Virtual Simulation Based Assessment of ADAS in Consumer Tests by openPASS

Felix Fahrenkrog
Manel Hammouda
Florian Fischer
BMW AG
Germany

Ludwig Maier
Hochschule München University of Applied Sciences
Germany

Paper Number 23-0191

ABSTRACT

Test of consumer protection organizations like the New Car Assessment Programmes (NCAPs) play an important role in the overall safety of modern vehicles. Being focused on passive safety over the past decades, the importance of active safety systems has grown in recent times more and more. To assess the performance of active safety systems, standardized test scenarios which are supposed to represent real world accidents are used today. The constantly increasing requirements and the goal of ensuring the robustness of those active safety systems lead to a vast amount of test scenarios. This trend is accompanied with the aim of testing more complex scenarios. In the future, it will hardly be possible to cover this amount of test by track tests alone. Therefore, new virtual methods to support the assessments are required. This paper aims to discuss the question: What are the requirements for these virtual methods to be implemented on manufacture and consumer rating organization side? To discuss the posed question, a two-stage process is foreseen. In the first step, an exemplary virtual assessment of safety oriented ADAS is conducted. For this purpose, consumer rating test scenarios are set-up within the simulation software openPASS. After the implementation, an assessment for one vehicle and one active safety function is conducted in this virtual environment. Finally, the difference between simulation and real vehicle tests is analyzed. In the second step, the learnings and findings from this study will be used to discuss the requirements for future virtual assessments. The demonstration study in openPASS will cover only an exemplary set of test scenarios. Furthermore, the study will only be conducted for one vehicle. The generalization of the study's findings needs to be investigated further.

INTRODUCTION

Today's vehicles are equipped with different Advanced driver assistance systems (ADAS), ranging from short-intervening autonomous braking systems (AEB) to continuously operating systems like Adaptive Cruise Control (ACC) or Lane Centering Assist (LCA). ADAS play a major role in the overall vehicle safety. To be able to compare different vehicles regarding their overall safety and to push the overall development of those systems, customer protection organizations like the European New Car Assessment Program (EuroNCAP) were established [1].

To be able to compare the different systems, it is necessary to use standardized test methods and scenarios. These test scenarios are developed by the individual customer protection organizations, mainly by looking into recent accident statistics. To assess the performance of the ADAS, real vehicle tests are conducted on test tracks with the help of surrogate collision targets like pedestrian, cyclist or vehicle dummies. Although it is tried to design the tests as realistic as possible, the tests cannot replicate the much more complex reality on public roads to the full extent.

In contrast to passive safety, the fast development of ADAS technology leads to regular updates for the testing procedure. This resulted in an increase of the number of test scenarios as well as of the requirements (see ***Figure 1***). Furthermore, customer protection organizations are becoming more and more focused on the topic of robustness [1], which lead to additional test scenarios where parameters like number of objects, collision point, and daytime are varied to evaluate the systems real world performance. This development further increases the number of tests as well. This triggers the question, whether traditional track testing will be sufficient in the future. In 2015 BMW already proposed a virtual testing approach combing different test tools for comprehensive real-world assessment of ADAS [2].

Euro NCAP initiated a working group to investigate the future test [3]. One promising testing approach is to apply virtual simulation to complete the picture of real-world tests. Virtual test can be applied easily in complex scenario and allow for testing much quicker than on a test track (Reference). On the hand there is always the question about validity of virtual simulation [5].

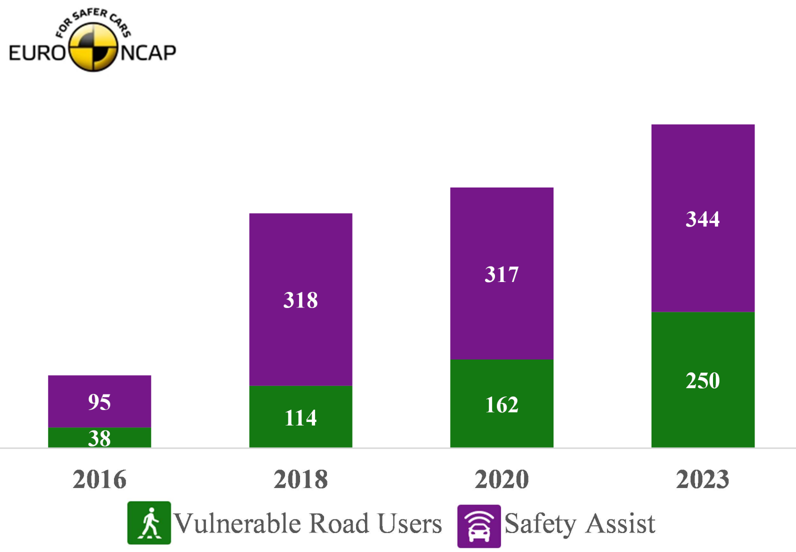

Figure 1. Increase of Euro NCAP test in the last years (based on [5]-[12])

This paper aims contribute to the general discussion about virtual testing of ADAS by addressing the following questions: What are the requirements for these virtual methods to be implemented on manufacture and consumer rating organization side? To approach this question first it is shown, how Euro NCAP scenarios for an effective virtual assessment can be implemented in the open-source simulation tool openPASS [13]. Afterwards, it is reported on the study that compared virtual tests with outcome of the test track test for one AEB function. This exemplary study is the initial point to discuss the future requirements for virtual testing. In the last part requirement for the virtual testing are discussed.

SIMULATION TOOL OPENPASS

Background

OpenPASS is an open-source software [13], which is being continuously developed by the Eclipse openPASS Working Group (WG). The openPASS WG manages the sim@openpass project under the roof of the Eclipse Foundation. It is the driving force behind related development of the simulation platform and its modules. The goal is to ensure a transparent and publicly available simulation framework for the assessment of safety systems. The idea of openPASS started with P.E.A.R.S. [14] in 2014. Back then inquiries within the P.E.A.R.S. group showed that many partners used own developed tools and that there was no specialist tool for the assessment of the safety performance of ADAS. Some partners – namely BMW, Mercedes and Volkswagen – want to address this lack of a common appropriated simulation by initiating an own software tool, which should cover the different baseline approaches of P.E.A.R.S. and which should allow also for a transparent virtual assessment. This led to the founding of the openPASS project in 2016.

openPASS covers two main use-cases, namely the PCM (PreCrash Matrix) crash re-simulation and scenario-based simulations. The PCM crash database is a subset of the GIDAS (German In-Depth Accident Study), in which real world crashes are reconstruct through on-site measurement and evidence. The PCM crash cases describe the trajectory of the traffic participants, and the road setting seconds before the crash. openPASS allows the re-

simulation of these crashes under the consideration of cars equipped with ADAS. In this simulation use case it is assessed whether the tested technology would avoid the crash in question.
The traffic/scenario-based simulation is indeed the most common use-case of openPASS since it offers more opportunities to investigate a larger scenario space and much more scenarios than the crashes re-simulation. This approach includes the stochastic variation of those scenarios, surrounding traffic as well as the intervention through detection of events and triggered actions and much more. Typically, this approach relies on comparison of the results in the baseline (situation without the technology in question) and the treatment condition (situation with the technology). For the comparison first a basic conflict scenario is described. This is done by means of defining the starting conditions of the ego-vehicle (e.g., velocity position), the potential conflicting partner (e.g., velocity, relative distance to the ego-vehicle, and trajectory of the maneuver to be executed). The surrounding traffic is stochastically varied in its position, driver characteristics and speed by openPASS. Then the scenario is run multiple times under variation of the starting conditions and the maneuver of the conflicting partner until a solid statement regarding the influence of the technology on the safety performance can be derived.
Thus, openPASS is capable to covering all by P.E.A.R.S. [14] and in the ISO21934 [15] named baseline approach for the prospective safety performance assessment by virtual simulation. However, openPASS can also be applied in other use cases. One example is the comparison of different technology-wise solutions in the early development stage. If it is applied in other use case adaptation of the used models might be necessary.
One of the most important characteristics of openPASS is the flexibility that it offers through the modular architecture. In fact, the simulation platform allows the connection of models as well as scenarios and maps to the simulation by the means of standardized interfaces and standards, as depicted in **Figure 2**.

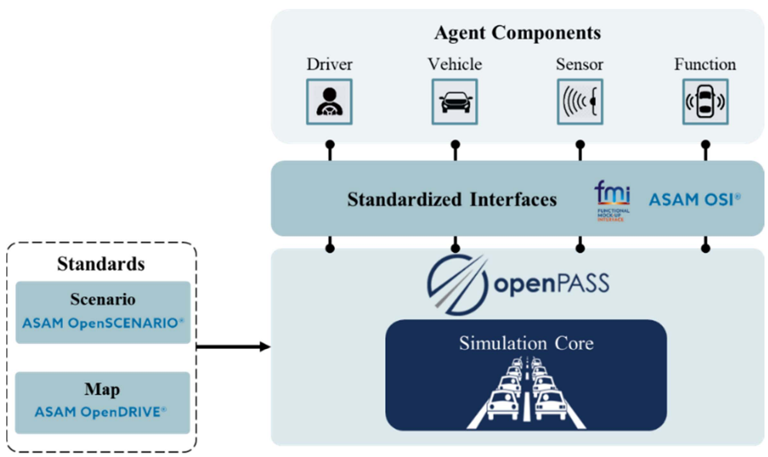

Figure 2: A set of supported standards and standardized interfaces in openPASS

The supported standards OpenSCENARIO [18] and OpenDRIVE [19] allow the description of the scenario setting and the road infrastructure with the possibility to trigger events at certain conditions. Furthermore, the supported standardized Interfaces like OSI (Open Simulation Interface, [16]) and FMI (Functional Mock-up Interface, [17]) enable the coupling of external models. FMI represents a crucial interface in this work. It is indeed an open format specification for exporting and importing simulation models. For instance, a Simulink model can be built as an FMU (Functional Mock-up Unit), which is a packaging format defined by FMI. It basically encapsulates a compiled code (as *.dll or *.so, depending on the platform) and an xml-file that is describes the inputs and outputs of the FMU (modelDescription.xml). This model can be coupled to openPASS using the FMU Wrapper Interface.

Input to openPASS simulation
The simulation tool requires different input files that are described in the following.

- **Scenario** (*.xosc): This file describes the overall situation in terms of the ASAM OpenSCENARIO 1.0 standard [18]. It includes an Init-Tag, which describes the initial setting in the simulation, such as the positions and the velocities of the traffic agents. Additionally, the scenario file contains a Story-Tag, that allows the triggering of certain events at certain conditions e.g., the ego vehicle performs a lane change to the right when the simulation time reaches 10 s. The end simulation time itself, i.e., how long should the simulation run, is also part of the scenario configuration. Furthermore, the scenario contains references to additionally needed files, like the scenery configuration and other relevant catalogs explained in the following.

- **Scenery** (*.xodr): The scenery file describes the road infrastructure following the ASAM OpenDRIVE 1.6 standard [19]. As such, it contains the description of the road network and its geometries e.g., the numbers of the lanes, the curvature of the road and the lane markings. Moreover, the scenery defines the traffic signs and traffic lights as well as any static object or obstacles that may occupy the road.
- **ProfilesCatalog** (*.xml): This catalog is the probabilistic heart of the simulation as it describes the composition of various simulation components and entities, using both, deterministic and stochastic definitions. It defines the composition of the traffic agents that may be vehicles (from the VehicleModelCatalog.xosc) or pedestrians (from the PedestrianModelCatalog.xosc) and their underlying components like sensors, driver assistance systems or driver models. The name ProfilesCatalog comes from the profiles idea that lies behind this file. A profile is indeed a template that defines how an agent is configured in terms of driver models, sensors, and assistance systems, each provided with a certain occurrence probability. As OpenSCENARIO does not support this level of probabilistic variations probabilities (yet), this file is not compliant to the standard.
- **VehicleCatalog** and **PedestrianCatalog**: (*.xosc): These catalogs follow the OpenSCENARIO standard and describe the physical parameters of available vehicles or pedestrians, respectively.
- **SimulationConfig** (simulationConfig.xml): This is the entry point for the simulation, containing the setup of the core, such as active observers, reference to the scenario, the initial random seed, and the number of invocations. Furthermore, the used spawner libraries are referenced in this file. The spawner is one the main core module in openPASS, that allows the spawning of traffic agents during pre-runtime and runtime following certain parameters distributions, such as velocities and time gaps. These parameters are defined in the ProfilesCatalog, explained above.
- **SystemConfigBlueprint** (systemConfigBlueprint.xml): This file consists of a superset of all possible components and their valid connections. Such components can be lateral and longitudinal controllers, assistance systems, prioritizes, driver models, and so on. Depending on the configured profiles and their probabilities, the core picks a subset of components to create one complete system. This file should only be edited by experienced users with a deep understanding of the framework architecture.

Output of openPASS simulation

Outputs are generated by individual observers, configured in the SimulationConfig, and collected within the folder results. This section describes the output files by the Observation_Log, as configured by the provided example configurations.

- **Simulation Output** (simulationOutput.xml): This file acts as a central entry point for further evaluations, such as the visualization. It contains central information about all executed invocations within an experiment, such as the executed scenario and the run results, which can be seen as current values from the random sampling of the given probabilities. As such, each run result contains, a list of participating moving entities (also referred to as agents), events related to the entities, such as collisions or activation of ADAS's, and a reference to the cyclics file. This file does not contain information about the actual position and movements of the different agents.
- **Cyclic Output** (Cyclics_Run_###.csv): This file contains the ground truth information for each agent at each time step. For each invocation, a new file is generated (starting with Cyclics_Run_000.csv) and referenced in the according run results in the simulationOutput.xml.

Next to the output files provided for each simulation runs, there is the option to visualize the output in a separate application, namely the opVisualizer.exe. It represents a 3D visualization of the simulations results, as shown in **Figure 5**; and allows an offline navigation throughout the simulation time and space. Additional to a 3D animation of the simulation results, the opVisualizer offers the representation of occurring events, such as collisions or ADAS warnings or interventions, and displays the sensor ranges.

TOOLCHAIN FOR VIRUTAL TESTING OF EUROPE NCAP SCENARIOS IN OPENPASS

Task of the toolchain

The general aim of the ENCAP-openPASS-Toolchain is to support the concept evaluation of scenarios from the Euro NCAP in the part of the active safety functions for AEB (Automated Emergency Braking)- and LSS (Lateral Support System). The toolchain aims to assess the system performance in the concept phase of function – a particular focus is on assessing the vehicle's sensor setup. For this purpose, an idealized sensor model is used. The tool verifies whether an existing sensor setup can cover the Euro NCAP scenarios [5][9]. In addition, by means of the tool also the functions logic can be assessed to provide a first prediction about its potential performance. The ENCAP-openPASS-Tool uses openPASS software. In this sense the tool is an addon tool to openPASS. The main task of this tool is to generate and define trajectories from the published protocols of Euro NCAP which leads to the defined collision point between the ego-vehicle and the target object as specified in the protocol [Quelle]. The tool also generates all the other required input files for the openPASS simulation. Finally, the tool, which is implemented in Python [Quelle], can trigger openPASS simulation and evaluate the outcome of the openPASS simulation.

Definition of scenario simulation parameters

To create and simulate the AEB or LSS-scenarios it is necessary to set the parameters for the simulation. There are different scenario specific parameters like the curvature of the trajectory, velocities or collision points at the front of the ego. All these parameters are defined in testing protocols of the Euro NCAP. Another scenario parameter that needs to be defined individually is to create a specific ego profile with its corresponding sensor setup. The ego vehicle is considered as a 2D-boudingbox which is created by the tool. The input values for that are the length, width, the position of the axes and the position of the center position. These dimensions are required for the simulation with openPASS.

To ensure the right collision point when defining the trajectory, the ego-vehicle's width has been divided in constant segments (e.g., every 5% of the vehicle's width) with potential collision points. The user can set the collision point as a percentage value. However, the default setting with a step-size of 5% allows the calculation of all Euro NCAP scenarios without any loss of accuracy. The sensor setup of the ego is idealized like it is shown in the introduction of openPASS. One sensor is described by the x- and y-position, its heading angle, the field of view (FOV) and the detection range. These parameters must be given in the ego-vehicle's coordinate system. For every simulation a specific ego profile can be implemented with one or more sensors. The targets are given as a catalog since they have fix dimensions according to the Euro NCAP protocols.

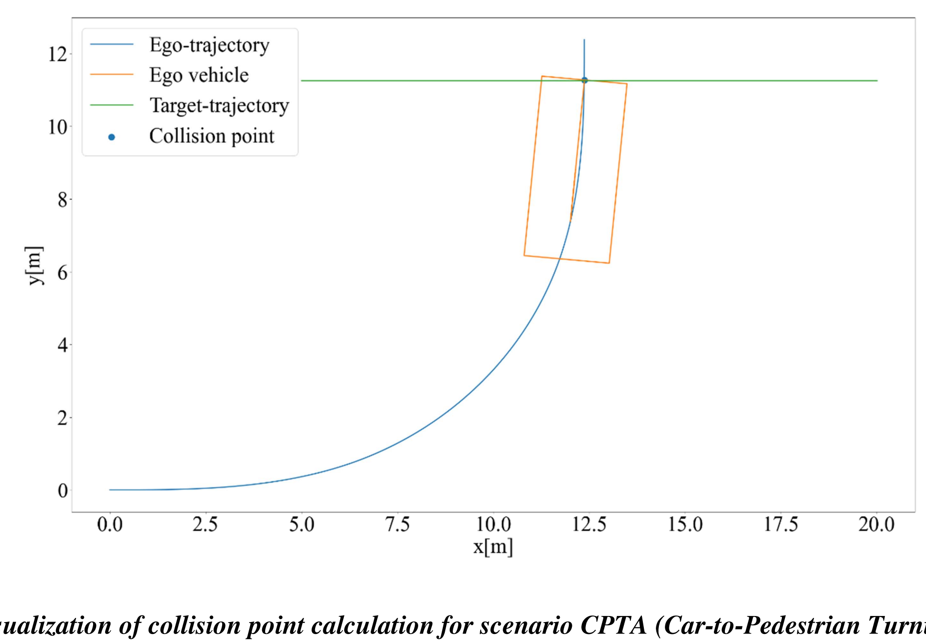

Figure 3: Visualization of collision point calculation for scenario CPTA (Car-to-Pedestrian Turning Adult).

Collision point calculation

To get a defined collision point between the ego and the target, it is important to know the collision point at the front of the ego. This value is given by the protocols or can be adjusted by the user. The second step is to create the ego trajectories according to the specifications of the Euro NCAP protocols. For instance, in case of a turning scenario, it is necessary to know the curvature in advance. These trajectories are created dependent on the velocity of the agents. The explanations from the protocols were deployed in the ENCAP-openPASS-tool for an automated creation. After the calculation of the ego trajectory the ego profile is positioned at all points of the trajectory stepwise. This procedure is repeated as often as the collision point at the front of the ego profile reaches a specific distance according to the given protocol. This is shown in the **Figure 3**. If the specific distance is reached, the target trajectory can be created in a reverse calculation starting from the end point. The length of the target trajectory depends on the simulation time.

Sequences in the simulation tool

The first step is to create the input files where the calculated trajectories are also implemented. This has been described above. In the next step, the openPASS is started. The simulation is triggered for all defined scenarios. The execution of the simulation of the different simulation runs automatically. Before the execution, the user can decide whether to include an already implemented active safety function as a FMU or not. The user can change the settings of the scenario in a GUI (Graphical User Interface), which is depicted in **Figure 4**.

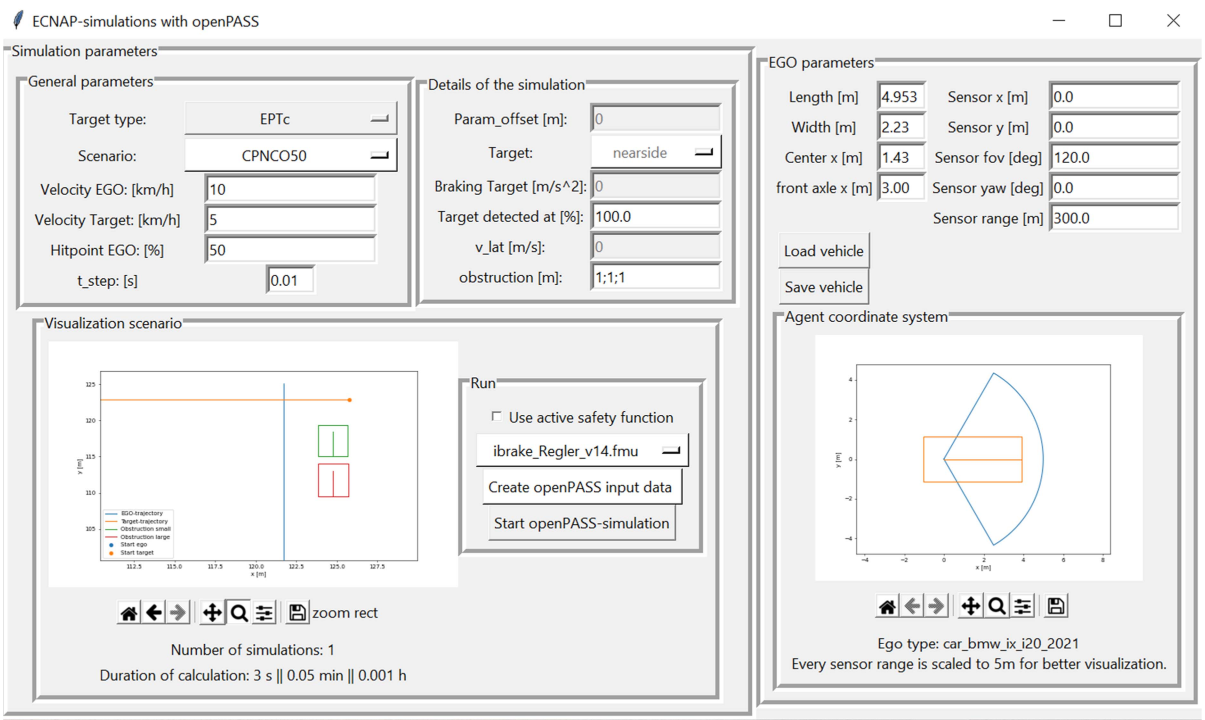

Figure 4: NCAP tool GUI interface for openPASS simulation.

The figure shows all the different changeable boundary conditions of one specific scenario on the left-hand side. The right-hand side encapsulates all the parameters which can be configured for the ego vehicle. For better explanation, the GUI visualizes the planned scenario and the ego vehicle as a plot in the bottom left part of the tool.

Evaluation

For each time step, several output variables can be analyzed, such as the velocity, the acceleration, the position of the agents and the visible and detected objects by the implemented sensor setup. The simulation verifies an agent as visible, if one point of the 2D-boundingbox lies within the field of view of the idealized 2D sensor setup. If an agent is detected, it is necessary that at least a minimum percentage of visible area is in the field of view of the sensor. This value can be setup by the user in the configuration files. The sensor model needs a minimum visible area of an object to correctly classify it, e.g., as a bicyclist. Indeed, this information can be found in the results files of openPASS. These trace files can be visualized in a 3D-visualization toll, namely the opVizualizer. An example of a visualized Euro NCAP scenario is shown in **Figure 5**. The red ego vehicle (0) and the bicyclist (1) are the relevant collision agents. Their trajectories are synchronized to meet at the calculated collision point at the front of the ego by the absence of an active safety function. The agents (2) and (3) simulate the necessary obstructions of the scenario. This timestep shows the moment when the bicyclist is detected from one example sensor setup.

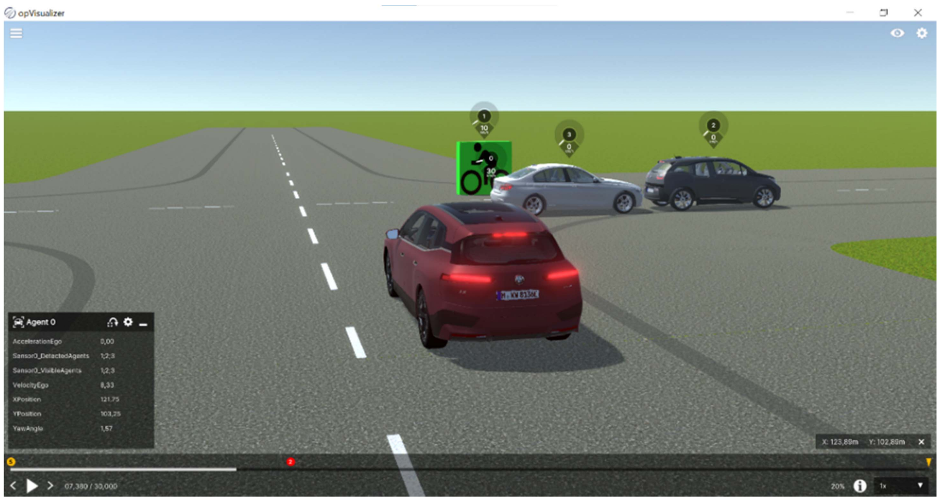

Figure 5: 3D visualization of openPASS simulation in the CBNAO (Car-to-Bicyclists Near Side Adult Obstructed) crossing scenario (ego vehicle: red vehicle).

COMPARISON VIRTUAL TEST AND TEST TRACK

The major purpose of this paper is to raise awareness about the urgent need to define the requirements of virtual testing for consumer ratings. To approach this question, an exemplary comparison of simulation results and real test tracks using some KPIs (Key Performance Indicators) are presented in this section. Therefore, rear-end Euro NCAP conflict scenarios are simulated in openPASS. The method and tools presented in the sections “Simulation tool openPASS” and “Toolchain for Virutal testing of europe NCAP Scenarios in openPASS” are applied to evaluate a concept model of an AEB (Automated Emergency Braking) function. It is important to note that the model that is deployed in this study does not fully represent the real function and may include some differences for the sake of simplification. The AEB model is implemented as a Simulink model, which is later built as a FMU following the FMI Standard. This kind of standardized encapsulation of models allows their co-simulation in different platforms that support FMI. In this case, the AEB FMU is connected to openPASS through an FMU-Wrapper. For the end-user, the FMU connection in openPASS consists of an xml-configuration of the inputs and outputs.
In this study, a total of 22 scenarios are implemented. These scenarios can be divided in 3 categories:

- CCRb (Car to Car Rear braking): These scenarios involve an Ego and a Target vehicle with a rear-end conflict, where the target travels with the same velocity as the Ego vehicle and then carries out a braking maneuver. The deceleration varies between -2 m/s^2 and -6 m/s^2. Here only a 100% overlapping is considered. The velocities of the two cars lay by 50 km/h and the distance between the vehicles is 12 m or 40 m. This leads to exactly 4 combinations.
- CCRm (Car to Car Rear moving): These scenarios involve an Ego and a Target vehicle with a rear-end conflict, where the target travels with the constant velocity of 20 km/h. The Ego velocity covers the interval [30,70] km/h with a 5 km/h step. With only 100% overlapping, a total of 9 scenarios is obtained.
- CCRs (Car to Car Rear standing) scenarios: These scenarios involve an Ego and a Target vehicle with a rear-end conflict, where the target is standing. The ego velocity varies between 10 km/h and 50 km/h with a 5 km/h step. This results as well in exactly 9 scenarios.

The above Euro NCAP scenarios are carried out during test tracks, where a real AEB function is tested. Indeed, the results of two real test runs following the same settings are available. To have a basis for a comparison between the test tracks results and the obtained simulation traces, multiple KPIs are computed. These KPIs include continuous variables courses over time, e.g., acceleration, distances, velocities, as well as discrete indicators, such as the trigger TTC (Time To Collision), the final minimal distance of Ego and the collision result. The TTC is the remaining time before a collision occurs between ego-vehicle and target object if none of the vehicles performs an evasive action. Therefore, it is the time needed to travel the net gap distance with the relative speed between the leading and the following agent. In this context, the trigger TTC represents the TTC where the AEB function starts the braking maneuver. It is indeed an indicator about the reaction rapidity of the function.

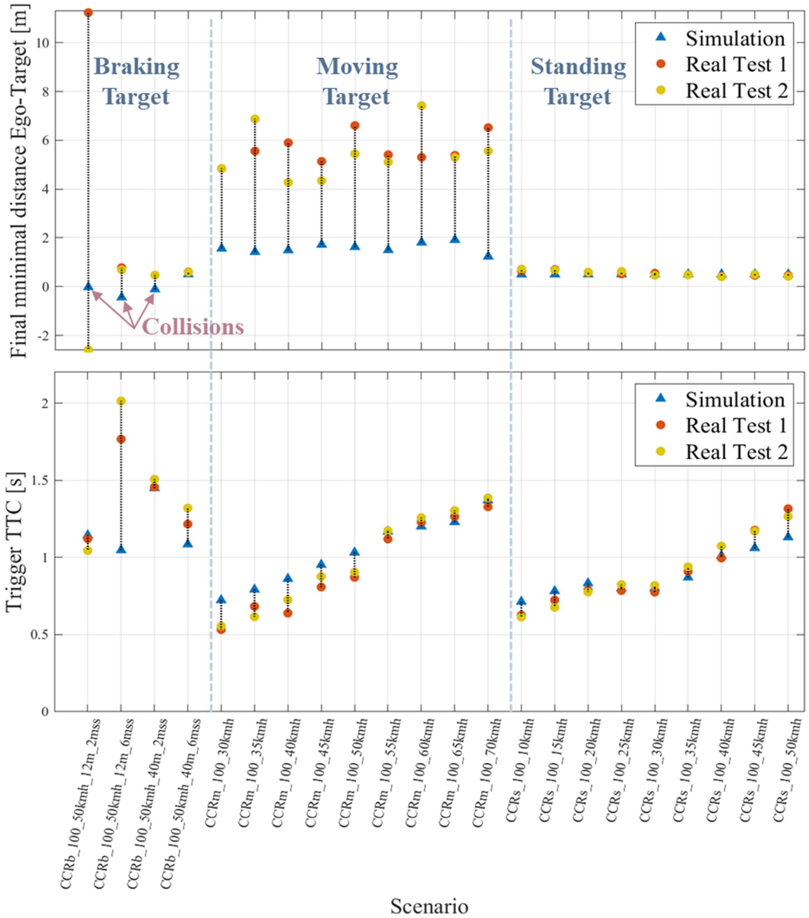

Figure 6: Comparison of the trigger TTC and the final minimal distance btw. Ego and Target in openPASS and in two test tracks.

Figure 6 presents the overall results of the comparison between the simulations and the two test tracks in terms of trigger TTC and the final minimal position of the Ego vehicle, i.e., the minimal net distance that separate the ego-vehicle from the target vehicle. A key indicator for evaluation is indeed the collision status. This is shown in the figure implicitly. A negative net distance between the two vehicles indicates the occurrence of a collision. As clearly depicted in **Figure 6**, three simulation runs involving a braking target scenario resulted in a collision. These include critical settings with an initial gap distance between the vehicles that is equal to 12 m additional to one scenario with 40 m gap distance. On the one hand, this may be explained by the relatively late TTC trigger in the simulation, especially for the second CCRb scenario, where the trigger TTC lies by 1 s in the simulation in contrast to both test tracks that lie around 1.8 s and 2 s (see **Figure 6**). Although the trigger TTC does not show a considerable difference compared to the other test tracks, the deceleration development over time may be the main reason for the collisions in the simulation. For the real test tracks, only the second recording marked in yellow of the first CCRb scenario resulted in a collision. All other cases, the real AEB function managed to avoid the crash. For the moving target scenarios (CCRm), the results in the simulation and in the real world align in terms of crash occurrence. Whereas the final net gap distance between Ego and Target lies around 2m in the simulation, the test recording show a certain scattering of the values 8 and 4 m. Clearly, even the real tests show variations themselves and do not completely overlap on many levels. The trigger TTC shows relatively similar values in both cases. Finally, the runs including a standing target (CCRs) demonstrate harmonized values in the final standing position around 0.5 m. The trigger TTC shows an increasing trend with the increase of the ego-vehicle's velocity similar to the moving target scenarios, however with lower values due to the larger difference velocities that would consequently lead to a faster reaction.
Figure *7* and ***Figure 8*** presents the time courses of two important signals, namely the ego-vehicle's acceleration and the net distance between ego-vehicle and the target-vehicle. Here, two exemplary scenarios are presented. The upper

one shows the CCRs scenario with an ego-vehicles velocity of 25 km/h, whereas the lower plots represent the results of a CCRb scenario with a net initial distance of 40 m and a target deceleration of -6 m/s^2. Both cases the system reacted to the given thread and avoided a collision (positive net distance in the end). The net distance signals show in this case a good alignment between the simulation and the real test tracks results. The acceleration courses in the CCRs scenario triggers approximately at the same time as in the real tests, whereas the CCRb scenario shows a slightly delayed reaction that does not affect the overall result. Nevertheless, the simulation results evolve in a rather discrete and smooth manner compared to the real test results, which show a more continuous evolution over time as well as some oscillations due to the real vehicle dynamics.

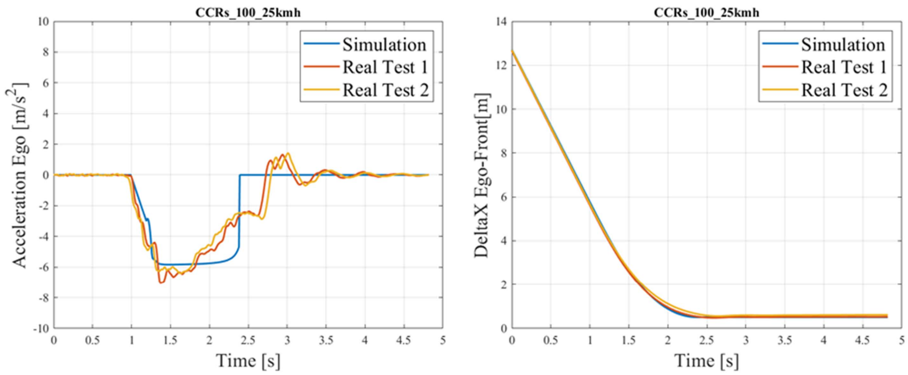

Figure 7: Comparison between simulation (openPASS) and two test tracks tests for acceleration and delta positions in CCRb.

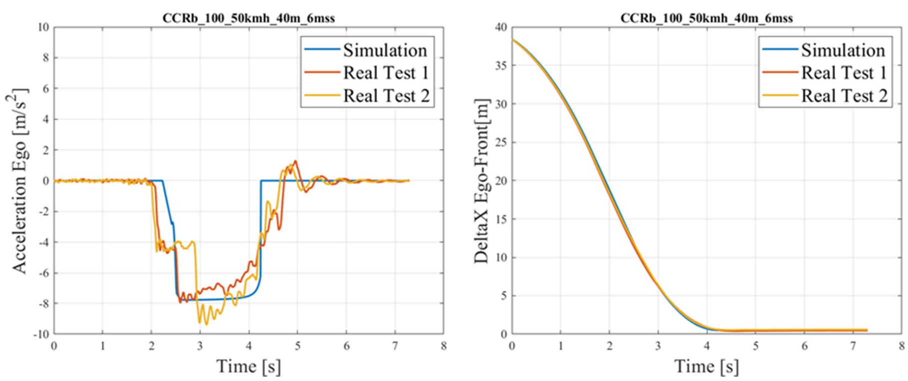

Figure 8: Comparison between simulation (openPASS) and two test tracks tests for acceleration and delta positions in CCRs.

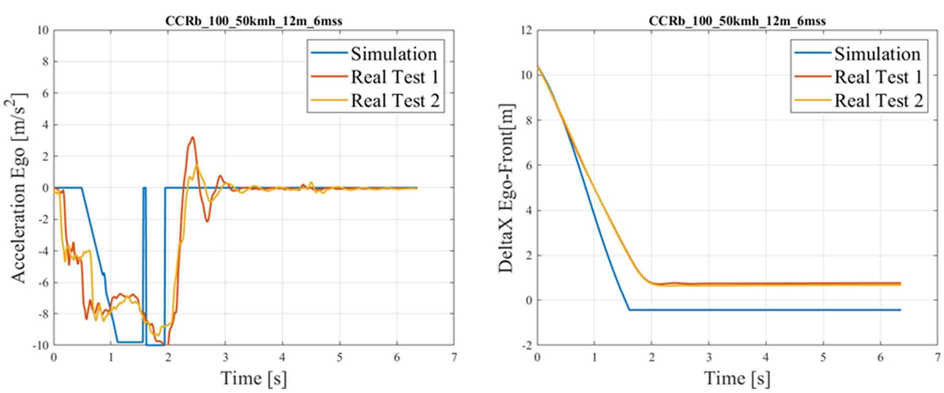

Figure 9: Comparison between simulation (openPASS) and two test tracks tests for acceleration and delta positions in CCRb.

Figure 9 corresponds to the CCRb scenario with a net initial distance of 12 m and a target deceleration of -6 m/s^2. This case includes a collision, as mentioned before. The longitudinal acceleration signal shows indeed a delayed activation of the braking maneuver as well as a slower development of the deceleration over time. This leads to a crash in the simulation, while in the test track test the collision was avoided. Further investigation in the reasons for the different behavior are planned. However, the example should indicate that different scenario poses different challenges for the simulation.

To conclude, this exemplary study showed that the simulation results for most of the scenarios align with the test track results. The deviations observed in the challenging CCRb scenarios may be explained by the simplified AEB function model in the simulation combined with the idealized object-based sensor models deployed in the simulation. Nevertheless, this study represents an adequate discussion basis for the needed requirements towards a widely accepted virtual testing approach for consumer ratings.

FUTURE REQUIREMENTS FOR VIRUTAL TESTING

The exemplary studies in the previous section give an indication what can be achieved by simulation and where challenges start to arise. It is obvious that there is potential to optimize the results further. Nevertheless, the learnings from this study should be used to approach the initial questions: What are the requirements for these virtual methods to be implemented on manufacture and consumer rating organization side?

The first aspect is to clarify the scope of the assessment. There are multiple approaches and tools to simulate the performance of ADAS and ADS to assess their safety performance. This paper provides only an example implementation. It is also clear that the different tools have different advantages and disadvantages. Certainly, there is not one simulation solution that fits all problems respectively. And even if such a solution would exist, it would come with compromises in terms of performance. Therefore, the first step before defining any requirements is to clearly identify the purpose and scope of the virtual assessment. Once this is clarified, the requirements and the simulation solution can be chosen.

1. **What do I want to assess exactly with the simulation?**
   This is the key question to be answered. This question should not be answered by a general statement, like e.g. I want to assess an AEB virtually. It is rather about the details, like do I want to focus on the perception, logic or actuator? What are the scenarios I am interested in, and which metrics do I intend to use? Do I need a quick answer or a very detailed answer? The answers to these questions will already point to the required solution for the virtual assessment. Depending on the considered use case, the solutions may differ. One example for defining the evaluation scope is given be by the P.E.A.R.S. initiative. It defined five aspects (metric to be used, technology, scenario, region for prediction) to be covered by each research question in the domain of prospective safety performance assessment (see [20]). These aspects will not hold true for each evaluation use case if the scope of the evaluation varies. But they provide hints about what should be specified before defining the further virtual assessment.
2. **Which accuracy of the simulation is required for the intended results?**
   The second question is equally important for defining the virtual assessment approach as the first one. Even if the virtual assessment use case is clearly described there are often multiple technical solution in terms of fidelity of models and simulation execution (e.g., simulation step size). It often requires a trade-off between accuracy and simulation effort. In general, it can be assumed that models with a higher fidelity will require also more computation time. Therefore, it is crucial to decide on the required accuracy. If the user is just interested in knowing whether a collision is avoided, potentially less detailed and faster models can be applied compared to question to determine the stopping distance in cm (e.g., vehicle model could be represented by a point mass model and not multi-body-system model). In the context, it must be understood that best model is not necessarily the most detailed one, but rather the fastest model that delivers the results in the asked accuracy.
3. **How can I generate trust and acceptance for my simulation result?**
   The biggest question of virtual assessment is always: does the simulation represent the real world? First, it needs to be recognized that a simulation is not the real world. Thus, there will be always a difference. The same would apply for a test on a test track. In this case the environment and surrounding traffic is represented by physical models (e.g., balloon dummy cars such as [21], artificial building such as described in [22]). However, these tests are widely accepted as representative for the real world. It is rather the question how large / how small this difference is and whether the difference in the evaluation use case is

acceptable. The answers to these questions are not straightforward. Therefore, it should be considered from the beginning what can be done to generate trust and acceptance for the simulation results. There are indeed different options. Examples are using transparent open-source approach (see openPASS above), a comprehensive V&V process for the virtual assessment tools, documentation, and publications. Whether these steps or the combination of these steps are sufficient will also depend on the stakeholder of the results.

4. **Which existing standards can be applied?**
   This question addresses again the simulation approach. The use of existing standards whenever applicable is recommended for different reasons. It will help to generate trust for the simulation results. It makes it easier to cooperated with third parties. Furthermore, it allows the exchange and usage of different models. In this context, the ASAM standards OpenSCENARIO [18] and OpenDRIVE [19] already represent a solid start basis. They allow to have a common definition of the executed scenario and the simulated road. Clearly, standards are only useful if they are applied by several different organizations. Thus, not only the organization applying the standards benefits from them, but also the standards themselves.
5. **Which aspects need to be harmonized respectively which not?**
   The last question addresses post assessment phase. It deals with the learnings of the study and how future assessment can benefit from theses learnings. It is quite likely that issues that one person encountered in his/her work might also be relevant for other. And as stated earlier standard help to generate trust for virtual assessment. At this stage more exchange between the different stakeholders is required. The German founded research project “Set Level [23] is here a good example for such activities leading to new standards (see OSI activities in Set Level [24]).

The final simulation result is an interplay between the simulation tool, the applied model – in particular vehicle, sensor and technology model – as well as the parametrization of these models. Different solutions and simulation approaches might lead depending on the simulation scope to sufficient accurate results. From a development perspective of a simulation tool, it is effort- and efficiency-wise reasonable to accept minor deviation in case the result is still accurate enough for the simulation use case. At this point it must be considered that the also real-world tests differ slight in their outcome. Therefore, it would be questionable to require from one tool a 100% precise answer while today test tools can also not provide a 100% precise answer.
Therefore, it is in the authors opinion not reasonable to set requirements for above mention simulation models (vehicle, sensors, and technology model) or tools. In the authors opinion, the definition of requirements should focus instead rather of the following aspects

- **Scenario format:** A clear description of the scenario to be tested is essential if standardized tests should be executed in a simulation. This description shall be delivered in standardized format to guarantee a consistent scenario interpretation as well exchangeability between different tools.
- **Interface of the simulation**: Standardized interface enable the possibility to exchange models between simulation tools. This becomes highly relevant if certain standardized models (e.g., environmental model) should be used during an assessment.
- **Models not related to the technology or vehicle**: This aspect is closely linked to the previous one. Today in test track test standardized objects are used to ensure similar conditions for everyone. The pendant to this in the virtual assessment are standardized models. However, to gain the maximum use out of the virtual environment larger variation can be considered. Thus, it is necessary to not only discuss the models but also the variation of these models.
- **Definition of metrics**: This aspect sounds obvious. However, the work of P.E.A.R.S. has showed that there could be easily different interpretation of one metric [25]. Therefore, a clear description of the evaluation criteria is required to ensure to ensure a consistent and harmonized calculation throughout different evaluations.
- **Required accuracy of simulation**: This is the core aspect of virtual assessment and should be one of the first aspects to be discussed for the virtual assessment, since it implicitly sets the requirements for the technical solution of the virtual assessment. Therefore, the discussion among the stakeholders shall focus on this aspect also to avoid misunderstandings.
- **Documentation and expected validation & verification activities**: Another vital part of the discussion between the different stakeholders is the documentation of the virtual assessment’s results and the expected V&V activities to be reported. The documentation has certainly a close link to the assessment metric. A standardization of the documentation format will help in the comparison of different results. The same

applies to the V&V process, which is of importance to demonstrate the correctness of the virtual results and to increase their acceptance.

CONCLUSIONS

Consumer ratings are an important tool to communicate and promote the value of safety oriented ADAS. They also allow to compare the performance of different safety systems from individual manufacturers. In the past this has been done by means of test track test. However, the increasing number of systems as well as the extension of the test spaces also requires the consideration of other test tools. Thus, virtual assessment approaches will play a major role in this area in the future. This paper will contribute to this topic in different ways. Starting with the simulation tool openPASS, the ENCAP toolchain have been implemented that allows to set up and asses a system in virtual Euro NCAP tests. The toolchain has been used to run a comparison between real-world test on a test track and simulation for one implementation. The assessed exemplary system was an AEB. For some scenarios the results of both environments are in good accordance, while the scenarios with higher dynamics showed some differences. It needs to be noted here that the function model and the vehicle model were not optimized for this assessment. Thus, it can be expected that improves even with the same simulation setup could be achieved.
This exemplary study allowed to investigate the difference between both test environment deeper. This led to the definition of relevant question to be answered in case a virtual assessment should be set up. Furthermore, six important aspects are described which should be focused on in the discussion consumer ratings, namely scenario format, interfaces for the simulation, non-technology models, metric, required accuracy and documentation including validation and verification of the simulation. To find appropriated solution the discussion should involve all relevant stakeholders.

REFERENCES

[1] Euro NCAP. 2022. "Euro NCAP Vision 2030 - a safer future for mobility." Euro NCAP road map.
[2] Helmer, T. 2016. Safety Performance Assessment of Assisted and automated driving. Presentation at SAE Government Industry meeting 2016, Washington D.C.
[3] Euro NCAP 2017. "Euro NCAP 2025 Roadmap – in pursuit of vision zero". Euro NCAP road map.
[4] Vergnaud, J., Piperno, A., Herveleu, F. 2020. „Validation Method: Virtual Testing", Virtual Testing AEBS, UNECE AEBS Meeting, https://wiki.unece.org/download/attachments/101554586/AEBS-12-07%20%28UTAC%29%20Virtual%20testing%20AEBS.pdf?api=v2
[5] Euro NCAP. 2019. "Test Protocol – AEB Car-to-Car systems", Test protocol Version 3.0.2.
[6] Euro NCAP. 2017. "Assessment Protocol – Safety assist", Test protocol Version 8.0.2.
[7] Euro NCAP. 2015. "Assessment Protocol – Safety assist", Test protocol Version 6.1.
[8] Euro NCAP. 2022. "Vulnerable Road User Testing Protocol", Test protocol Version 9.0.1.
[9] Euro NCAP. 2020. "Test Protocol – AEB VRU systems", Test protocol Version 3.0.3.
[10] Euro NCAP. 2019. "Test Protocol – AEB VRU systems", Test protocol Version 2.0.4.
[11] Euro NCAP. 2017. "Test Protocol – AEB VRU systems", Test protocol Version 2.0.2.
[12] Euro NCAP. 2015. "Test Protocol – AEB VRU systems", Test protocol Version 1.0.1.
[13] openPASS. 2022. "openPASS Working Group" website: https://openpass.eclipse.org/, accessed 18th November 2022.
[14] P.E.A.R.S. 2021. "Prospective effectiveness Assessment for road safety". Presentation at 12th UNECE VMAD Sub Group 2 Virtual Testing meeting, https://wiki.unece.org/display/trans/VMAD-12th+SG2+session
[15] ISO. 2021."Road vehicles — Prospective safety performance assessment of pre-crash technology by virtual simulation — Part 1: State-of-the-art and general method overview", ISO Standard 21934-1, https://www.iso.org/standard/76497.html
[16] ASAM. 2022. "OSI - Open Simulation Interface", https://www.asam.net/standards/detail/osi/, accessed 4th November 2022
[17] Modelica Association Project. 2014. "Functional Mock-up Interface for Model Exchange and Co-Simulation", https://fmi-standard.org/assets/releases/FMI_for_ModelExchange_and_CoSimulation_v2.0.pdf , July 25, 2014
[18] ASAM. 2022. "OpenSCENARIO". V1.2.0. https://www.asam.net/standards/detail/openscenario/, accessed 4th November 2022

[19] ASAM. 2020 "ASAM OpenDRIVE V1.6.0 User Guide". V1.6.0 https://www.asam.net/index.php-?eID=dumpFile&t=f&f=3495&token=56b15ffd9dfe23ad8f759523c806fc1f1a90a0e8, accessed 4th November 2022

[20] Page et al. 2015. "A comprehensive and harmonized method for assessing the effectiveness of advanced driver assistance systems by virtual simulation: the P.E.A.R.S. initiative". 24th ESV Conference. Gothenburg, Sweden

[21] Euro NCAP. 2018. "Global Vehicle Target Specification". Technical Bulletin, Volume 1.0, https://cdn.euroncap.com/media/39159/tb-025-global-vehicle-target-specification-for-euro-ncap-v10.pdf

[22] Sayer, J.R. 2018. "Connected/Automated Vehicle and Infrastructure Research". Final Report- SPR 1695, Research Project No. OR14- 053, University of Michigan Transportation Research Institute, Ann Arbor, https://www.michigan.gov/MDOT/-/media/Project/Websites/MDOT/Programs/Research-Administration/Final-Reports/SPR-1695-Report.pdf

[23] Köster, F. and Rude, S. 2021. "SET Level Mid-term Event Introduction and Challenges". Presentation at Set level mid-term event. https://setlevel.de/assets/mid-term-presentation/Introduction.pdf

[24] Mai, P. R., Aue, R., Friedmann, L., Habedank, C., Rosenberger, P., Sachsenweger H. 2022. "Industrial Standards for Development of Automated Driving" Presentation at Set level final event. https://setlevel.de/assets/s2v2-industrial-standards.pdf

[25] Wimmer, P.; Düring, M.; Chajmowicz, H.; Granum, F.; King, J.; Kolk, H.; Op den Camp, O.; Scognamiglio, P.; Wagner, M. 2019. "Toward harmonizing prospective effectiveness assessment for road safety: Comparing tools in standard test case simulations", Traffic Injury Prevention, Volume 20 p. 139-145, https://doi.org/10.1080-/15389588.2019.1616086