (VIVID), Promoted by the German Federal Ministry of Education and Research, based on a Decision of the Deutsche Bundestag under Grant 16ME0173.

ABSTRACT Many recent approaches for automated driving (AD) functions currently include components relying on deep neural networks (DNNs). One approach in order to test AD functions is the scenario-based approach. This work formalizes and evaluates the parameter discretization process required in order to yield concrete scenarios for which an AD function can be tested. Using a common perception algorithm for camera images, a simulation case study is conducted for a simple static scenario containing one other vehicle. The results are analyzed with methods akin to those applied in the domain of computational fluid dynamics (CFD). The performance of the perception algorithm shows strong fluctuations even for small input changes and displays unpredictable outliers even at very small discretization steps. The convergence criteria as known from CFD fail, meaning that no parametrization is found which is sufficient for the validation of the perception component. Indeed, the results do not indicate consistent improvement with a finer discretization. These results agree well with theoretical attributes known for existing neural networks. However, the impact appears to be large even for the most basic scenario without malicious input. This indicates the necessity of directing more attention towards the parameter discretization process of the scenario-based testing approach to enable the safety argumentation of AD functions.

INDEX TERMS Artificial intelligence, autonomous vehicles, concrete scenarios, deep learning, error testing, intelligent vehicles, logical scenarios, machine learning, neural networks, software testing.

I. INTRODUCTION

Scenario-based testing has become indispensable for the release of AD functions, as the real, necessary kilometers driven exceed any financial budget [1]. For this reason, attempts are being made to use simulations and thus reduce costs. There are already some frameworks and simulation environments that are able to simulate different scenarios with their variety of parameter combinations [2], [3], [4]. A predominant classification represents the subdivision of the scenarios into functional, logical and concrete scenarios [5]. The degree of abstraction decreases from the functional to the concrete scenarios and the level of detail of the scenario description increases. In logical scenarios, parameters are defined that can take up a certain range between two specified limit values. For the concrete scenarios, this continuous space is discretized for each parameter and different parameter combinations are tested to make statements for previously defined metrics such as the criticality of a scenario. In the scenario-based testing community two different approaches exist. The first approach tries to identify for each parameter valid probability distributions with density functions to identify critical parameter combinations [6]. The disadvantage here is that the parameter distribution has to be known and therefore a lot of real traffic data has to be accessible. The other approach is to sample the parameter space within range [7]. The advantage is that no prior knowledge regarding the distribution is necessary to perform a discretization of the parameter space. However, there is a lack of methods to argue sufficient coverage over the validation space [8].

Another important part of enabling AD functions are DNNs which are applied to an increasing number of tasks [9].

The associate editor coordinating the review of this manuscript and approving it for publication was Ganesh Naik.

The tasks include perception, prediction and planning where DNN have shown large progress in the last years [10]. Especially for perception, DNN methods lead benchmark datasets [11] and outperform traditional methods by large margins [12]. Despite the performance of these methods, concerns remain regarding their use in safety-critical systems [9], [13] resulting in new standardization projects to relate safety and artificial intelligence [14].

Therefore, this paper addresses the sampling of the parameter space within range for the scenario-based testing of AD function as described by [7], specifically with respect to achieving sufficient coverage of the parameter space for testing of a DNNs.

II. RELATED WORKS IN LITERATURE

This work addresses the discretization of parameter spaces for the testing of AD functions. First, existing approaches to perform this discretization are discussed. In order to argue safety of the function for all parameters in the parameter range beyond the tested values it is also necessary to consider interpolation. This is followed by the general properties of DNN which affect the interpolation errors. Finally, the basics from the domain of CFD are introduced in order to later adapt them to estimate the discretization error introduced in the scenario-based testing of AD functions.

A. DISCRETIZATION

A number of works attempt to generate concrete scenarios by discretizing parameter ranges. Most of these works focus on assistance systems such as lane keep assistance (LKA) [15] or active cruise control (ACC) [16]. However, these works typically provide no argumentation for the discretization which is chosen [8], [15], [16], [17]. Similar observations are made for other works specifically considering perception systems with DNN [18], [19]. Another work does not consider discretization explicitly, instead defining it implicitly by specifying the number of scenarios to generate [20]. Overall, parameter discretization for the development of concrete scenarios still lacks a safety argumentation.

B. INTERPOLATION

Practical effort of test execution limit the number of concrete scenarios which can be tested. However, other parameter values lying between the tested values may be encountered in the real world. The validation procedure should therefore consider interpolation within the validation domain to guarantee the safety requirements are also met for parameter values which could not be tested [21], [22]. Therefore, this work attempts to explicitly quantify the errors induced during the interpolation procedure. For interpolation of results, the mathematical properties are well understood. Bounding the error of an interpolation requires bounded derivatives of the underlying function. Given such a smooth function, the interpolation error is proportional to h^{n+1} for a given discretization step h and an interpolation of order n [23]. The same is true for a piece wise linear interpolation [24]. This formal order of the interpolation is required in order to apply convergence criteria from the domain of CFD.

C. DEEP NEURAL NETWORKS

Current DNN are used to approximate arbitrary functions. The feedforward network typically uses an activation function such as a rectified linear unit (ReLU) for the internal processing [25]. The activation function is applied by each neuron within a typical feedforward network [26]. While the ReLU activation function is frequently applied [27], many different activation functions have been proposed [28]. The mathematical properties differ between different activation functions. While the ReLU function is not, other activation functions such as Mish and Swish are smooth [28].

In convolutional networks, multiple convolutions and activation functions are sequentially applied [29]. For non smooth activation functions, no bounding for the error is possible. However, even for smooth activation functions it is not clear if a bounding of the error is achieved for the overall network. DNN generally display a lack of robustness towards adversarial pertubations which may even occur for highly trained networks [26]. While there are various hypotheses on the existance of adversarial examples, their existence for various architectures has been empirically shown [30]. However, the lack of robustness observed for adversarial examples relates to malicious input. This work attempts to quantify the error for a practical scenario discretization and interpolation of the results.

D. ERROR ESTIMATION IN COMPUTATIONAL FLUID DYNAMICS

This section details the methods developed and applied in this work based on concepts in the domain of CFD. This work attempts to leverage existing knowledge from CFD regarding discretization and verification of simulation results. The content in this section is taken from [21] unless noted otherwise.

While CFD also includes other error sources such as the computation of a discrete solution, the general problem statement of discretizing a grid and interpolating results is similar to scenario discretization. In order to estimate errors, the grid is required to be in the asymptotic range where errors show convergence. To ensure asymptotic range, the solutions f_i are calculated on three successively refined grids. For a constant grid refinement factor r it is possible to calculate the observed order of accuracy p as:

$$p = \frac{\ln\left(\frac{f_3 - f_2}{f_2 - f_1}\right)}{\ln r} \tag{1}$$

This observed order of accuracy represents the behavior of the numerical solution which is compared with the formal order. The formal order is a mathematical property of the numerical scheme which can be examined with a Taylor expansion. Only if the formal order and the observed order match, the discretization error estimates can be expected to be accurate.

If the asymptotic range is achieved, the error can be estimated using methods such as Roache's grid convergence index (GCI):

$$\mathrm{GCI} = \frac{F_\mathrm{s}}{r^{p-1}} \cdot \left| \frac{f_\mathrm{h} - f_\mathrm{rh}}{f_\mathrm{h}} \right| \tag{2}$$

The GCI considers a safety factor F_s, the grid refinement factor r and the normalized error between two grids. The GCI provides an estimate of the upper bound of the error of the solution obtained on the respective grid. Similarly, the objective in the scenario domain is to obtain and upper bound of the error regarding the safety for the respective parameter discretization.

III. METHODS

This section details the methods used in this work to assess the discretization of scenarios relying on the results from literature stated above.

In order to do so, an exemplary system under test (SuT) and logical scenario are chosen for evaluation. The concrete scenarios yielded by a discretization of the parameter space are then executed in a suitable simulation environment. This simulation result is then evaluated regarding safety by computing a safety score. These results are then compared with the methods for error estimation adapted from CFD. Each component is chosen as a minimum working example to illustrate the basic structure of the discretization process.

An overview of the method is provided in Fig. 1 and each of the steps is discussed in the following subsections.

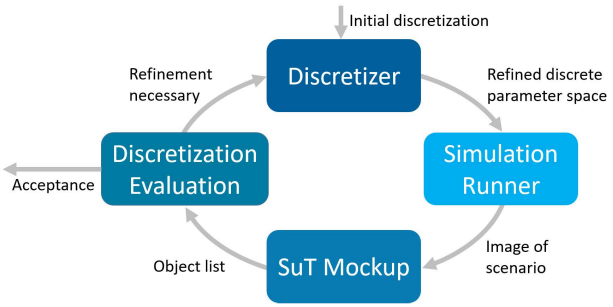

FIGURE 1. Overview of the applied analysis method.

A. SYSTEM UNDER TEST

The SuT is chosen to offer a simple implementation while including attributes generally also found in more complex systems. AD systems generally include DNN components which are by nature black box systems. This work chooses a perception algorithm since this allows treating the system as open loop. A simple case is a camera based 2D object detection algorithm which relies on a DNN. The objective of the algorithm is to predict the object category as well as the location with a bounding box [12], more specifically a 2D bounding box in image coordinates. A pretrained YOLOv5 [31] is used as a common object detector for simple implementation. It is worth noting that YOLOv5 uses the SiLU activation function which is smooth [32].

B. SCENARIO SIMULATION

For simplicity, the scenario is kept entirely static. This means that each scenario can be characterized by a single perception output and a single corresponding safety score. A single vehicle in front of the ego vehicle with a very basic background is a very simple scenario and the resulting image is depicted in Fig. 2. For this scenario a single parameter, namely the distance between the vehicle and the ego vehicle, is varied. The implementation is performed in CarMaker and the image is exported and passed to the perception system which is the SuT.

FIGURE 2. Image of simple example scenario.

C. SAFETY SCORE

For the evaluation of the output from the SuT, this work defines a safety score. This safety score is a metric which acts as a proxy to quantify safety. It is chosen to reflect the quality of the perception which implicitly assumes a correlation for the safety for the downstream driving task. This correlation has already been established in literature on macro level [33]. The most common object detection metric is average precision (AP) [34], [35] which was introduced by the Pascal visual object classes (VOC) challenge [36]. It has since been applied in different variants across various detection benchmarks in the context of driving [37], [38], [39]. Pascal VOC relies on ranking detections by their confidence and assigning true positives and false negatives based on intersection over union (IoU) [36]. This assignment of ground truth and predictions is generally required for perception evaluation with IoU being a commonly applied metric [40], [41].

As noted in previous sections, the interpolation accuracy depends on the smoothness of the underlying function. One option is to apply the AP across multiple images as performed in [18]. However, any such grouping of multiple images introduces a hyperparameter. Additionally, it does not fully

address the issue that the confidence score thresholds introduce discontinuities in the final score. Therefore, the safety score is modified from existing perception metrics in this work while maintaining the common practice of considering both confidence scores and IoU for the evaluation. More specifically, this work relies on the safety score s obtained by multiplying the confidence score c and IoU. It should be noted that similar ideas are applied for loss functions by [42] and [43].

$$s = c \cdot \text{IoU} \tag{3}$$

This score is maximized across detections for the association procedure to suppress low quality detections. The detection with the highest score is associated and the same value is used as the final value of the safety score. While this procedure does not account for false positives, this is inconsequential for the given scenario where a vehicle is always present.

D. GRID REFINEMENT AND ERROR ESTIMATION

The methods explained in section II-D are detailed or modified as follows.

For simplicity, a uniform grid defined by the grid size h is used. The grid refinement is always undertaken with a grid refinement factor of $r = 2$. All calculations are based on the results of the finest grid. Any points missing on the coarser grid are obtained by a simple linear interpolation with second order accuracy $p = 2$. The observed order of accuracy is evaluated for all points individually as shown in section IV.

As the results show, the GCI cannot be used due to instabilities in the observed order p. In addition, the empirical safety factor F_s is unknown while normalization is unnecessary since the safety score is already normalized between 0 and 1. Therefore, the difference between the fine grid and the coarse grid defined as Δs is directly observed instead. The grid refinement is terminated according to the availability of computation power.

IV. RESULTS

This section presents the safety score results over an input parameter range, the convergence of the error and the error itself.

A. SAFETY SCORE RESULTS

Fig. 3 depicts the general results for parameter values between 10-200 m with a spatial resolution of 1 cm. Generally, the results display an unpredictable fluctuation. In addition, some values show outliers which seemingly occur at random.

Fig. 4 shows detailed results at higher spatial resolution for a limited parameter range. The range was arbitrarily chosen to be 50-51 m, but qualitatively similar results are obtained for other value ranges. The same general trend of unpredictable fluctuations for the safety score is observed. For distance changes of 1 mm the deviations are approximately 3%. Notably, the fluctuation is high even in regions where the performance in terms of safety score is high.

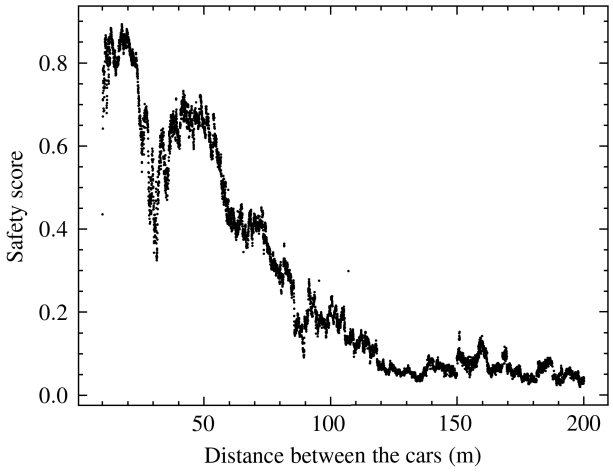

FIGURE 3. Safety score results at 1 cm resolution for 10-200 m show fluctuations.

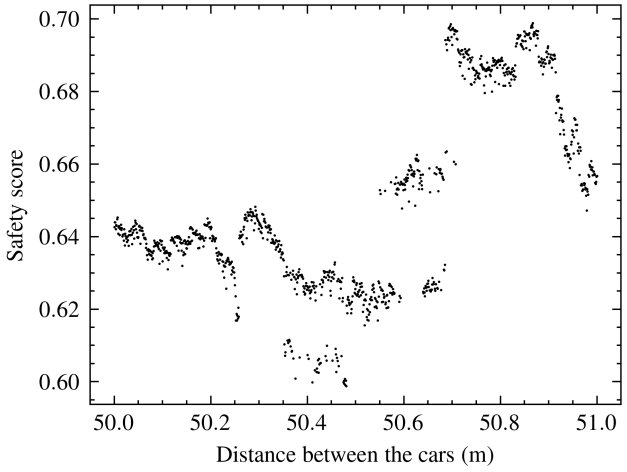

FIGURE 4. Safety score result fluctuations persist at 1 mm resolution between 50-51 m.

B. ERROR CONVERGENCE

As discussed in the section II-D, the approach is to consider the development of the error across multiple grids. Fig. 5 shows the cumulative density function (CDF) of the observed order of convergence calculated for each grid point. The number of the grid in the legend corresponds to the grid iteration, each with a refinement factor of two. Only a selected number of grids is displayed in the image for brevity, but similar results are obtained for the other grids.

The values are always calculated by comparing the interpolation on the coarse grid with the results from the finer grid. Since the theoretical order of the linear interpolation is second order, convergence would be achieved if the observed order matches the theoretical order. However, the spread of the CDF shows that no convergence is achieved. Notably, the mean is closer to zero than to the expected theoretical order of two. While it is theoretically possible that even finer

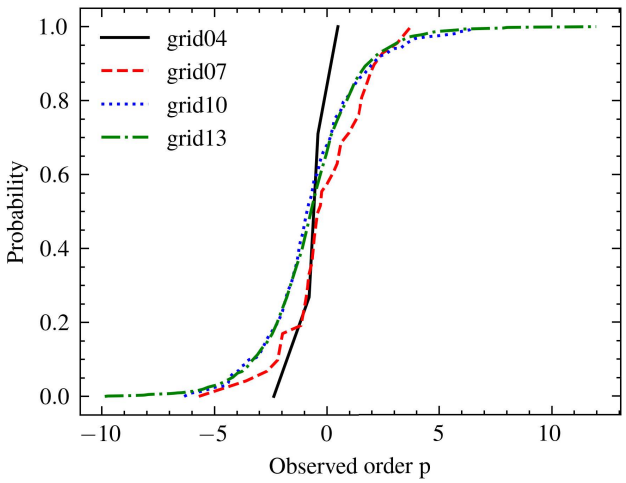

FIGURE 5. The CDF of the observed order *p* for the error does not converge.

grids may lead to different results, no such simulations are conducted, because the advantage of the simulation-based testing approach is lost.

C. ERROR VALUES

This section will now directly show the errors that can be observed for different grids as shown in Fig. 6 Each curve represents a CDF of the safety score error between a given interpolation and the refined grid for all points of the finer grid.

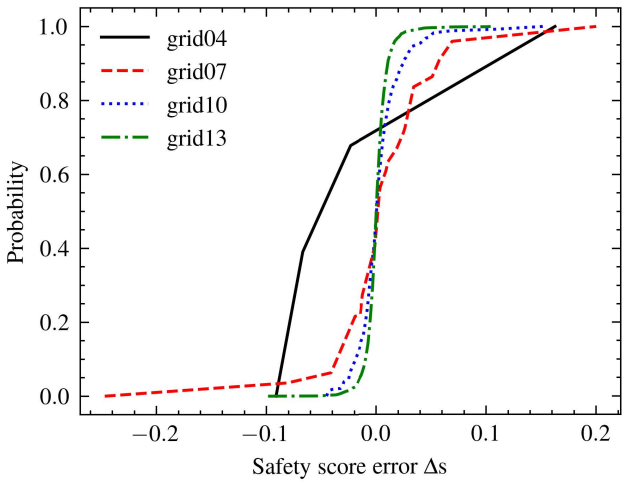

FIGURE 6. The CDF of the safety score error Δs shows improved average error with finer grids but outliers remain.

The first observation is that the finer grids have a steeper gradient in the middle. Basically, this means that the average error for all points does indeed decrease with a finer discretization. However, the graph still shows significant outliers which do not improve reliably as the grid is refined. The outliers show errors of approximately 10% even for the finest grid. This aligns with the results showing the lack of error convergence.

V. DISCUSSION

This section will first consider the reasons for the fluctuations and outliers observed in section IV-A. One explanation is randomness in the components used in the toolchain. To rule out this possibility, verification experiments are conducted for both the simulation environment and the perception component. While the detector shows deterministic behavior, the simulation results output as image by CarMaker are not entirely deterministic. However, the effect of the non-deterministic changes in the input image on the perception safety score is around 10^{-4} and thus insufficient to explain the large variation observed in Fig. 3. Since the detector is pretrained on real data a performance drop induced by the domain gap between the training data and the simulated evaluation data is likely [13], [44]. Nevertheless, an overall drop in performance does not explain the local fluctuation observed for the quality of the perception output. Additionally, the fluctuations are observed across the whole parameter range including regions where the performance is high. Therefore, the large variation in output for small changes in input indicates a volatile behavior of the perception component.

The metric chosen in this work explicitly includes the confidence score output by the detector. It is possible to argue that a metric which does not include the confidence score may show more stable behavior. However, the detection task fundamentally aims at predicting the class-specific likelihood, thus distinguishing the target objects from the background [12], [45]. Moreover, common metrics such as AP [34], postprocessing operations such as filtering detections and non-maximum supression [46] or tracking [47] all rely on the confidence score. We argue that this widespread use of the confidence score as well as the very definition of the detection task being the estimation of a likelihood justify the use as performed in this work. In addition, the use of AP appears to yield similar fluctuations [18].

It is theoretically possible that a finer discretization may achieve different results. However, the current results were already obtained at mm scale with increasing computational requirements for each refinement step. It is noteworthy that these fluctuations occur at scales which are one order of magnitude smaller than the positional accuracy which is verifiable in practical validation experiments [48]. Therefore, even if a validation were theoretically possible with finer discretizations it may not be possible to apply this method in practice with existing experimental setups.

Overall, the results indicate that the observed behavior is indeed due to the inherent attributes of the SuT rather than due to artifacts in the toolchain or the metric. Most importantly, no convergence of the error is observed with fluctuations and outliers persisting even for fine discretizations. Preliminary verification experiments indicate that this fundamental behavior remains the same even for other scenario parameters

even though the magnitude of the error and its fluctuations differ. This severe lack of robustness is observed even for the simplest of scenarios. However, these results also show that the presented method of observing the error convergence can identify lack of robustness in a given SuT.

A remaining question is whether this result is restricted to the model used as SuT in this work. Since only one pretrained detector is studied in this work, no definitive conclusion regarding transfer to other model architectures can be drawn. However, it is possible to consider observations from literature regarding robustness of DNN in general. In the context of adversarial examples, good transfer between different architectures has already been demonstrated empirically [9], [30]. More generally, the lack of robustness to either targeted or common perturbations across architectures is acknowledged in [13]. Additionally, one other work obtains similar results to this work with considerable fluctuation of the perception performance [18]. The results there are obtained with the detector Mask-RCNN [49], which differs in architecture from the detector used in this work. This indicates that the results obtained in this work are not unique to the specific detection architecture. When combined, these findings suggest that the results from this work may also transfer to other tasks and architectures.

Notably, while the ResNet [50] backbone used by Mask-RCNN relies on the ReLU function, the YOLOv5 network [31] applied in this work uses the smooth SiLU function. This means that a smooth activation function is insufficient to produce bounded output errors.

VI. CONCLUSION

This work studies the effect of parameter discretization as part of the scenario-based testing approach for AD functions. Using a common perception algorithm for camera images, a simulation case study is conducted for a simple scenario. The results show strong fluctuations and outliers of the SuT which persist across grid discretizations. Notably, the error does not converge, meaning that no parametrization was found which could be shown to be sufficient for the validation of the perception component. This also means that the method presented in this work is suitable to identify any lack of robustness in the SuT.

Evidence from literature suggests results from this work may transfer to other perception tasks and architectures. However, further investigation is required to obtain conclusive empirical evidence.

Nevertheless, the results of this work cast doubt on the current scenario-based validation process when sampling from a parameter range. Since discretization and interpolation are always part of this approach, it is unclear if applying it to the safety validation of existing DNN components is possible. One possibly complimentary approach may be optimization based approaches or sampling from a distribution [6]. However, it is currently unclear if any combination of these methods is feasible. We hope that this paper motivates further research on this aspect of scenario discretization which has so far mostly been neglected in literature.

ACKNOWLEDGMENT

The authors would like to thank the Open Access Publishing Fund of Technical University of Darmstadt.

REFERENCES

[1] H. Winner, S. Hakuli, and G. Wolf, *Handbuch Fahrerassistenzsysteme*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds. Wiesbaden, Germany: Springer, 2015, pp. 1167–1186.

[2] J. Zhou, R. Schmied, A. Sandalek, H. Kokal, and L. D. Re, "A framework for virtual testing of ADAS," *SAE Int. J. Passenger Cars-Electron. Electr. Syst.*, vol. 9, no. 1, pp. 66–73, Apr. 2016.

[3] R. Lattarulo, J. Pérez, and M. Dendaluce, "A complete framework for developing and testing automated driving controllers," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 258–263, Jul. 2017.

[4] D. Nalic, A. Pandurevic, A. Eichberger, and B. Rogic, "Design and implementation of a co-simulation framework for testing of automated driving systems," *Sustainability*, vol. 12, no. 24, p. 10476, Dec. 2020.

[5] T. Menzel, G. Bagschik, L. Isensee, A. Schomburg, and M. Maurer, "From functional to logical scenarios: Detailing a keyword-based scenario description for execution in a simulation environment," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 2383–2390.

[6] C. Neurohr, L. Westhofen, M. Butz, M. H. Bollmann, U. Eberle, and R. Galbas, "Criticality analysis for the verification and validation of automated vehicles," *IEEE Access*, vol. 9, pp. 18016–18041, 2021.

[7] S. Riedmaier, T. Ponn, D. Ludwig, B. Schick, and F. Diermeyer, "Survey on scenario-based safety assessment of automated vehicles," *IEEE Access*, vol. 8, pp. 87456–87477, 2020.

[8] S. Jesenski, J. E. Stellet, F. Schiegg, and J. M. Zollner, "Generation of scenes in intersections for the validation of highly automated driving functions," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 502–509.

[9] S. Houben et al., "Inspect, understand, overcome: A survey of practical methods for ai safety," in *Deep Neural Networks and Data for Automated Driving*, T. Fingscheidt, H. Gottschalk, and S. Houben, Eds. Cham, Switzerland: Springer, 2022, pp. 3–78.

[10] Y. Huang and Y. Chen, "Autonomous driving with deep learning: A survey of state-of-art technologies," 2020, *arXiv:2006.06091*.

[11] D. Feng, C. Haase-Schütz, L. Rosenbaum, H. Hertlein, C. Gläser, F. Timm, W. Wiesbeck, and K. Dietmayer, "Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 1341–1360, Mar. 2021.

[12] X. Wu, D. Sahoo, and S. C. Hoi, "Recent advances in deep learning for object detection," *Neurocomputing*, vol. 396, pp. 39–64, Jul. 2020.

[13] O. Willers, S. Sudholt, S. Raafatnia, and S. Abrecht, "Safety concerns and mitigation approaches regarding the use of deep learning in safety-critical perception tasks," in *Computer Safety, Reliability, and Security* (Lecture Notes in Computer Science), vol. 12235, A. Casimiro, Ed. Cham, Switzerland: Springer, 2020, pp. 336–350.

[14] *Safety and Artificial Intelligence*, Standards ISO/TC 22/SC 32/WG 14, 1999.

[15] T. Ponn, D. Fratzke, C. Gnandt, and M. Lienkamp, "Towards certification of autonomous driving: Systematic test case generation for a comprehensive but economically-feasible assessment of lane keeping assist algorithms," in *Proc. 5th Int. Conf. Vehicle Technol. Intell. Transp. Syst.*, 2019, pp. 333–342.

[16] L. Huang, Q. Xia, F. Xie, H.-L. Xiu, and H. Shu, "Study on the test scenarios of level 2 automated vehicles," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 49–54.

[17] E. Rocklage, H. Kraft, A. Karatas, and J. Seewig, "Automated scenario generation for regression testing of autonomous vehicles," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 476–483.

[18] J. Bernhard, T. Schulik, M. Schutera, and E. Sax, "Adaptive test case selection for DNN-based perception functions," in *Proc. IEEE Int. Symp. Syst. Eng. (ISSE)*, Sep. 2021, pp. 1–7.

[19] C. Gladisch, C. Heinzemann, M. Herrmann, and M. Woehrle, "Leveraging combinatorial testing for safety-critical computer vision datasets," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 324–325.
[20] B. Kim, A. Jarandikar, J. Shum, S. Shiraishi, and M. Yamaura, "The SMT-based automatic road network generation in vehicle simulation environment," in *Proc. 13th Int. Conf. Embedded Softw.*, Oct. 2016, pp. 1–10.
[21] W. L. Oberkampf and C. J. Roy, *Verification and Validation in Scientific Computing*. Cambridge, U.K.: Cambridge Univ. Press, 2010.
[22] S. Riedmaier, B. Danquah, B. Schick, and F. Diermeyer, "Unified framework and survey for model verification, validation and uncertainty quantification," *Arch. Comput. Methods Eng.*, vol. 28, no. 4, pp. 2655–2688, Jun. 2021.
[23] W. Dahmen and A. Reusken, *Numerik Für Ingenieure Und Naturwissenschaftler*. Berlin, Germany: Springer, 2008.
[24] A. Meister, N. Henze, F. Hettlich, M. Brokate, G. Schranz-Kirlinger, and T. Sonar, "Interpolation—Splines und mehr," in *Grundwissen Mathematikstudium*, M. Brokate, N. Henze, F. Hettlich, A. Meister, G. Schranz-Kirlinger, and T. Sonar, Eds. Berlin, Germany: Springer, 2016, pp. 397–437.
[25] M. Mehrabi, A. Tchamkerten, and M. I. Yousefi, "Bounds on the approximation power of feedforward neural networks," 2018, *arXiv:1806.11416*.
[26] D. Gopinath, H. Converse, C. Pasareanu, and A. Taly, "Property inference for deep neural networks," in *Proc. 34th IEEE/ACM Int. Conf. Automated Softw. Eng. (ASE)*, Nov. 2019, pp. 797–809.
[27] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions," 2017, *arXiv:1710.05941*.
[28] D. Misra, "Mish: A self regularized non-monotonic activation function," 2019, *arXiv:1908.08681*.
[29] Y. LeCun, K. Kavukcuoglu, and C. Farabet, "Convolutional networks and applications in vision," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2010, pp. 253–256.
[30] A. Serban, E. Poll, and J. Visser, "Adversarial examples on object recognition," *ACM Comput. Surveys*, vol. 53, no. 3, pp. 1–38, May 2021.
[31] G. Jocher, A. Chaurasia, J. Borovec, A. Stoken, NanoCode012, Y. Kwon, T. Xie, J. Fang, and imyhxy. (2020). *Yolov5*. [Online]. Available: https://github.com/ultralytics/yolov5
[32] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," *Neural Netw.*, vol. 107, pp. 3–11, Nov. 2018.
[33] Y. Guo, H. Caesar, O. Beijbom, J. Philion, and S. Fidler, "The efficacy of neural planning metrics: A meta-analysis of PKL on nuScenes," 2020, *arXiv:2010.09350*.
[34] K. Oksuz, B. C. Cam, S. Kalkan, and E. Akbas, "One metric to measure them all: Localisation recall precision (LRP) for evaluating visual detection tasks," 2020, *arXiv:2011.10772*.
[35] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 261–318, 2020.
[36] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Sep. 2009.
[37] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.
[38] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "NuScenes: A multimodal dataset for autonomous driving," 2019, *arXiv:1903.11027*.
[39] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, and V. Vasudevan, "Scalability in perception for autonomous driving: Waymo open dataset," 2019, *arXiv:1912.04838*.
[40] M. Hoss, M. Scholtes, and L. Eckstein, "A review of testing object-based environment perception for safe automated driving," *Automot. Innov.*, vol. 5, no. 3, pp. 223–250, Aug. 2022.
[41] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 658–666.
[42] T. Huang, Z. Liu, X. Chen, and X. Bai, "EPNet: Enhancing point features with image semantics for 3D object detection," in *Proc. ECCV*, 2020, pp. 35–52.
[43] Y. Hu, Z. Ding, R. Ge, W. Shao, L. Huang, K. Li, and Q. Liu, "AFDetV2: Rethinking the necessity of the second stage for object detection from point clouds," 2021, *arXiv:2112.09205*.
[44] S. Zhao, X. Yue, S. Zhang, B. Li, H. Zhao, B. Wu, R. Krishna, J. E. Gonzalez, A. L. Sangiovanni-Vincentelli, S. A. Seshia, and K. Keutzer, "A review of single-source deep unsupervised visual domain adaptation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 2, pp. 473–493, Feb. 2022.
[45] X. Zhou, V. Koltun, and P. Krähenbuhl, "Probabilistic two-stage detection," 2021, *arXiv:2103.07461*.
[46] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-NMS - improving object detection with one line of code," 2017, *arXiv:1704.04503*.
[47] Z. Pang, Z. Li, and N. Wang, "SimpleTrack: Understanding and rethinking 3D multi-object tracking," 2021, *arXiv:2111.09621*.
[48] M. Holder, L. Elster, and H. Winner, "Digitalize the twin: A method for calibration of reference data for transfer real-world test drives into simulation," *Energies*, vol. 15, no. 3, p. 989, Jan. 2022.
[49] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.
[50] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015, *arXiv:1512.03385*.

KEN T. MORI received the B.S. and M.S. degrees in mechanical and process engineering from the Technical University of Darmstadt, Hesse, Germany, in 2020.

He is currently working as a Research Assistant with the Institute of Automotive Engineering, Technical University of Darmstadt, since 2020.

XU LIANG received the B.S. degree in automotive engineering from the Technical University of Wuhan, Hubei, China, in 2017. He is currently pursuing the M.S. degree in mechanical and process engineering with the Technical University of Darmstadt.

LUKAS ELSTER received the B.S. and M.S. degrees in mechanical and process engineering from the Technical University of Darmstadt, in 2020.

Since October 2020, he is with the Institute of Automotive Engineering, Technical University of Darmstadt as a Research Assistant.

STEVEN PETERS was born in 1987. He received the Ph.D. degree (Dr.-Ing.), in 2013 from the Karlsruhe Institute of Technology, Karlsruhe, Baden-Württemberg, Germany.

From 2016 to 2022, he worked as a Manager of artificial intelligence research at Mercedes-Benz AG, Germany. He is currently a Full Professor with the Technical University of Darmstadt, Darmstadt, Germany. He has been the Head of the Department of Mechanical Engineering, Institute of Automotive Engineering, since 2022.