How Does Simulation-Based Testing for Self-Driving Cars Match Human Perception?

CHRISTIAN BIRCHLER, Zurich University of Applied Sciences, Switzerland and University of Bern, Switzerland
TANZIL KOMBARABETTU MOHAMMED, University of Zurich, Switzerland
POOJA RANI, University of Zurich, Switzerland
TEODORA NECHITA, Zurich University of Applied Sciences, Switzerland
TIMO KEHRER, University of Bern, Switzerland
SEBASTIANO PANICHELLA, Zurich University of Applied Sciences, Switzerland

Software metrics such as coverage or mutation scores have been investigated for the automated quality assessment of test suites. While traditional tools rely on software metrics, the field of self-driving cars (SDCs) has primarily focused on simulation-based test case generation using quality metrics such as the out-of-bound (OOB) parameter to determine if a test case fails or passes. However, it remains unclear to what extent this quality metric aligns with the human perception of the safety and realism of SDCs. To address this (reality) gap, we conducted an empirical study involving 50 participants to investigate the factors that determine how humans perceive SDC test cases as safe, unsafe, realistic, or unrealistic. To this aim, we developed a framework leveraging virtual reality (VR) technologies, called SDC-Alabaster, to immerse the study participants into the virtual environment of SDC simulators. Our findings indicate that the human assessment of safety and realism of failing/passing test cases can vary based on different factors, such as the test's complexity and the possibility of interacting with the SDC. Especially for the assessment of realism, the participants' age leads to a different perception. This study highlights the need for more research on simulation testing quality metrics and the importance of human perception in evaluating SDCs.

CCS Concepts: • **Software and its engineering → Empirical software validation**.

Additional Key Words and Phrases: Software Testing, Self-driving Cars, Simulation, VR, Human Perception

ACM Reference Format:
Christian Birchler, Tanzil Kombarabettu Mohammed, Pooja Rani, Teodora Nechita, Timo Kehrer, and Sebastiano Panichella. 2024. How Does Simulation-Based Testing for Self-Driving Cars Match Human Perception?. *Proc. ACM Softw. Eng.* 1, FSE, Article 42 (July 2024), 22 pages. https://doi.org/10.1145/3643768

1 INTRODUCTION

In recent years, the development of autonomous systems has impacted our society in many aspects of our life [15, 20]. For instance, humans no longer rely on vacuuming their houses or mowing their grasses; nowadays, we have robots that can do (or will do) our chores [8]. However, specific safety-critical instances of autonomous systems such as unmanned aerial vehicles (UAVs) and

Authors' Contact Information: Christian Birchler, Zurich University of Applied Sciences, Winterthur, Switzerland and University of Bern, Bern, Switzerland, christian.birchler@unibe.ch; Tanzil Kombarabettu Mohammed, University of Zurich, Zurich, Switzerland, tanzil.kombarabettumohammed@uzh.ch; Pooja Rani, University of Zurich, Zurich, Switzerland, rani@ifi.uzh.ch; Teodora Nechita, Zurich University of Applied Sciences, Winterthur, Switzerland, teodora.nechita@zhaw.ch; Timo Kehrer, University of Bern, Bern, Switzerland, timo.kehrer@unibe.ch; Sebastiano Panichella, Zurich University of Applied Sciences, Winterthur, Switzerland, sebastiano.panichella@zhaw.ch.

ACM 2994-970X/2024/7-ART42
https://doi.org/10.1145/3643768

(a) Failing Test: SDC driving off-lane (unsafe).

(b) Passing Test: SDC driving in-lane (safe).

Fig. 1. Examples of simulation-based tests of an SDC.

self-driving cars (SDCs) [36, 37, 61, 63, 65, 75] can experience failures that can harm humans or the environment [29].

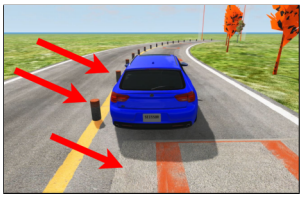
Testing safety-critical autonomous systems is crucial to avoid harmful incidents in real environments [2, 10, 23, 38, 72]. To that end, simulation environments have been widely adopted to test cyber-physical systems (CPS) in general [38, 39, 50], and SDCs in particular [9, 22]. Simulation-based testing is easier to replicate and is more cost-efficient than field testing [31]. Figure 1 illustrates two test cases where an SDC model is deployed in a virtual environment, and the simulated car behaves according to the control algorithms. A test case is said to pass if the car's behavior can be considered safe, while unsafe behavior constitutes a failing test case. Figure 1a shows an unsafe behavior (failing test) as the SDC drives off the lane, while Figure 1b shows a passing test.

Current research on simulation-based test case generation of SDCs relies on an oracle that determines if a system under test is safe or unsafe based on safety metrics [10, 25, 51], particularly the out-of-bound (OOB) metric. The metric is largely adopted for assessing the safety behavior of SDCs [25, 49, 51] and is referred to as a metric related to the lateral position of the SDC [36].

Both test cases illustrated in Figure 1 are classified using the OOB metric [11] and align with the human perception of safety. However, it is yet unclear whether STSG metrics (e.g., OOB) serve as meaningful oracles for assessing the safety behavior of SDCs. For instance, the test cases in Figure 2 are marked pass according to the OOB metric, as the SDC is keeping the lane. On the contrary, from a human standpoint, we can consider the behavior of the SDC hardly as safe. In the first test case using the BeamNG.tech simulator [27] (see Figure 2a), the SDC approaches solid delineators after ignoring a speed bump. Despite maintaining the lane at a speed of 50 km/h, there is a high risk of an accident when classifying this test case as a pass based on the OOB metric. In the second test case using the CARLA simulator [22], shown in Figure 2b, the SDC ignores the red signal. Since the car stays in the lane, it meets the OOB metric, leading to a false passing test case.

Inspecting the OOB metric reveals that it is measured at a single point in time in simulation, which is insufficient to identify unsafe behaviors. For instance, Figure 2a shows the speed bumps on the right lane, and evaluating the SDC at a single point is insufficient to assess its safety over these speed bumps. Unlike real-world speed bumps, which are smooth and rounded, the test bumps have sharp edges that damage the SDC even at reasonable speeds (from a human viewpoint). Similarly, Figure 2b shows another instance where we observe the red light signal, but the SDC ignores it. It is unclear whether the red signal was already there before the SDC drove past it or the signal turned red just after the SDC analyzed the simulation scene. We hypothesize that current simulation-based testing of SDCs does not always align with the human perception of safety [25, 49, 51] and realism [4, 48, 55, 71], which are relevant aspects for an effective assessment. Hence, our primary goal is to understand and characterize this mismatch by answering the following question:

> When and why do safety metrics of simulation-based test cases of SDCs match human perception?



(a) SDC in BeamNG.tech driving with 50 km/h close to obstacles

(b) SDC in CARLA crossing a red signal without stopping

Fig. 2. Examples of unsafe tests with valid OOB criteria

To address the problem of *safety* and *realism* of test cases described in our motivating examples, we conducted an empirical study involving 50 participants using our framework named SDC-Alabaster. The framework employs virtual reality (VR) technologies [60] (i) to immerse humans in virtual SDCs so that they can sense and experience the virtual environment similarly to the real world, and (ii) to enable SDC developers and researchers to analyze the human perception of *safety* and *realism* of SDC test cases. The participants in our study are asked to assess the level of *safety* and *realism* of multiple, diverse simulation-based test cases. Moreover, we provide the participants the possibility to experience simulation-based test cases in which they have the possibility to influence the behavior of (i.e., interact with) the SDC. We experimented with two representative SDC simulators as virtual environments, BeamNG.tech and CARLA, which are widely used in academia and industry.

The paper contributes and complements previous research as follows:

- We propose a methodology implemented in the SDC-Alabaster framework, a VR-based technological approach, to examine how quality metrics align with human perception of safety and realism in simulation-based testing. This is to address the *Reality Gap* [4, 37, 48, 55, 71] problem, a significant concern in simulation-based testing (Section 7);
- We propose the first empirical study that investigates the perception of *realism* and *safety* in SDC test cases of 50 participants using VR technology. We publicly share a replication package with the code to reproduce our results (Section 9);
- We share a first taxonomy of factors influencing the perceived realism of SDC simulators and discuss the confounding factors and implications of our work.

Our results show the impact of using VR in assessing SDCs, highlighting the dynamic nature of safety perception in test cases: *"Safety perception of SDC test cases is not static."* Such results emphasize the importance of human interaction with the vehicle when evaluating SDCs using VR.

The paper covers background (Section 2), study design (Section 3), our framework, experiments, and methodology. Section 4 presents our results, followed by discussions in Section 5 and threats to validity in Section 6. We discuss related work and conclusions in Section 7 and Section 8.

2 BACKGROUND

This section provides a background on existing technologies used in our study. Specifically, we briefly overview the simulators, test generators, and a test runner for SDCs, as well as the VR technology.

2.1 SDC Simulators

To investigate when the safety metrics for SDCs match the human perception, we use two state-of-the-art SDC simulators, namely BeamNG.tech and CARLA. The selection of these simulators is based on two criteria: (i) BeamNG.tech is mainly used in academia and gets more attention in industrial contexts [10, 25, 47, 51], and (ii) CARLA is well-known in industry and academia [22, 30, 76]. We

did not use Udacity, Apollo, SVL, and DeepDrive simulators since their active development has been stopped or they have too long release cycles.

2.1.1 BeamNG.tech. BeamNG.tech is a well-known reference simulator used in recent years in several software engineering studies and SDC testing competitions [10, 11, 25, 28, 51]. The BeamNG.tech simulator comes along with a soft-body physics engine that allows deformations and more realistic crashes and impacting forces on objects.

2.1.2 CARLA. Another widely used simulator in academia and practice is CARLA [22, 30, 76, 78]. The differences between CARLA and BeamNG.tech are twofold. On the one hand, CARLA comes with a rigid-body physics engine that differently than a soft-body physics engine (of BeamNG.tech) does not deform objects; e.g., when a crash happens, the objects remain rigid.

2.2 Test Generators & Test Runner

We use existing test generators to generate test cases automatically for both simulators. We use test generators from the tool competition of the *SBFT* workshop [25, 51] where the actual road in the simulation environments is the result of interpolating the road points that are generated by the test generator. To run test cases in simulation environments, we need a test runner that executes the test cases and reports the test outcomes. For this we use the SDC-Scissor [10] tool since it has implemented a test runner that monitors the OOB metrics, which is suitable for our study.

2.3 Virtual Reality

The notion of VR refers to the immersive experience of users being inside a virtual world. In our study, we want to provide the study participant with an immersive experience of the test cases to have more accurate feedback on their perception of the safety and realism of SDC. We leverage VR headsets and tooling for the simulation environments to achieve this goal.

Headset & VR connection with simulation environments. We use the HTC Vive Pro 2 headset to provide the study participants with a 360° VR experience. The headset connects via wire to an external device with a dedicated GPU for high-resolution VR rendering. Most SDC simulators do not support VR out of the box. This is also the case for BeamNG.tech and CARLA. Therefore, for our study, we use third-party tools to enable the missing VR support for both simulators. For BeamNG.tech, we use vorpX, a specialized tool to transform any visual output to the screen to a compatible input for VR headsets so that it provides an immersive feeling for the user. The vorpX software gives a broader view angle when wearing a VR headset. The user can move the head and explore the virtual environment according to the head movement. In the case of the CARLA simulator, Silvera et al. [60] implemented an extension of CARLA, allowing the simulator to be compatible with the HTC Vive Pro 2 VR headset. When launching the CARLA application, passing the `-VR` flag puts the simulator into VR mode so that can be used with the headset.

3 METHODOLOGY

Using SDC-Alabaster (Section 3.3.2), we conducted an empirical study involving 50 participants (recruiting explained in Section 3.4), with several steps (summarized by Figure 4) devised to collect different types of evidence and data to answer our main question: *When and why do safety metrics of simulation-based test cases of self-driving cars match human perception?* The usage of SDC-Alabaster immerses the study participants in virtual SDCs (Figure 3) by leveraging VR technologies (Section 3.3).

(a) Participant with a VR headset (b) Outside view (c) Driver view

Fig. 3. One of our participants immersed in virtual SDCs with SDC-Alabaster

3.1 Research Questions

We structured our study around three main research questions (RQs), in which the participants were asked to assess the level of safety and realism of multiple simulation-based test cases for SDCs.

3.1.1 RQ_1: Human-based Assessment of Safety. Our first research question focuses on the perception of safety compared to the OOB metric:

> RQ_1: *To what extent does the OOB safety metric for simulation-based test cases of SDCs align with human safety assessment?*

RQ_1 explores participants' perceptions of safety levels for SDC test failures with/without VR technology. We hypothesize that the OOB safety metric may not align with human safety perception. We evaluate alignment through Likert-scale responses from participants, correlating it with test case outcomes (Section 4.1). Statistical tests on experimental and survey data are used to investigate the impact of simulators (BeamNG.tech vs. CARLA), driving views (outside and driver's view), and test case complexity (with/without obstacles/vehicles) on SDC safety perception.

3.1.2 RQ_2: Impact of Human Interaction on the Assessments of SDCs. Once we know how humans perceive the safety of SDC test cases and how this is related to the OOB metric (RQ_1), we investigate whether human-based interactions with the virtual SDC affect the safety perception of the test case. We argue that the safety perception of a SDC can vary when having the ability to interact, i.e., the possibility to accelerate/deaccelerate the vehicle manually, and previous VR research has shown that interactions can influence the environment positively or negatively [33, 34, 46, 53]. This aspects deserves investigation since it can help developers and researchers in designing better test cases and evaluation metrics, which lead us to our second research question:

> RQ_2: *To what extent does the safety assessment of simulation-based SDC test cases vary when humans can interact with the SDC?*

3.1.3 RQ_3: Human-Based Assessment of Realism. We argue that the level of realism of SDC simulation-based test cases is another important factor influencing the safety perception of SDCs. The notion of realism relates to the *Reality Gap* (discussed in Section 7), a critical concern regarding the oracle problem in simulation-based testing: *"due to the different properties of simulated and real contexts, the former may not be a faithful mirroring of the latter"*. While recent studies provide solutions for addressing this problem, e.g., by leveraging domain randomization techniques or using data from real-world observations [17, 37, 40, 77], there is no prior study that studied/characterized the perception of realism of SDC test cases from human participants when using VR technologies [33, 46, 53]. Hence, we complement RQ_1 and RQ_2, by addressing a third research question:

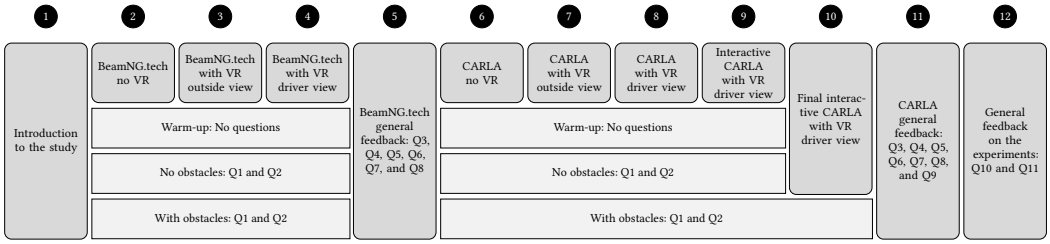


Fig. 4. Design overview with survey question IDs from Table 1

RQ$_3$: *What are the main reality-gap characteristics perceived by humans in SDC test cases?*

Hence, after the experiments for RQ$_1$ and RQ$_2$, we ask the study participants to evaluate the level of realism for BeamNG.tech and CARLA. Then, we develop a taxonomy of aspects influencing these environments' realism to help improve simulation environments for effective testing of SDCs so that different properties of simulated and real contexts are minimized.

3.2 Design Overview

Figure 4 overviews the design of our study involving 12 steps: In step ❶, we welcome and introduce the study participant by explaining the context and the procedure for the experiments. The participant in step ❷ sits before a computer screen and experiences three simulation-based test cases with the BeamNG.tech simulator. While sitting before a computer, the participant wears a VR headset for the next steps. In step ❸, the participant experiences three test cases with the BeamNG.tech simulator observing the SDC from an *outside view* perspective while in step ❹, the participant experiences three test cases with the BeamNG.tech simulator from a *driver view* perspective. The step ❺ focuses on general feedback on the experiments with the BeamNG.tech simulator. Then, the steps ❷,❸,❹ are repeated for the CARLA simulator in ❻, ❼, ❽. In step ❾ with CARLA, the participant, while wearing a VR headset from a driver's view, experiences three test cases in which they can control the SDC speed with a keyboard. In addition to step ❾, one group of participants in step ❿ will experience a crash with the SDC. The step ⓫ focuses on general feedback on the experiments involving CARLA while the step ⓬ focuses on general feedback on the study.

For the steps ❷ - ❹, and ❻ - ❾, the participant experiences three test cases. The first test case is the warm-up so that the participant can familiarize with the simulation environment. The second test case has no obstacles, and the third test case has obstacles (i.e., has higher complexity). At step ❿, the participant only experiences the complex test case with obstacles.

3.3 Design Implementation

We implement our design by conducting experiments with our test runner called SDC-Alabaster. The test runner uses three distinct test cases created by a test generator (see Section 2.2). The participants give responses to our survey questionnaires using *Google Forms*.

3.3.1 Test Cases. We use three test cases generated by the *Frenetic* test generator, the top-ranked state-of-the-art tool in the SBST tool competition [16]. The first test case is the warm-up that lets the participant familiarize with the simulation environment and view setting, e.g., the VR headset and the simulator. Hence, no survey question for this first warm-up test case is provided. The second test case does not have obstacles, while the third involves obstacles (higher complexity).

3.3.2 SDC-Alabaster. We extend the existing test runner SDC-Scissor (see Section 2.2) by implementing SDC-Alabaster (SDC hum**A**n-in-the **L**oop simul**A**tion-**BAS**ed **T**esting s**E**lf-driving ca**R**s).

Table 1. Survey questions with Likert-scale (LS), Open answer (OA), and Single-choice (SC) types

ID	Question	Type
Q1	What is the perceived safety of the Scenario?	LS
Q2	Justify the perceived safety of the Scenario.	OA
Q3	How would you scale the realism of scenarios generated by test cases in the simulator?	LS
Q4	Justify the level of realism of scenarios generated by test cases.	OA
Q5	How would you scale the driving of AI of the simulator?	LS
Q6	Justify the driving of AI from the simulator.	OA
Q7	How would you scale overall experience with the simulator?	LS
Q8	Justify overall experience with the simulator.	OA
Q9	How do you compare safety with and without interaction?	OA
Q10	Did this experiment change the way you thought about the safety of self-driving cars?	SC
Q11	Please write in a few words on your experience and suggestions.	OA

SDC-Alabaster implements an interface to run test cases with the CARLA simulator in steps ❻ - ❿. As for BeamNG.tech, we add obstacles to the test cases in CARLA to achieve similar complexity levels. Additionally, for steps ❾-❿, the participants controlled the SDC speed with the keyboard.

Test cases generated are processed differently between BeamNG.tech and CARLA since CARLA. Automatically generated test cases in BeamNG.tech (Section 2) consist of a sequence of XY-coordinates (i.e., the road points). The CARLA simulator, however, does not have all the road points defined in the test. SDC-Alabaster segments road definitions using start and end points of segments for test cases in CARLA. It facilitates user immersion and safety evaluation by adjusting test cases for CARLA and utilizing VR headsets for proviging an immersive experiences.

3.3.3 Survey Questionnaires. We employ *Google Forms* as a survey tool for our questionnaires. Table 1 summarizes participant questions, having multiple choice (MC), open answer (OA), and Likert scale (LS) questions (with values from 1-5, where 1 for very unsafe, 5 for very safe, and 3 for neutral) to address our research questions (RQs). Participants answered Q1 and Q2 after the second and third test cases, respectively, with the first test case serving as a warm-up without safety assessment. To limit biases, the participants took breaks between sessions. For Q3-Q8, participants provide responses after all three simulator test executions, i.e., at step ❺ for BeamNG.tech and step ⓫ for CARLA. Note that at step ⓫, we include an additional question, Q9, for experiments involving CARLA, which includes interactive scenarios requiring keyboard inputs to control the SDC's speed.

3.3.4 Experimental Setting. We conduct experiments in a dedicated, soundproof room to eliminate external distractions. Participants sit at a table equipped with a desktop computer, laptop, and a VR headset. They use the laptop, running the *Google Forms* application, to complete the survey questionnaires and the desktop computer for non-VR experiments. For VR experiments, participants use the HTC Vive Pro 2 headset powered by a *nVidia GeForce RTX 3080* and *Windows 10* operating system. Additional extensions are employed to allow a full VR experience to participants, such as vorpX for BeamNG.tech's VR support and the *DReyeVR* extension for CARLA, are used. We also integrate SDC-Alabaster to facilitate testing with both BeamNG.tech and CARLA simulators. Participants can interact with specific SDC test cases, adjusting the speed using the keyboard. The duration of the experiments varies between 70 and 90 minutes.

3.4 Study Participants

We recruit participants via email invitations sent to industrial partners, university students, and researchers across departments. We target various mailing lists, including non-computer science organizations, and leverage social media platforms (e.g., Twitter and LinkedIn). We use physical/digital flyers to attract diverse participants, ensuring a broad range of backgrounds and education levels.

3.4.1 Pre-survey. When participants sign up for our experiments, we email them a pre-survey created with *Google Forms* to collect demographic information. This survey includes an introduction to the topic, an overview of the experiment (including expected time and location), and a recommendation to wear contact lenses; it also provides details about the simulator and VR headset used. Furthermore, the pre-survey includes a disclaimer regarding confidentiality and anonymity and a warning about potential VR-related accidents or fatalities that the participants could experience. Following this section, we gather background information on participants, as detailed in the Appendix (appx.) of our replication package (Section 9). These questions cover testing and driving experience, VR technology usage, age, and gender. This additional information helps us investigate potential confounding factors affecting safety and realism perception.

3.5 Data Collection

We gather data from two primary sources: the survey (both pre-experiment and during the experiments) and the simulation logs collected during participant experiments.

3.5.1 Survey Data. For both BeamNG.tech and CARLA simulators, participants evaluate test cases considering the questions reported in Table 1. Specifically, for steps ❷ - ❹ and ❻ - ❾, Likert-scale and text data are collected for each test case except the warm-up case. For step ❿, only Likert-scale and text data are collected for test cases with obstacles. Additionally, at steps ❺ and ⓫, general feedback on the simulators is collected after the test executions with all viewpoints. Complementary, participants rate the perceived safety and realism of each simulator using Likert-scale values based on their own driving experiences. Finally, general feedback on the experiments is collected at step ⓬. In total, we collected 21 Likert-scale, 23 open, and 1 single-choice response per participant during the experiments. In addition to the experimental survey, we gather data from the pre-survey (Section 3.4.1) to obtain participant demographics.

3.5.2 Simulation Data. For each test case in each participant's experiment, we collect relevant data, saving logs (see Section 9) in JSON files of SDC-Alabaster. These logs include timestamped vehicle position coordinates, sensor data (e.g., fuel, gear, wheel speed), and OOB metric violations (i.e., driving off the lane), categorizing the test as pass or fail based on this metric. Additionally, on CARLA, the log structure includes also weather condition details. It is important to note that to enhance our findings further, we also analyze participants' quantitative and qualitative insights both with and without VR headsets, as well as when experiencing different driving views.

3.6 Data Analysis

3.6.1 RQ_1 & RQ_2: Perceived Level of Safety. We utilize various visualizations, including stacked barplots and boxplots, to assess safety and realism perceptions. We apply statistical tests: Wilcoxon rank-sum, and Vargha-Delaney to determine the effective size. For RQ_1, we mainly analyze responses from the test cases where the participant has no interaction with the SDC; for RQ_2, we analyze the data where the participant has some direct interactions with the SDC by a keyboard to control the vehicle's speed. In RQ_2, we explore how SDC interactions affect the safety and realism perceptions of participants. For this, we analyze Likert-scale scores and qualitative feedback. We employ stacked bar plots to examine data spread across the two categories in steps ❽ and ❾.

3.6.2 RQ_3: Taxonomy on Realism. With RQ_3, we examine the realism of SDC test cases and their correlation with human safety assessments. We identify and categorize factors affecting test case realism in a taxonomy based on the participant responses in question Q4 at steps ❺ and ⓫.

We adopt a two-step approach for the initial taxonomy creation. Initially, two authors analyze responses grouped by the simulators; one author focuses on Q4 from step ❺ with the BeamNG.tech

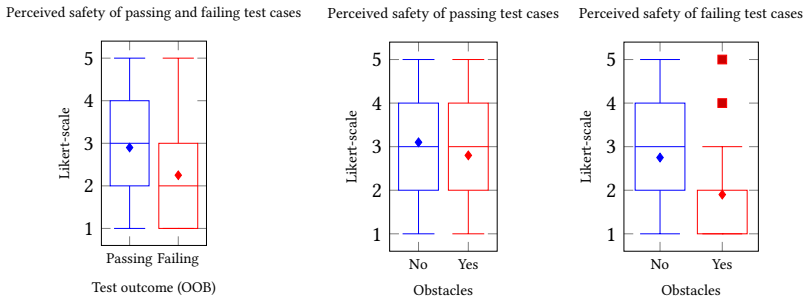


Fig. 5. Perceived safety of failing and passing tests grouped by scenario's complexity

simulator, and the other on Q4 from step ⓫ with the CARLA simulator. Each author proposes categories via an open-card sorting method [62]. In the second step, both authors collaboratively define a meta-taxonomy by discussing their proposed categories. Subsequently, this meta-taxonomy is employed to label all Q4 responses for BeamNG.tech and CARLA (steps ❺ and ⓫). To do this, the two authors designing the meta-taxonomy and a third author conduct a hybrid card sorting labeling process using online spreadsheets. They individually assign each response to the meta-taxonomy categories or create new categories when necessary. A collaborative approach is employed for validation, where each of the three co-authors reviews and addresses any disagreements in assignments during an online meeting.

4 RESULTS

This section presents the results for RQ_1, focusing on participants' safety perception of test cases, and RQ_2, examining how this perception changes when participants can interact with the SDC. For RQ_3, we discuss the taxonomy obtained by classifying participants' comments on test case realism.

4.1 RQ_1: Human-Based Assessment of Safety Metrics

To address RQ_1, we analyzed Likert scale values across various data subgroups. These subgroups included comparisons between test outcomes (failures/successes based on OOB) and different test case complexities (with/without obstacles). This allowed us to identify factors influencing perceived safety among participants. We present boxplots and statistical tests (appx. B.1) for each subgroup.

4.1.1 Safety Perception of Failing vs. Passing Test Cases. Figure 5 illustrates perceived safety distributions for test cases grouped by test outcome (OOB metric). We found a significant difference (appx. B.1) in how participants rate safety for failing and passing test cases on a Likert scale.

> *Finding 1:* The passing test cases (i.e., the cases where the OOB metric is not violated) have a higher perception of safety from the participants than those failing (OOB metric is violated).

The aforementioned Finding 1 is somewhat expected and is aligned with comments from study participants (appx. C.1). These comments pertain to the BeamNG.tech simulator, excluding VR and obstacles. We selected these comments for their exclusive focus on SDC lane-keeping, providing qualitative insights into the OOB metric without obstacle influence. Notably, among comments where the SDC violates the OOB metric (test case failure), safety concerns are recurrent: *"As the car did not drive all the time on the street, I felt unsafe. [...]."- (P3/B1/S1)*"; *"When the car starts to go off the road when driving in a curve, it feels pretty unsafe."- (P31/B1/S1)*"; *"Not Very Safe since the car sometimes drove a bit from the road."- (P45/B1/S1).*

On passing test cases where the OOB metric is not violated, we can find that the participants gave consistent comments in terms of safety: *"The car was driving in lane and at a safe speed considering*

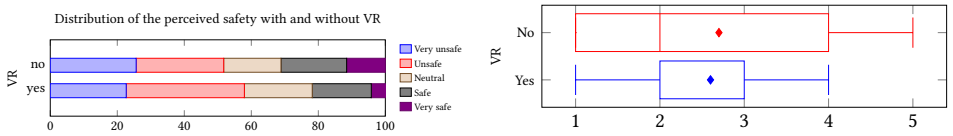


Fig. 6. VR vs. no VR

the road is empty." - (P16/B1/S1); *"The car was following the path in a safe way and was not speeding up too much." - (P25/B1/S1).*

All comments that support Finding 1 are listed in appx. C.1.

4.1.2 Safety Perception with and without Obstacles. Participants assessed test cases with varying complexity, including additional obstacles. Figure 5 displays differences in perceived safety, with statistical significance reported in appx. B.1. Concretely, failing test cases are generally seen as less safe, but those with added obstacles are perceived as even less safe. In contrast to passing test cases, perceived safety remains largely unaffected by an increasing complexity of scenarios (e.g., additional obstacles). As shown in appx. B.1, no significant statistical differences were observed in the samples, leading us to conclude:

> *Finding 2:* There is no statistical difference in safety perception between scenarios with and without obstacles when the OOB metric is not violated. However, when the car goes out of bounds, the scenario is perceived as significantly less safe with obstacles ($p = 3.52 * 10^{-16}$).

From participants, we received qualitative support for Finding 2. For those feeling unsafe with scene obstacles, here are representative answers: *"The car crashed toward an obstacle, and even running over bumps was not so smooth as humans would do. Definitively more unsafe than the previous scenario."- (P1/B1/S2)*; *"Ran off the road in a curve and hit obstacles without slowing down, which resulted in flat tires."- (P24/B1/S2).*

In participants who felt safe or neutral when obstacles were present, consistent comments were reported: *"It car was running smooth with obstacles, there was a moment when it was too close to one of the obstacle" - (P16/B1/S2)*; *"The vehicle does well to avoid obstacles while maintaining the safe speed" - (P18/B1/S2)*; *"The driver accelerated over all the obstacles and did not have a perfect finish." - (P40/B1/S2)*; *"Car was driving well. Only at the end, it went off the road, but there was no object it bumped into." - (P45/B1/S2).*

All comments that support Finding 2 are reported in appx. C.1.

4.1.3 Safety Perception, with VR and without VR. To assess the impact of VR on safety perception, we categorized data into *with VR* and *without VR* groups. Appx. B.1 shows no statistically significant difference. However, Figure 6 reveals that *without VR* had more *very unsafe* and *very unsafe* responses. This is also evident from the smaller interquartile range in *with VR* (compared to the *without VR*).

> *Finding 3:* The utilization of VR had a minor impact on safety perception. However, participants using VR tended to perceive scenarios as somewhat less safe, though this difference was not statistically significant (Wilcoxon rank-sum test, $p = 0.16$).

Certain participant comments support Finding 3. For instance, a neutral participant stated: *"The prespective doesnt change much with the VR" - (P22/B2/S1).* Another example is a comment from a participant who felt very unsafe: *"The same as without the VR glasses. The car was not able to keep the middle of the lane and was driving badly compared to a human." - (P28/B2/S1).*

In Figure 7, we note a decrease in test case safety perception across various viewpoints. Statistical differences are evident in appx. B.1, supporting the following general finding:

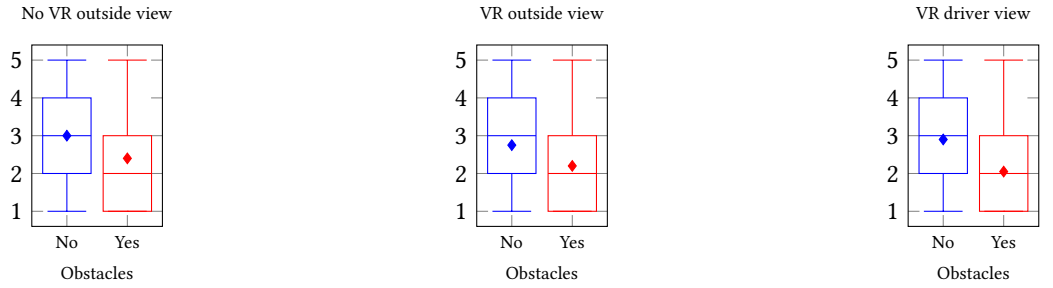


Fig. 7. Different VR-related views grouped by scenario's complexity

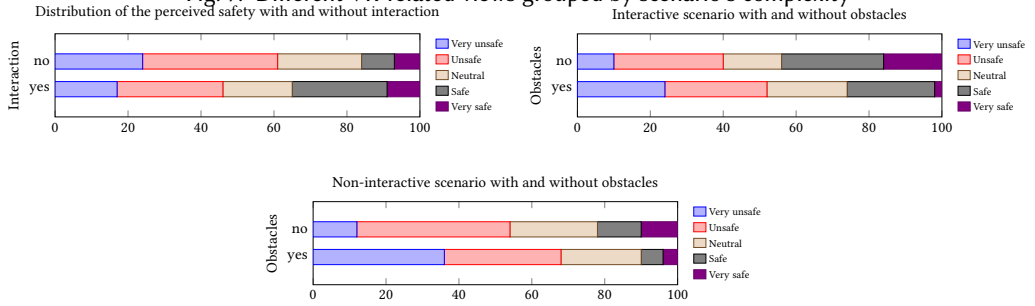


Fig. 8. Safety perception with and without interaction with the SDC (grouped by complexity)

Finding 4: Overall, participants found the test cases less safe with obstacles.

Participants' general comments during the experiment for each simulator qualitatively support Finding 4. Representative comments on BeamNG.tech driving behavior include: *"It did not look at safety lines, which is very dangerous if other traffic is involved. It also ran off the road multiple times, which can easily lead to a loss of control. Also, the car crashed into easily avoidable obstacles." - (P24/B)*; *"At least the AI seems to have an understanding of the general elements of the simulation, like the road. However, it seems to struggle with bumps in the middle of the road and also seems to drive too fast in curvy situations." - (P31/B).*

4.1.4 Different Views with Different Complexity. In the case of CARLA, we got the following representative comments on the driving behavior with regard to different complexity of the scenario: *"Except at the roundabouts, the car followed traffic rules, signals, and speed limits. However, it kept crashing and losing control in the roundabouts." - (P27/C)*; *"In most scenarios, the AI did well. From what I have seen during the simulations, it is not able to drive around roundabouts and does not stop at stop signs." - (P31/C)*; *"Very slow driving, unsmooth behavior, always too close to roundabout and abrupt stopping in front of obstacles." - (P41/C).*

We observed that the perception of safety dropped when the complexity increased (i.e., adding obstacles to the scenario). This observation is coherent among both simulators, BeamNG.tech and CARLA, as reported by the participants during the experiment.

4.2 RQ_2: Impact of Human Interaction on the Assessments of SDCs

To assess the safety perception of test cases with human interaction with the SDC, participants controlled the SDC's speed during the test execution. Figure 8 shows the Likert scale of responses. We compared the responses where participants could and could not control the car when obstacles were present.

4.2.1 Safety Perception with and without Interaction with the SDC. In general, interacting with the SDC enhances participants' perception of safety. From appx. B.2, we observe a statistically significant difference, leading to the following finding:

> *Finding 5:* Safety perception of test cases is not static: When users can interact with the SDC, participants feel significantly safer ($p = 0.013$) compared to when they cannot.

The participants' justification supports Finding5, e.g., controlling the SDC speed enhances safety perception, as P1 reported: *"The fact I could control the car when needed gave me a safer perception of the driving experience. Moreover, I could speed up the car when I wanted to." - (P1).* However, not all participants perceived interaction-based test cases as inherently safe. For instance, participant P4 commented: *"With a bit of control, it feels safer, especially being able to adjust the speed in dangerous situations. However, it is still not safe since the car ends up going off-road at the end of the scenario." - (P4).* While the SDC remains self-steering, it may still crash despite having speed control capability.

4.2.2 Safety Perception for with and without Obstacles. When interactive test cases involved obstacles, participants perceived them as less safe than obstacle-free scenarios. A statistically significant difference leads to the following finding:

> *Finding 6:* Incorporating obstacles into the simulation, where participants interact with the SDC, leads to significantly lower perceived safety in test cases ($p = 0.026$) compared to obstacle-free interactive scenarios.

This finding is also coherent with the answers of the study participants, e.g., by P4: *"It felt safer, especially since it was stopping the speed when it had another car in front. However, it still went to the footpath, making it not safe" - (P4).* From the comment, we observe safer perception through speed control. P20 also states: *"it could have stopped before hitting the camion" - (P20).*

However, as the study participant could not control the SDC's steering, some accidents remained unavoidable, as reported by P19: *"Hit the bike driver" (P19).* P40 gave a clearer comment: *"Two matters: 1) driver keeps its distance to the can in the front, but with sharp breaks instead of slowing down the car. 2) unable to avoid strange behaviors and drove next to a car with unstable drive and had an accident" (P40).* The participant could maintain the distance by adjusting the speed, but accidents could occur during lane changes.

In non-interactive test cases, obstacles induced insecurity among participants. However, the level of safety they felt when obstacles were included was higher in the case where the participants could interact with the SDC. This leads to the following finding:

> *Finding 7:* In the simulation, obstacles in non-interactive SDC test cases reduce the safety perception ($p = 0.013$). Yet, the ability to interact with the car raises more discomfort (making participants feel less safe) when obstacles are present.

Besides the statistical tests, we also noted participant comments supporting Finding 7. Some expressed discomfort in obstacle scenarios without the ability to control the car, as evident in the following example: *"The car was breaking and accelerating a lot while being behind the other car, and also the other car was not behaving safely on the road, ending the simulation with an accident between the two, so it felt quite unsafe overall." (P25).* Some participants also experienced the worst-case scenario without control, as reported by P28: *"It drove extremely close up to the ambulance car and finally crashed into it. therefore, the worst case happens." (P28).*

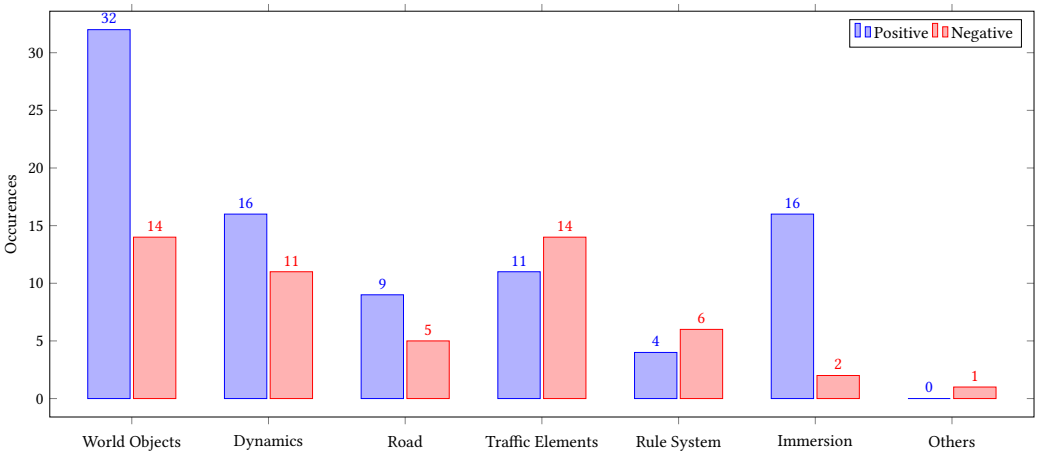


Fig. 9. Taxonomy of positive and negative factors impacting the perceived test cases' realism

Table 2. Taxonomy description including # of positive and negative comments on the perception of realism

Category	Description	Occurrences		
		Positive	Negative	Total
World Objects	This category relates to comments of participants on the accuracy of visual looks and design of all elements in the virtual environment, such as the weather, landscape, car design, traffic objects, etc., and how the graphical resolution is perceived.	32	14	46
Dynamics	This category relates to participants' comments on the physical dynamics of the elements in the virtual environment. For example, if the movement of the cars is physically realistic and reasonable or if crashes are realistically simulated from a physical perspective.	16	11	27
Road	This category relates to participants' comments on the road itself; to what extent the shape, surface, and structure are reasonably expected in the real world.	9	5	14
Traffic Elements	This category relates to participants' comments on the placement of the elements in the virtual environment. Furthermore, this category considers comments on the location and scale of the placed elements but also the quantity of the elements.	11	14	25
Rule System	This category relates to participants' comments on the traffic laws and the common sense of humans for resolving certain issues in specific traffic situations. A car should, for example, stop at a red signal and stop signs. Furthermore, the car should not drive recklessly and avoid dangerous situations (e.g., driving too close to other vehicles).	4	6	10
Immersion	This category relates to participants' comments on the immersive experiences. It applies to comments where participants express their feelings on how they experience the virtual environment and how they acoustically, visually, physically, and haptically sense it.	16	2	18
Others	This category relates to participants' comments that do not fit into the above categories.	0	1	1

4.3 RQ$_3$: Taxonomy on Realism

Realism is a crucial aspect to consider when evaluating test case *safety*. We created a taxonomy to gauge the perceived realism of study participants. Two coders used open card sorting on 50 comments each to establish categories, which were later reviewed by a third coder. Table 2 presents the seven resulting categories with their descriptions.

Next, two coders independently classified 100 comments using the designed taxonomy. Disagreements were resolved by a third coder. Table 2 and Figure 9 show the classification of comments related to question Q4 in steps ❺ and ⓫. We categorized comments as *positives* (increasing realism)

and *negatives* (decreasing realism) in the taxonomy. We observe that most classifications fall under *World Objects*, totaling 46, with 32 positives and 14 negatives.

> *Finding 8:* Several factors (e.g., the surroundings, car design, and object scale) impact the participants' perceived realism. The *World Objects* category dominates with 32 *positive* (e.g., car design) and 14 *negative* (e.g., traffic objects) aspects affecting realism perception.

Examples of positive comments with the BeamNG.tech simulator: *"The realism is quite good, especially in the car design. The car structure was damaged after crashing; the wheels were getting broken, and there was smoke coming out. The inside view of the car was also pretty real, with the driver's hand moving the steering wheel and all the car panel commands. [...] ." - (B/P4)*; *"They respect the scale from the objects." - (B/P22)*. Examples of positive comments for the CARLA simulator: *"The surroundings have more detail, which made it feel more realistic." - (C/P31)*; *"The environment (lighting, obstacles) feels quite real." - (C/P17)*. An example of a negative comment: *"The grass, the horizon as well, and the red vertical lines do not look very realistic." - (B/P3)*. Besides finding in Section 8, we noted that the *Immersion* category generally received positive comments about perceived realism.

> *Finding 9:* The *Immersion* category primarily comprises comments on factors that affect realism (e.g., view, perspective). It includes 16 *positive* (e.g., the realism of driver's seat) and 2 *negative* (e.g., low realism outside the vehicle) comments influencing participants' perceived realism.

This finding is reasonable since a driver sits in the driver's seat, unlike the perspective in a video game. The following quotes support this: *"The driver seat simulator felt very realistic." - (B/P14)*; *"It was different when I sat in the car than from outside, so it felt more real. But still looked like a game, so not that realistic." - (B/P21)*. In summary, comments on *Immersion* were positive, indicating that the driver seat viewpoint and VR usage enhanced perceived realism.

5 DISCUSSION

We first discuss safety considerations for simulation-based tests, including RQ_1 and interactive test cases RQ_2. Then, we delve into realism by discussing the taxonomy of influencing factors.

5.1 RQ_1 & RQ_2: Human-Based Safety Assessment of Simulation-Based Test Cases

The study participants perceived passing test cases (OOB metric not violated) as safer than failing ones (Finding 1), aligning with the OOB metric-based test oracle. This observation is supported by [36], where participants' assessment of driving quality correlates with metrics related to the SDC's lateral position. The OOB metric generally reflects test case safety. However, the extent to which the safety perception varies depending on certain simulation factors (e.g., obstacle inclusion) remains unclear. Hence, we conducted experiments with test cases featuring additional obstacles.

In Section 2, we found that adding obstacles to a passing test case does not significantly affect safety perception. However, participants perceive failing test cases as less safe with additional obstacles. Therefore, human safety perception does not proportionally align with the OOB metric. The OOB metric can be violated, but it still does not distinguish the case if there are additional obstacles in the test case, but the human does and perceives the test case unsafer.

We experimented with different immersion levels (i.e., various viewpoints), and as reported in Finding 3, participants using VR headsets perceived test cases as slightly less safe. This perception change is minimal when evaluating VR. Consequently, when using humans as oracles, outcomes vary based on immersion levels in virtual environments. Hence, similar human-based studies on simulation-based test cases for SDCs [36] may exhibit a slight bias if immersion is not considered. When grouping safety perceptions of test cases by their assessed viewpoints, cases with obstacles

were generally perceived as less safe than those without obstacles (Finding 4). Thus, using the OOB metric as an oracle may not always accurately represent safety perceptions from a human perspective. This observation aligns with the example illustrated by Figure 2a and Figure 2b.

As shown in Finding 5, participants perceived test cases as safer when they could control the vehicle's speed (i.e., they express a higher trust level in the SDC behavior), which means that the safety perception of simulation-based test cases depends on the user interaction levels. Having control over the vehicle impacts safety perception, which may not align with the OOB metric. In the case of test cases involving participant interaction, safety perception generally decreases when obstacles are present, as indicated by Finding 6. This aligns with the findings for non-interactive test cases, as highlighted in Finding 7.

5.2 RQ_3: Taxonomy on Test Cases' Realism

As shown in Finding 8, most participants' comments on Question Q4 fall under the *World Objects* category. As discussed in Section 1, we conjecture that assessing test case safety should also consider realism. The importance of *World Objects*, with respect to realism, confirms the fact that pure lane-keeping (as it is the focus of OOB) is not enough for doing a realistic safety assessment. Given that most comments related to test case realism are categorized as *World Objects*, it becomes essential to prioritize when evaluating test case safety. The *Immersion* category predominantly features comments expressing a positive or heightened sense of realism, as revealed in Finding 9. Participants' immersion, particularly their viewpoint, influences perceived realism. Notably, the driver seat perspective yields a higher realism perception, as evident in comments on Finding 9, consequently impacting safety perception. The importance of immersion, with respect to realism, confirms that static 2D assessment (again, as it is the focus for OOB) is not enough for doing a realistic safety assessment.

When we take a closer look at the participants' demographics and how they assess the level of realism, we observed that the participants in the age range between 18 and 30 years tend to assess the test cases 17% more realistically (Likert scale) than the older participants. Another insight is that we do not observe a different assessment of realism among the genders. Hence, there are confounding factors that influence the perception of realism, such as the age of the participant. This aspect suggests that the reality-gap characteristics are not deterministic measures as they depend on the human perception that might vary, as for the case of the participants'age.

5.3 Implications & Lessons Learned

The oracle definition for SDCs is many-fold as the safety has different aspects characterizing it. The OOB metric may not always reflect human safety perception in test cases due to various unaccounted factors. To enhance simulation-based testing, SDC testers and practitioners should consider devising alternative metrics that better align with human safety perception. Interacting with the car boosts perceived safety, potentially due to distrust in the AI driving the SDC. Future research should explore this further, ruling out other influencing factors. If low trust in AI is the main issue, this suggests shaping the direction of autonomous driving research toward increasing the level of trustworthiness of SDCs, which represents an important limiting factor to SDC real-world adoption.

As motivated in Section 1, realism significantly influences the safety perception of SDCs, as reflected in participants' comments on Q4. For this reason, we have created a taxonomy of factors that affect realism in simulation-based SDC testing, to guide future research in the field. The taxonomy provides an overview of factors impacting the realism of SDC simulation-based testing. We argue that our taxonomy is instrumental in supporting future research on the perceived *reality-gap*, which is critical to bridge the gap between the simulation-based outcome of a test case and

what happens eventually in the real world. Furthermore, we think the taxonomy provides a base for investigating similar limitations in other CPS application domains, which leverage simulation environments and target to improve the human perception of the realism and safety of CPSs.

6 THREATS TO VALIDITY

6.1 Threats to Internal Validity

The study participants rated safety and realism based on their immersion into the scenario. To limit the risks of unbiased assessments, we employed modern VR technology (HTC Vive Pro 2) to enhance immersion. The simulators, BeamNG.tech and CARLA, utilize distinct predefined maps. BeamNG.tech employs a flat map from the SBST tool competition [51], while CARLA uses built-in urban-like maps, which impose some constraints on road definition. These differing maps may lead to varying perceptions of test case safety and realism due to their distinct natures. This is something we plan to investigate for future work.

The different personal interactions with the study participants might influence the participants' focus during the experiments. To limit this risk, we used a protocol sheet during the experiments to ensure that all steps of the experiments were equally performed to minimize this threat.

6.2 Threats to External Validity

We recruited study participants primarily from an academic computer science background, which may not represent the general population. To address this potential bias, we ensured diversity in terms of age, gender, and driving experience, reducing the influence of factors beyond professional background. Another concern is the focus on the OOB metric, which may introduce bias as there are various metrics for evaluating SDCs in simulation environments. We chose OOB due to its widespread use among researchers and practitioners, as documented in recent studies [11, 25, 28, 36, 51]. Our study's limited use of only two simulators, BeamNG.tech and CARLA, restricts the generalizability of our findings to these specific platforms. However, we selected them because they are widely adopted in academia and industry, ensuring the reproducibility of our results compared to less-maintained options such as Udacity[1] and SVL [56].

7 RELATED WORK

In this section, we elaborate on related work on testing in virtual environments and assessing the quality of oracles in the context of CPS. We group the recent and ongoing research concerning topics that are relevant to our investigation such as (i) simulation-based testing, (ii) the testing metrics adopted, the oracle problem, and (iii) VR in software engineering.

7.1 Simulation-Based Testing

The automated testing of cyber-physical systems (CPSs) remains an ongoing research challenge [61, 75]. In this context, simulation-based testing emerges as a promising approach to enhance testing practices for safety-critical systems [10, 11, 14, 49, 54] and to support test automation [4, 5, 70, 71, 73]. Past research on testing CPS in simulation environments focused on monitoring CPS and predicting unsafe states [61, 65] of the systems using simulation environments [65, 74] as well as generating scenarios programmatically [52] or based on real-world observations [24, 64]. Recent research also proposed cost-effective regression testing techniques, including test selection [10], prioritization [7, 11] and minimization techniques to expose CPS faults or bugs earlier in the development and testing process. This research effort fundamentally contributed toward more robust and reliable simulation-based testing practices. However, it remains challenging to replicate

[1] https://github.com/udacity/self-driving-car

the same bugs observed in physical tests within simulations [3, 71] and generate representative simulated test cases that uncover realistic bugs [4]. Hence, previous research in the field was conducted on the premise that simulation environments sufficiently represent, with high fidelity, safety-critical aspects of the real world according to human judgments. In our paper, we hypothesize that the current simulation-based testing of SDCs (and general CPSs) does not always align with the human perception of safety and realism, which heavily impacts the effectiveness of simulation-based testing in general. To that end, in our research, we investigated when and why the safety metrics of simulation-based test cases of SDCs match human perception.

7.2 Testing Metrics & Oracle Problem

Automatically inferring the expected test outcome from a given input remains an unsolved challenge, which is known as the oracle problem. Many research papers propose some techniques to address this problem in the context of traditional software systems, such as generating oracles [6] or improving already existing test oracles [35, 67–69]. In either case, the previous research does not show an approach that produces fully optimal and effective oracles. However, while the oracle problem still remains an open challenge that requires humans to define the oracle, for the sake of test automation, several code coverage and mutation score metrics have been proposed for quantitatively assessing the quality of traditional software systems.

Software engineering for CPS is increasingly explored, with recent efforts mainly focused on bug characterization [26], testing [1, 21, 79], and verification [18] of self-adaptive CPSs. Another emerging area of research is related to the automated generation of oracles for testing and localizing faults in CPSs based on simulation technologies. For instance, Menghi *et al.* [44] proposed SOCRaTes, an approach to automatically generate online test oracles in Simulink able to handle CPS Simulink models featuring continuous behaviors and involving uncertainties. The oracles are generated from requirements specified in a signal logic-based language. In this context, for the sake of test automation, just like traditional software testing, simulation-based testing of SDCs relies on an oracle that determines whether the observed behavior of a system under test is safe or unsafe. To that end, current research on automated safety assessment focuses primarily on a limited set of temporal and non-temporal safety metrics for SDCs [10, 25, 51, 66]. In particular, the out-of-bound (OOB) non-temporal metric is largely adopted for assessing SDCs in simulation-based testing [25, 49, 51], to determine if a test case fails or passes. However, it is yet unclear whether this metric serves as a meaningful oracle for assessing the safety behavior of SDCs in simulation-based testing in general.

This study is built on our hypothesis that current simulation-based testing of SDCs does not always align with the human perception of safety and realism, and for this reason, we focus on understanding and characterizing this mismatch in our research. Close to our work, a recent study [36] conducted a human-based study and observed that correlations between the computed quality metrics and the perceived quality by humans are meaningful for assessing the test quality for SDCs. However, such previous work did not investigate the factors that define the test quality and realism of the simulation environments from a human point of view with the use of virtual reality [60] as done in our work.

A critical concern concerning the oracle problem in simulation-based testing is represented by the *Reality Gap* [4, 37, 48, 55, 71]. Due to the different properties of simulated and real contexts, the former may not be a faithful mirroring of the latter. Simulations are necessarily simplified for computational feasibility yet reflect real-world phenomena at a given level of veracity, the extent of which is the result of a trade-off between accuracy and computational time [19]. Robotics simulations rely on the replication of phenomena that are difficult to accurately replicate, e.g., simulating actuators (i.e., torque characteristics, gear backlash), sensors (i.e., noise, latency), and rendered images (i.e., reflections, refraction, textures). This gap between reality and simulation

is commonly referred to as the *reality-gap* [19]. A closely related problem concerns the concrete realistic *bug reproduction* and exposure in simulation environments [4, 71]. It is indeed challenging to capture the same bugs as physical tests [3, 71] and to *generate effective test cases* that can expose real-world bugs in simulation [4]. While recent studies provide solutions for addressing the reality gap (e.g., leveraging domain randomization techniques or using data from real-world observations) [17, 19, 40, 41, 57, 77] in the development phase of CPS, there is no prior study that investigated and/or characterized the perception of realism of SDC test cases from human participants. This study focuses on addressing this specific open question in the context of RQ3.

7.3 Immersion Technology in Software Engineering

Furthermore, using VR for software engineering was also considered by [32, 43] but with another focus as well. They used VR to gain design knowledge from legacy systems by using different visualization approaches and immersion technologies. Furthermore, most papers [42, 58, 59] referring to the potential use of VR and AR for the workspace of software development teams. In general, the use of VR and AR in software engineering is not well studied yet, and the only papers available or mainly vision papers for future research [45]. However, in our work, we present a practical application of VR for assessing the test oracles with a Human-in-the-Loop approach.

8 CONCLUSION

In this study, we explored when and why safety metrics align with human perception in SDC testing. We conducted an empirical study with 50 participants from diverse backgrounds, evaluating their perception of test case safety and realism. We observed that the safety perception of SDC significantly decreases as test case complexity rises. Interestingly, safety perception improves when participants can control the SDC's speed, indicating that OOB metric is not sufficient to match/model human (more subjective) factors. Additionally, realism perception varies with the complexity of scenarios (i.e., object additions) and different participant viewpoints. These findings emphasize the need for more meaningful safety metrics that align with human perception of *safety* and *realism* to bridge the current problem of the *reality-gap* in simulation-based testing. In future work plans, we aim to extend our study by varying weather and light conditions, adding more objects, and incorporating alternative safety metrics beyond the conventional single-objective OOB metric used in BeamNG.tech and CARLA [66].

9 DATA AVAILABILITY

A replication package with data, code, and appendices is publicly available on Zenodo [12, 13].

ACKNOWLEDGMENTS

We thank the Horizon 2020 (EU Commission) support for the COSMOS project, Project No. 957254.

10 CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Christian Birchler: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Project Administration, Resources, Software, Validation, Visualization, Writing – Original Draft Preparation. **Tanzil Kombarabettu Mohammed**: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Project Administration, Resources, Software, Visualization. **Pooja Rani**: Methodology, Supervision, Writing – Review & Editing. **Teodora Nechita**: Data Curation, Methodology. **Timo Kehrer**: Methodology, Resources, Supervision, Writing – Review & Editing. **Sebastiano Panichella**: Conceptualization, Funding Acquisition, Methodology, Project Administration, Resources, Supervision, Writing – Review & Editing.

REFERENCES

[1] Raja Ben Abdessalem, Shiva Nejati, Lionel C. Briand, and Thomas Stifter. 2018. Testing vision-based control systems using learnable evolutionary algorithms. In *International Conference on Software Engineering*. 1016–1026. https://doi.org/10.1145/3180155.3180160

[2] Raja Ben Abdessalem, Annibale Panichella, Shiva Nejati, Lionel C. Briand, and Thomas Stifter. 2020. Automated repair of feature interaction failures in automated driving systems. In *International Symposium on Software Testing and Analysis*. ACM, 88–100. https://doi.org/10.1145/3395363.3397386

[3] Afsoon Afzal, Deborah S. Katz, Claire Le Goues, and Christopher Steven Timperley. 2020. A Study on the Challenges of Using Robotics Simulators for Testing. arXiv:2004.07368 https://arxiv.org/abs/2004.07368

[4] Afsoon Afzal, Deborah S. Katz, Claire Le Goues, and Christopher Steven Timperley. 2021. Simulation for Robotics Test Automation: Developer Perspectives. In *Conference on Software Testing, Verification and Validation*. IEEE, 263–274. https://doi.org/10.1109/ICST49551.2021.00036

[5] Miguel Alcon, Hamid Tabani, Jaume Abella, and Francisco J. Cazorla. 2021. Enabling Unit Testing of Already-Integrated AI Software Systems: The Case of Apollo for Autonomous Driving. In *Conference on Digital System Design*. IEEE, 426–433. https://doi.org/10.1109/DSD53832.2021.00071

[6] Aitor Arrieta, Maialen Otaegi, Liping Han, Goiuria Sagardui, Shaukat Ali, and Maite Arratibel. 2022. Automating Test Oracle Generation in DevOps for Industrial Elevators. In *International Conference on Software Analysis, Evolution and Reengineering*. IEEE, 284–288. https://doi.org/10.1109/SANER53432.2022.00044

[7] Aitor Arrieta, Shuai Wang, Goiuria Sagardui, and Leire Etxeberria. 2019. Search-Based test case prioritization for simulation-Based testing of cyber-Physical system product lines. *J. Syst. Softw.* 149 (2019), 1–34. https://doi.org/10.1016/j.jss.2018.09.055

[8] BBC. 2023. Robots to do 39% of domestic chores by 2033, say experts. https://www.bbc.com/news/technology-64718842. Accessed: 2023-01-04.

[9] BeamNG.tech. [n. d.]. BeamNG.research. https://documentation.beamng.com/beamng_tech/. Accessed: 2022-07-31.

[10] Christian Birchler, Sajad Khatiri, Bill Bosshard, Alessio Gambi, and Sebastiano Panichella. 2023. Machine learning-based test selection for simulation-based testing of self-driving cars software. *Empir. Softw. Eng.* 28, 3 (2023), 71. https://doi.org/10.1007/s10664-023-10286-y

[11] Christian Birchler, Sajad Khatiri, Pouria Derakhshanfar, Sebastiano Panichella, and Annibale Panichella. 2023. Single and Multi-objective Test Cases Prioritization for Self-driving Cars in Virtual Environments. *ACM Trans. Softw. Eng. Methodol.* 32, 2 (2023), 28:1–28:30. https://doi.org/10.1145/3533818

[12] Christian Birchler, Tanzil Kombarabettu Mohammed, Pooja Rani, Teodora Nechita, Timo Kehrer, and Sebastiano Panichella. 2024. Replication Package - "How does Simulation-based Testing for Self-driving Cars match Human Perception?". https://doi.org/10.5281/zenodo.10570961

[13] Christian Birchler, Tanzil Kombarabettu Mohammed, Pooja Rani, Teodora Nechita, Timo Kehrer, and Sebastiano Panichella. 2024. Replication Package - "How does Simulation-based Testing for Self-driving Cars match Human Perception?". https://doi.org/10.5281/zenodo.10570960

[14] Christian Birchler, Cyrill Rohrbach, Hyeongkyun Kim, Alessio Gambi, Tianhai Liu, Jens Horneber, Timo Kehrer, and Sebastiano Panichella. 2023. TEASER: Simulation-Based CAN Bus Regression Testing for Self-Driving Cars Software. In *International Conference on Automated Software Engineering*. 2058–2061. https://doi.org/10.1109/ASE56229.2023.00154

[15] Tim Bohne, Gurunatraj Parthasarathy, and Benjamin Kisliuk. 2023. A systematic approach to the development of long-term autonomous robotic systems for agriculture. In *43. GIL-Jahrestagung, Resiliente Agri-Food-Systeme (LNI, Vol. P-330)*. Gesellschaft für Informatik e.V., 285–290. https://dl.gi.de/20.500.12116/40260

[16] Ezequiel Castellano, Ahmet Cetinkaya, Cédric Ho Thanh, Stefan Klikovits, Xiaoyi Zhang, and Paolo Arcaini. 2021. Frenetic at the SBST 2021 Tool Competition. In *International Workshop on Search-Based Software Testing*. IEEE, 36–37. https://doi.org/10.1109/SBST52555.2021.00016

[17] Yevgen Chebotar, Ankur Handa, Viktor Makoviychuk, Miles Macklin, Jan Issac, Nathan D. Ratliff, and Dieter Fox. 2019. Closing the Sim-to-Real Loop: Adapting Simulation Randomization with Real World Experience. In *International Conference on Robotics and Automation*. IEEE, 8973–8979. https://doi.org/10.1109/ICRA.2019.8793789

[18] Shafiul Azam Chowdhury, Sohil Lal Shrestha, Taylor T. Johnson, and Christoph Csallner. 2020. SLEMI: equivalence modulo input (EMI) based mutation of CPS models for finding compiler bugs in Simulink. In *International Conference on Software Engineering*. 335–346. https://doi.org/10.1145/3377811.3380381

[19] Jack Collins, Ross Brown, Jurgen Leitner, and David Howard. 2020. Traversing the reality gap via simulator tuning. *arXiv preprint arXiv:2003.01369* (2020).

[20] Hugo Leonardo da Silva Araujo, Mohammad Reza Mousavi, and Mahsa Varshosaz. 2023. Testing, Validation, and Verification of Robotic and Autonomous Systems: A Systematic Review. *ACM Trans. Softw. Eng. Methodol.* 32, 2 (2023), 51:1–51:61. https://doi.org/10.1145/3542945

[21] Jyotirmoy V. Deshmukh, Marko Horvat, Xiaoqing Jin, Rupak Majumdar, and Vinayak S. Prabhu. 2017. Testing Cyber-Physical Systems through Bayesian Optimization. *ACM Trans. Embed. Comput. Syst.* 16, 5s (2017), 170:1–170:18. https://doi.org/10.1145/3126521

[22] Alexey Dosovitskiy, Germán Ros, Felipe Codevilla, Antonio M. López, and Vladlen Koltun. 2017. CARLA: An Open Urban Driving Simulator. In *Annual Conference on Robot Learning (Proceedings of Machine Learning Research, Vol. 78)*. PMLR, 1–16. http://proceedings.mlr.press/v78/dosovitskiy17a.html

[23] Alessio Gambi, Tri Huynh, and Gordon Fraser. 2019. Automatically reconstructing car crashes from police reports for testing self-driving cars. In *International Conference on Software Engineering: Companion Proceedings*. IEEE / ACM, 290–291. https://doi.org/10.1109/ICSE-Companion.2019.00119

[24] Alessio Gambi, Tri Huynh, and Gordon Fraser. 2019. Generating effective test cases for self-driving cars from police reports. In *Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. ACM, 257–267. https://doi.org/10.1145/3338906.3338942

[25] Alessio Gambi, Gunel Jahangirova, Vincenzo Riccio, and Fiorella Zampetti. 2022. SBST Tool Competition 2022. In *International Workshop on Search-Based Software Testing*. IEEE, 25–32. https://doi.org/10.1145/3526072.3527538

[26] Joshua Garcia, Yang Feng, Junjie Shen, Sumaya Almanee, Yuan Xia, and Qi Alfred Chen. 2020. A comprehensive study of autonomous vehicle bugs. In *International Conference on Software Engineering*. ACM, 385–396. https://doi.org/10.1145/3377811.3380397

[27] BeamNG GmbH. 2023. BeamNG.tech. https://beamng.tech/

[28] BeamNG GmbH. 2023. Publications based on BeamNG.tech. https://beamng.tech/research/

[29] The Guardian. 2018. Self-driving Uber kills Arizona woman in first fatal crash involving pedestrian. https://www.theguardian.com/technology/2018/mar/19/uber-self-driving-car-kills-woman-arizona-tempe

[30] Rodrigo Gutiérrez-Moreno, Rafael Barea, Elena López Guillén, Javier Araluce, and Luis Miguel Bergasa. 2022. Reinforcement Learning-Based Autonomous Driving at Intersections in CARLA Simulator. *Sensors* 22, 21 (2022), 8373. https://doi.org/10.3390/s22218373

[31] Carl Hildebrandt and Sebastian G. Elbaum. 2021. World-in-the-Loop Simulation for Autonomous Systems Validation. In *International Conference on Robotics and Automation*. IEEE, 10912–10919. https://doi.org/10.1109/ICRA48506.2021.9561240

[32] Adrian Hoff, Michael Nieke, and Christoph Seidl. 2021. Towards immersive software archaeology: regaining legacy systems' design knowledge via interactive exploration in virtual reality. In *Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. ACM, 1455–1458. https://doi.org/10.1145/3468264.3473128

[33] Jiawei Huang and Alexander Klippel. 2020. The Effects of Visual Realism on Spatial Memory and Exploration Patterns in Virtual Reality. In *Symposium on Virtual Reality Software and Technology*. ACM, 18:1–18:11. https://doi.org/10.1145/3385956.3418945

[34] Jiawei Huang, Melissa S. Lucash, Mark B. Simpson, Casey Helgeson, and Alexander Klippel. 2019. Visualizing Natural Environments from Data in Virtual Reality: Combining Realism and Uncertainty. In *Conference on Virtual Reality and 3D User Interfaces*. IEEE, 1485–1488. https://doi.org/10.1109/VR.2019.8797996

[35] Gunel Jahangirova, David Clark, Mark Harman, and Paolo Tonella. 2016. Test oracle assessment and improvement. In *International Symposium on Software Testing and Analysis*. ACM, 247–258. https://doi.org/10.1145/2931037.2931062

[36] Gunel Jahangirova, Andrea Stocco, and Paolo Tonella. 2021. Quality Metrics and Oracles for Autonomous Vehicles Testing. In *Conference on Software Testing, Verification and Validation*. IEEE, 194–204. https://doi.org/10.1109/ICST49551.2021.00030

[37] Sajad Khatiri, Sebastiano Panichella, and Paolo Tonella. 2023. Simulation-based Test Case Generation for Unmanned Aerial Vehicles in the Neighborhood of Real Flights. In *International Conference on Software Testing, Verification and Validation*. IEEE, 281–292. https://doi.org/10.1109/ICST57152.2023.00034

[38] Sajad Khatiri, Sebastiano Panichella, and Paolo Tonella. 2024. Simulation-based Testing of Unmanned Aerial Vehicles with Aerialist. In *International Conference on Software Engineering (ICSE)*.

[39] Sajad Khatiri, Prasun Saurabh, Timothy Zimmermann, Charith Munasinghe, Christian Birchler, and Sebastiano Panichella. 2024. SBFT Tool Competition 2024 - CPS-UAV Test Case Generation Track. In *IEEE/ACM International Workshop on Search-Based and Fuzz Testing, SBFT@ICSE 2024*.

[40] Sylvain Koos, Jean-Baptiste Mouret, and Stéphane Doncieux. 2013. The Transferability Approach: Crossing the Reality Gap in Evolutionary Robotics. *IEEE Trans. Evol. Comput.* 17, 1 (2013), 122–145. https://doi.org/10.1109/TEVC.2012.2185849

[41] Timothy E Lee, Jonathan Tremblay, Thang To, Jia Cheng, Terry Mosier, Oliver Kroemer, Dieter Fox, and Stan Birchfield. 2020. Camera-to-robot pose estimation from a single image. In *International Conference on Robotics and Automation*. IEEE, 9426–9432.

[42] Rohit Mehra, Vibhu Saujanya Sharma, Vikrant Kaulgud, Sanjay Podder, and Adam P. Burden. 2020. Immersive IDE: Towards Leveraging Virtual Reality for creating an Immersive Software Development Environment. In *International Conference on Software Engineering, Workshops*. ACM, 177–180. https://doi.org/10.1145/3387940.3392234

[43] Rohit Mehra, Vibhu Saujanya Sharma, Vikrant Kaulgud, Sanjay Podder, and Adam P. Burden. 2020. Towards Immersive Comprehension of Software Systems Using Augmented Reality - An Empirical Evaluation. In *International Conference on Automated Software Engineering*. IEEE, 1267–1269. https://doi.org/10.1145/3324884.3418907

[44] Claudio Menghi, Shiva Nejati, Khouloud Gaaloul, and Lionel C. Briand. 2019. Generating automated and online test oracles for Simulink models with continuous and uncertain behaviors. In *Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 27–38. https://doi.org/10.1145/3338906.3338920

[45] Leonel Merino, Mircea Lungu, and Christoph Seidl. 2020. Unleashing the Potentials of Immersive Augmented Reality for Software Engineering. In *International Conference on Software Analysis, Evolution and Reengineering*. IEEE, 517–521. https://doi.org/10.1109/SANER48275.2020.9054812

[46] Elena Molina, Alejandro Ríos Jerez, and Núria Pelechano Gómez. 2020. Avatars rendering and its effect on perceived realism in Virtual Reality. In *International Conference on Artificial Intelligence and Virtual Reality*. IEEE, 222–225. https://doi.org/10.1109/AIVR50618.2020.00046

[47] Saasha Nair, Sina Shafaei, Daniel Auge, and Alois C. Knoll. 2021. An Evaluation of "Crash Prediction Networks" (CPN) for Autonomous Driving Scenarios in CARLA Simulator. In *Workshop on Artificial Intelligence Safety (CEUR Workshop Proceedings, Vol. 2808)*. CEUR-WS.org. http://ceur-ws.org/Vol-2808/Paper_10.pdf

[48] Anthony Ngo, Max Paul Bauer, and Michael Resch. 2021. A Multi-Layered Approach for Measuring the Simulation-to-Reality Gap of Radar Perception for Autonomous Driving. In *International Intelligent Transportation Systems Conference*. IEEE, 4008–4014. https://doi.org/10.1109/ITSC48978.2021.9564521

[49] Vuong Nguyen, Stefan Huber, and Alessio Gambi. 2021. SALVO: Automated Generation of Diversified Tests for Self-driving Cars from Existing Maps. In *International Conference on Artificial Intelligence Testing*. IEEE, 128–135. https://doi.org/10.1109/AITEST52744.2021.00033

[50] Nvidia 2020. NVIDIA DRIVE Constellation. https://developer.nvidia.com/drive/drive-constellation

[51] Sebastiano Panichella, Alessio Gambi, Fiorella Zampetti, and Vincenzo Riccio. 2021. SBST Tool Competition 2021. In *International Workshop on Search-Based Software Testing*. IEEE, 20–27. https://doi.org/10.1109/SBST52555.2021.00011

[52] Mingyu Park, Hoon Jang, Taejoon Byun, and Yunja Choi. 2020. Property-based testing for LG home appliances using accelerated software-in-the-loop simulation. In *International Conference on Software Engineering*. ACM, 120–129. https://doi.org/10.1145/3377813.3381346

[53] Yi-Hao Peng, Carolyn Yu, Shi-Hong Liu, Chung-Wei Wang, Paul Taele, Neng-Hao Yu, and Mike Y. Chen. 2020. WalkingVibe: Reducing Virtual Reality Sickness and Improving Realism while Walking in VR using Unobtrusive Head-mounted Vibrotactile Feedback. In *Conference on Human Factors in Computing Systems*. ACM, 1–12. https://doi.org/10.1145/3313831.3376847

[54] Andrea Piazzoni, Jim Cherian, Mohamed Azhar, Jing Yew Yap, James Lee Wei Shung, and Roshan Vijay. 2021. ViSTA: a Framework for Virtual Scenario-based Testing of Autonomous Vehicles. In *International Conference on Artificial Intelligence Testing*. IEEE, 143–150. https://doi.org/10.1109/AITEST52744.2021.00035

[55] Fabio Reway, Abdul Hoffmann, Diogo Wachtel, Werner Huber, Alois C. Knoll, and Eduardo Parente Ribeiro. 2020. Test Method for Measuring the Simulation-to-Reality Gap of Camera-based Object Detection Algorithms for Autonomous Driving. In *Intelligent Vehicles Symposium*. IEEE, 1249–1256. https://doi.org/10.1109/IV47402.2020.9304567

[56] Guodong Rong, Byung Hyun Shin, Hadi Tabatabaee, Qiang Lu, Steve Lemke, Martins Mozeiko, Eric Boise, Geehoon Uhm, Mark Gerow, Shalin Mehta, Eugene Agafonov, Tae Hyung Kim, Eric Sterner, Keunhae Ushiroda, Michael Reyes, Dmitry Zelenkovsky, and Seonman Kim. 2020. LGSVL Simulator: A High Fidelity Simulator for Autonomous Driving. (2020), 1–6. https://doi.org/10.1109/ITSC45102.2020.9294422

[57] Erica Salvato, Gianfranco Fenu, Eric Medvet, and Felice Andrea Pellegrino. 2021. Crossing the Reality Gap: A Survey on Sim-to-Real Transferability of Robot Controllers in Reinforcement Learning. *IEEE Access* 9 (2021), 153171–153187. https://doi.org/10.1109/ACCESS.2021.3126658

[58] Vibhu Saujanya Sharma, Rohit Mehra, Vikrant Kaulgud, and Sanjay Podder. 2018. An immersive future for software engineering: avenues and approaches. In *International Conference on Software Engineering: New Ideas and Emerging Results*. ACM, 105–108. https://doi.org/10.1145/3183399.3183414

[59] Vibhu Saujanya Sharma, Rohit Mehra, Vikrant Kaulgud, and Sanjay Podder. 2019. An extended reality approach for creating immersive software project workspaces. In *International Workshop on Cooperative and Human Aspects of Software Engineering*. IEEE / ACM, 27–30. https://doi.org/10.1109/CHASE.2019.00013

[60] Gustavo Silvera, Abhijat Biswas, and Henny Admoni. 2022. DReyeVR: Democratizing Virtual Reality Driving Simulation for Behavioural & Interaction Research. In *International Conference on Human-Robot Interaction*, Daisuke Sakamoto, Astrid Weiss, Laura M. Hiatt, and Masahiro Shiomi (Eds.). IEEE / ACM, 639–643. https://doi.org/10.1109/HRI53351.2022.9889526

[61] Andrea Di Sorbo, Fiorella Zampetti, Aaron Visaggio, Massimiliano Di Penta, and Sebastiano Panichella. 2023. Automated Identification and Qualitative Characterization of Safety Concerns Reported in UAV Software Platforms. *ACM Trans. Softw. Eng. Methodol.* 32, 3 (2023), 67:1–67:37. https://doi.org/10.1145/3564821

[62] Donna Spencer. 2009. *Card sorting: Designing usable categories.* Rosenfeld Media.

[63] Jack Stilgoe. 2021. How can we know a self-driving car is safe? *Ethics Inf. Technol.* 23, 4 (2021), 635–647. https://doi.org/10.1007/s10676-021-09602-1

[64] Andrea Stocco, Brian Pulfer, and Paolo Tonella. 2023. Mind the Gap! A Study on the Transferability of Virtual Versus Physical-World Testing of Autonomous Driving Systems. *IEEE Trans. Software Eng.* 49, 4 (2023), 1928–1940. https://doi.org/10.1109/TSE.2022.3202311

[65] Andrea Stocco, Michael Weiss, Marco Calzana, and Paolo Tonella. 2020. Misbehaviour prediction for autonomous driving systems. In *International Conference on Software Engineering*. ACM, 359–371. https://doi.org/10.1145/3377811.3380353

[66] Shuncheng Tang, Zhenya Zhang, Yi Zhang, Jixiang Zhou, Yan Guo, Shuang Liu, Shengjian Guo, Yan-Fu Li, Lei Ma, Yinxing Xue, and Yang Liu. 2023. A Survey on Automated Driving System Testing: Landscapes and Trends. *ACM Trans. Softw. Eng. Methodol.* 32, 5 (2023), 124:1–124:62. https://doi.org/10.1145/3579642

[67] Valerio Terragni, Gunel Jahangirova, Mauro Pezzè, and Paolo Tonella. 2021. Improving assertion oracles with evolutionary computation. In *Genetic and Evolutionary Computation Conference, Companion Volume*. ACM, 45–46. https://doi.org/10.1145/3449726.3462722

[68] Valerio Terragni, Gunel Jahangirova, Paolo Tonella, and Mauro Pezzè. 2020. Evolutionary improvement of assertion oracles. In *Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. ACM, 1178–1189. https://doi.org/10.1145/3368089.3409758

[69] Valerio Terragni, Gunel Jahangirova, Paolo Tonella, and Mauro Pezzè. 2021. GAssert: A Fully Automated Tool to Improve Assertion Oracles. In *International Conference on Software Engineering: Companion Proceedings*. IEEE, 85–88. https://doi.org/10.1109/ICSE-Companion52605.2021.00042

[70] Christopher Steven Timperley, Afsoon Afzal, Deborah S Katz, Jam Marcos Hernandez, and Claire Le Goues. 2018. Crashing simulated planes is cheap: Can simulation detect robotics bugs early?. In *International Conference on Software Testing, Verification and Validation*. IEEE, 331–342.

[71] Dinghua Wang, Shuqing Li, Guanping Xiao, Yepang Liu, and Yulei Sui. 2021. An exploratory study of autopilot software bugs in unmanned aerial vehicles. In *Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. ACM, 20–31. https://doi.org/10.1145/3468264.3468559

[72] Lingfeng Wang and K.C. Tan. 2005. Software testing for safety critical applications. *IEEE Instrumentation & Measurement Magazine* 8, 2 (2005), 38–47. https://doi.org/10.1109/MIM.2005.1438843

[73] Franz Wotawa. 2021. On the Use of Available Testing Methods for Verification & Validation of AI-based Software and Systems. In *Workshop on Artificial Intelligence Safety (CEUR Workshop Proceedings, Vol. 2808)*. CEUR-WS.org. http://ceur-ws.org/Vol-2808/Paper_29.pdf

[74] Qinghua Xu, Shaukat Ali, and Tao Yue. 2021. Digital Twin-based Anomaly Detection in Cyber-physical Systems. In *Conference on Software Testing, Verification and Validation*. IEEE, 205–216. https://doi.org/10.1109/ICST49551.2021.00031

[75] Fiorella Zampetti, Ritu Kapur, Massimiliano Di Penta, and Sebastiano Panichella. 2022. An empirical characterization of software bugs in open-source Cyber–Physical Systems. *Journal of Systems and Software* 192 (2022), 111425. https://doi.org/10.1016/j.jss.2022.111425

[76] Eleni Zapridou, Ezio Bartocci, and Panagiotis Katsaros. 2020. Runtime Verification of Autonomous Driving Systems in CARLA. In *Runtime Verification - International Conference (Lecture Notes in Computer Science, Vol. 12399)*. Springer, 172–183. https://doi.org/10.1007/978-3-030-60508-7_9

[77] Fangyi Zhang, Jürgen Leitner, Zongyuan Ge, Michael Milford, and Peter Corke. 2019. Adversarial discriminative sim-to-real transfer of visuo-motor policies. *Int. J. Robotics Res.* 38, 10-11 (2019). https://doi.org/10.1177/0278364919870227

[78] Wei Zhang, Siyu Fu, Zixu Cao, Zhiyuan Jiang, Shunqing Zhang, and Shugong Xu. 2020. An SDR-in-the-Loop Carla Simulator for C-V2X-Based Autonomous Driving. In *Conference on Computer Communications*. IEEE, 1270–1271. https://doi.org/10.1109/INFOCOMWKSHPS50562.2020.9162743

[79] Husheng Zhou, Wei Li, Zelun Kong, Junfeng Guo, Yuqun Zhang, Bei Yu, Lingming Zhang, and Cong Liu. 2020. DeepBillboard: systematic physical-world testing of autonomous driving systems. In *International Conference on Software Engineering*. 347–358. https://doi.org/10.1145/3377811.3380422

Received 2023-09-28; accepted 2024-01-23