AAI ReplicaR: A Virtual Testing Platform with Comprehensive Testing Methodology for Autonomous Driving Systems

Awais Briandez, Ali Bin Wahid

Automotive Artificial Intelligence (AAI) GmbH

Essential building block for ADS Homologation

Abstract

AAI ReplicaR is a comprehensive virtual testing platform, certified by **TÜV SÜD [7]**, designed for the **verification and validation (V&V**) of **autonomous driving systems (ADS).** The platform supports diverse use cases, including scenario exploration, generation, and execution. This paper details the **Software in Loop (SIL)** testing with **ReplicaR's** Object List and Sensor Simulator, alongside the **Carla co-simulation client**, enabling the simulation of ADS in ReplicaR-supported environments. Utilizing **ASAM OpenDrive®** and capability of large-scale simulation mapping creation, ReplicaR facilitates the exploration of both known and unknown scenarios, critical for moving towards homologation. The platform's robust testing and validation processes enable thorough assessment and refinement of ADS, ensuring compliance with regulatory and safety standards.

Keywords: *Autonomous Driving Systems (ADS), Verification and Validation (V&V), Homologation, ReplicaR, Software In Loop (SIL), Scenario Generation, Carla Co-simulation, ASAM OpenDrive®*

1. Introduction

Autonomous driving systems (ADS) necessitate exhaustive testing and validation to ensure they meet the highest safety and regulatory standards. AAI ReplicaR provides a comprehensive framework for such rigorous testing through detailed scenario generation, execution, and data analysis. Certified by TÜV SÜD, ReplicaR supports various facets of ADS development, from defining operational design domains to generating synthetic data for validation.

2. AAI ReplicaR: An Overview

AAI ReplicaR (Figure 2) is a cloud-based simulation and testing platform designed specifically for autonomous vehicle development and validation. The platform interface includes several key sections and modules, each serving a specific purpose in the ADS workflow: The Operational Design Domain (ODD) section, with tools like **MyODD**, allows users to define and manage the static environmental conditions under which the ADS operates. Scenario Generation includes **MyScenarios** for creating custom scenarios and **AutoScenarios** for automated scenario generation using AAI Naturalistic Traffic Simulation. The **TestSuites** section enables users to create and manage

Figure 1 AAI ReplicaR enable users to integrate and test their vehicle systems, including ADAS/AD software, subsystem models, and hardware simulations for scenario-based-testing.

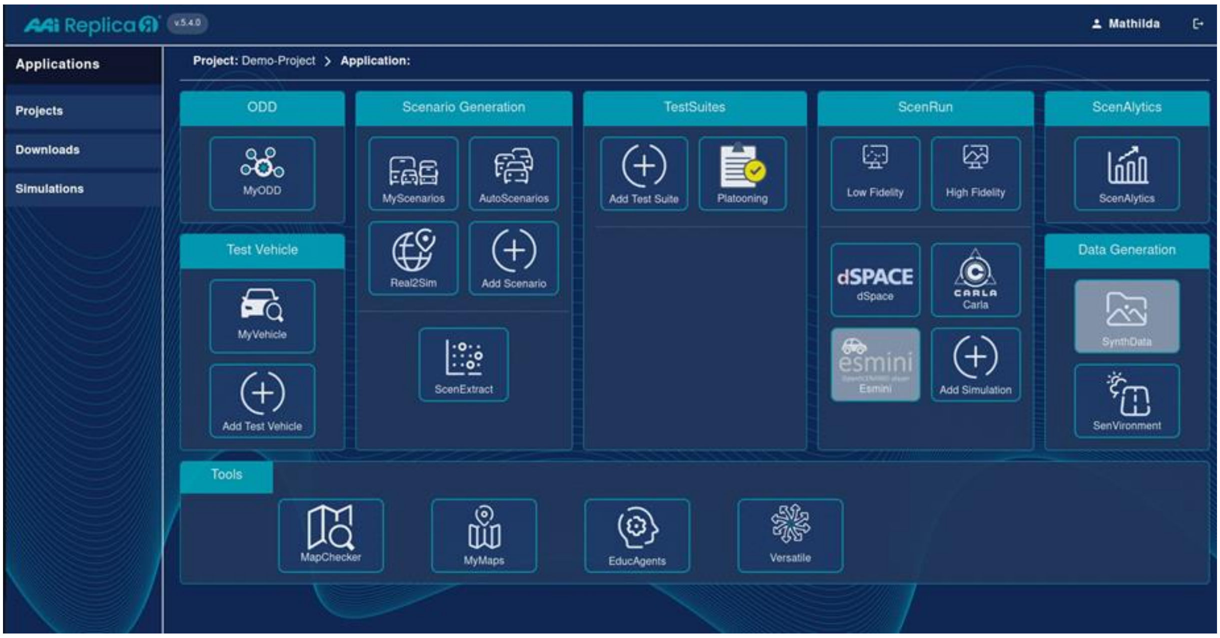

Figure 2 ReplicaR Platform Graphical User Interface Overview

collections of test cases, including specialized modules for vehicle platooning. In the Test Vehicle section, users can manage and configure test vehicles with tools like **MyVehicle** and Add Test Vehicle along with its Sensor Configuration which currently includes various cameras and Lidars. **ScenRun** offers options for running low (object list) and high-fidelity (Physics-based Sensor) simulations, also providing integration with 3rd party Simulation Engines like dSPACE, Carla, and Esmini with an ability to add more. Data Generation tools like **SynthData** and **SenVironment** generate synthetic data in ASAM OSI3 or ROS2 Bag dumps (for training Perception models on Synthetic data or for open-loop simulation) and simulate sensor environments which may then be exported as FBX-Scenes.

Additional tools include MapChecker for validating ASAM OpenDrive maps, **MyMaps** for extracting/creating ASAM OpenDrive, **EducAgents** to train traffic agents with reinforcement learning, and **Versatile** for editing and enhancing map and scene data.

3. Scenario Generation

Currently, ReplicaR incorporates two techniques for generating scenarios under the product names **AutoScenario** and **MyScenario**. AutoScenario employs AAI Naturalistic traffic (data-driven), while MyScenario uses Morphosynthesis (a statistical and controlled, knowledge-based approach). With ReplicaR's extensive database of ASAM OpenDrive maps (Europe Region extendable to Americas/Asia), along with its capability to import ASAM OpenDrive maps, these techniques can be used to explore or generate critical scenarios.

To evaluate the effectiveness of scenario generation methodologies, we consider three metrics: Automation Level, Definition, and Relevancy/Criticality.

Automation Level: Measures the extent of automation in scenario generation and exploration; higher levels indicate minimal human intervention.

Definition: Assesses the abstraction level of scenarios; concrete and logical definitions are based on detailed data and structured frameworks.

Relevancy/Criticality: Evaluates the importance and impact of scenarios in autonomous driving system testing; high relevancy signifies essential and influential scenarios for system reliability and safety.

These metrics are used to assess the scenario generation and exploration capabilities of AAI ReplicaR. Table 1. Shows the two methodologies on this scale.

Technique	Automation Level	Definition	Relevancy/Criticality
ReplicaR's Traffic	High	Not Applicable	Low
ReplicaR's Morphosynthesis	Manual with Human Insight	Concrete, Logical	High

Table 1 AAI Scenario generation methods and key evaluation metrics

3.1. ASAM OpenDrive Import

Core component of any scenario is a map, with ReplicaR's feature of importing ASAM OpenDrive files, the map goes through a validation check in which its validity is checked as per AAI traffic flow and basic geometrical checks like continuity, overlaps etc. ReplicaR supports ASAM OpenDrive 1.4/5/6.

3.2. Scenario Generation via ReplicaR-Traffic

AAI has developed an extensive library of traffic scenarios based on naturalistic and verified mathematical algorithms (decision-making and motion models). This enables the platform to generate logs in simulation time (accelerated time frame) via ReplicaR GUI or ReplicaR Connect (which is programmatic control of ReplicaR technologies.

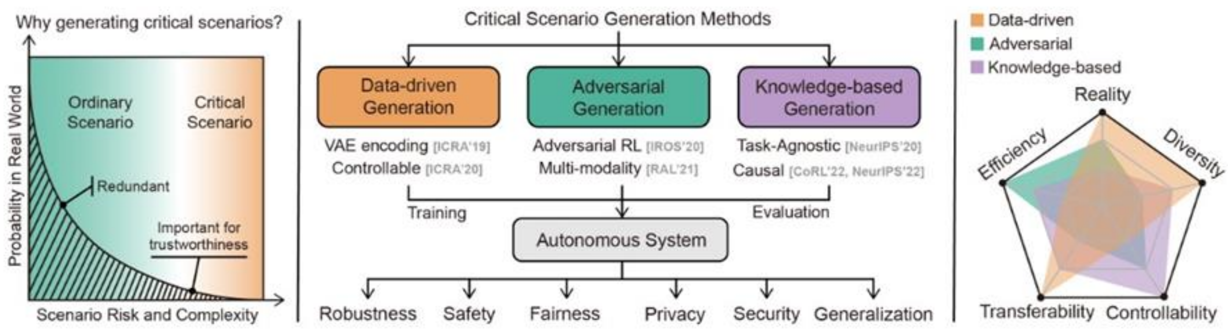

Figure 3 An overview of Scenario Generation methodologies [1]

3.2.1 Features

1. Support for highway and urban regions.
2. Configurable traffic density.
3. Configurable spawning strategies (lane-based, perception-based and everywhere).
4. Configurable multiple traffic agents (mathematically modeled, supervised, or reinforcement learning).
5. Configurable driver behavior as a measure of aggressiveness.
6. Configurable vehicle types (car, bus, van, etc.) and pedestrians.

3.2.2 Impact:

Running traffic simulations produces log files with hundreds of feasible trajectory profiles of traffic agents. These logs are then processed through a Critical Scenario Extraction Engine, which can generate unknown scenarios, a crucial step for scenario exploration.

3.2.3 Looking Forward:

Referring to Table 1, current traffic-based scenario generation lacks control over the "Definition Scale" (mentioned in the table) since creating traffic logs is not a direct manifestation of defining scenario abstraction. Research and development are ongoing to create a methodology linking traffic configuration with scenario abstraction. This will enable users to define functional level scenarios and employ traffic to create more critical scenarios, thereby improving the criticality score for traffic.

3.3. Scenario Generation via ReplicaR-Morphosynthesis

Knowledge-based statistical modeling for scenarios is the core of the Morphosynthesis module. Scenarios can be designed via the UI shown in Figure 4 or via ReplicaR Connect. This module is part of the MyScenario product, allowing for more controlled scenario creation on a selected map.

Figure 4 Intuitive User Interface for creating user defined scenarios

3.2.4 *Features:*

1. Scenarios can be created via the ReplicaR UI with a maximum of 4 Actors (scenario participants other than VUT), 2 Pedestrians, and Road Objects as obstacles.
2. Intuitive routing is provided through the ReplicaR UI, allowing users to define the route for all scenario participants.
3. The ReplicaR UI enables users to select not only the route but also a single lane change maneuver and single acceleration maneuver, with configuration details provided in the following points.
4. Speed can be configured through the ReplicaR UI within a range of 30 km/h delta.
5. The plausibility radius is set by the user to control the horizon of relevancy for the scenario.
6. All the parameters mentioned above can also be set in a JSON dictionary, which can be uploaded using the UI. Additional parameters include “lane change aggressiveness.”
7. The *Morphosynthesis* module uses these parameters to smartly reduce the number of possible scenarios to critical ones by employing measures like trajectory-based hazard analysis and kinematic feasibility.
8. If route information is missing in the uploaded scenario JSON, *Morphosynthesis* runs a validation on the given data and uses an internal routing algorithm to configure a valid route. Any invalid route halts the flow, and warnings are provided in the ReplicaR system logs.
9. The following parameters can be configured in a scenario:
   i Spawn point: Position as geo-coordinates or s-value, road ID, lane ID
   ii End point: Same as spawn point
   iii Lane change maneuver: Same as spawn point with direction and aggressiveness as additional parameters
   iv Acceleration maneuver: Same as spawn point with magnitude and applied duration as additional parameters.

3.2.5 *Impact*

Morphosynthesis, with its knowledge-based statistical modeling for scenarios, significantly enhances the capability of ReplicaR’s scenario generation. Here are the

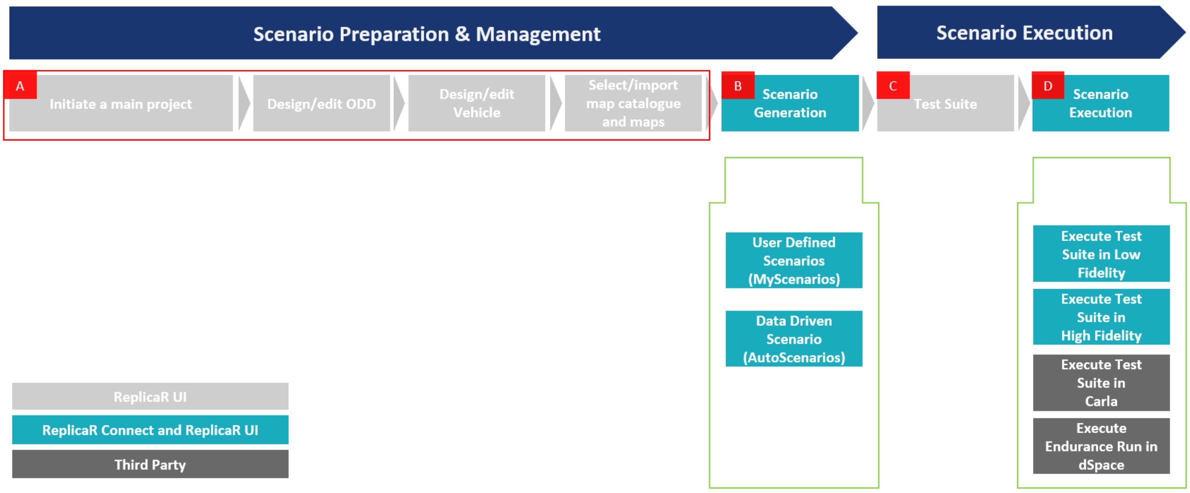

Figure 5 ReplicaR connect are SDK libraries to programmatically interact with ReplicaR platform to automate the scenario generation and execution process

key impacts:

1. *Morphosynthesis* enables the creation of detailed scenarios with minimal user input (unlike a trajectory). Users can easily configure parameters through an intuitive UI or by uploading a JSON dictionary, making the process straightforward and user-friendly.
2. The technique automatically reduces the vast number of possible scenarios to the most critical ones using trajectory-based hazard analysis and kinematic feasibility. This automation ensures that the scenarios generated are highly relevant for testing ADS functions.

3.2.6 Looking Forward

1. *Morphosynthesis* will be updated to have a more intuitive interface and upgrades on the Definition and Automation Scale as shown in Table 1 AAI Scenario generation methods and key evaluation metrics.
2. New features will include selection of Pegasus categorization (ISO-34502) as scenario intent, along with the spawn points for the SUT and traffic participants. This will empower *Morphosynthesis* module to capture the scenario intent in an intuitive and automated way along with capturing the higher Abstraction of Scenario Definition.

4. Using Scenario Generation via ReplicaR Connect

The ReplicaR Connect Python SDK empowers users to programmatically manage various simulation projects, including Low Fidelity, High Fidelity, My-Scenario, and Auto-Scenario. It provides comprehensive functionalities for authentication, vehicle management, test suite management, and co-simulation. The SDK facilitates seamless interaction with the ReplicaR Cloud, allowing users to retrieve, configure, and trigger simulations with selected environments and co-sim machines. Additionally, it supports advanced map management and offers robust monitoring capabilities to track the status and progress of applications. The ReplicaR Connect SDK is designed for users to enhance the efficiency and effectiveness of scenario generation and management in automotive simulation projects.

4.1. Scope of Enhancements in ReplicaR Connect for Simulation Engineers

Referring to Figure 5. following are the salient features:

A. *Requirement Creation:* For ODD, Vehicle, and Maps (using ReplicaR UI):

- i *Users can initiate a main project*
- ii *Design/edit ODD*
- iii *Design/edit Vehicle*
- iv *Select/import maps*

B. *Scenario Generation Technology:* Scenario generation, whether knowledge-based or data-driven, can be accessed via ReplicaR Connect.

C. *Test Suite Generation:* Test suite generation for execution should be analysis-based. ReplicaR provides tools for visualizing scenarios manually, allowing for the selection of relevant and plausible scenarios. This also ensures that even random selections without any visualization are possible.

D. *Scenario Execution Technology:* Scenario execution, whether object-list-based or physics/sensor-based, can be fully managed through ReplicaR Connect.

4.2. Features:

1. Authentication: Login to the ReplicaR Cloud.
2. Vehicle Management: Retrieve and select vehicles (Vehicle Model described in the MyVehicle application as VUT (Vehicle under test)).
3. Test Suite Management: Retrieve and select test suites (created by ReplicaR Scenario Generation Techniques).
4. Co-Simulation Management: Check availability of co-sim machines which are running ReplicaR Sensor Simulation Clients.
5. High Fidelity Applications: Retrieve and manage high fidelity projects (managing configurations to trigger Sensor Simulation). Trigger simulations with selected environment and co-sim machines. Monitor the status and progress of execution loops with SUT.

6. Low Fidelity Applications: Retrieve and trigger low fidelity projects (Object list-based Simulation).
7. Map Management: Retrieve and manage maps. Update map settings.
8. My-Scenario Application: Retrieve and manage MyScenario projects. Trigger scenarios with selected parameters. Monitor the status and progress of MyScenario projects (use of Morphosynthesis).
9. Auto-Scenario Application: Retrieve and manage AutoScenario projects. Trigger scenarios with selected parameters. Monitor the status and progress of AutoScenario projects (use of Naturalistic Traffic).

4.3. Impact:

The ReplicaR SDK facilitates seamless interaction with the ReplicaR Cloud, enabling efficient scenario generation and management through Python. It supports comprehensive features for authentication, vehicle management, test suite management, co-simulation, and both high and low fidelity applications. The SDK's ability to handle various application types, including MyScenario and AutoScenario, provides a robust framework for simulation and testing in automotive scenarios. This versatility greatly enhances the ability to develop, test, and validate scenarios in a controlled and automated manner.

5. ReplicaR-Carla Co-simulation

ReplicaR provides a co-simulation framework for the Carla engine which enables users to execute or replay any scenario extracted using ReplicaR, for this Carla-cosim client can be downloaded from ReplicaR platform.

5.1. Methodology:

Carla is a widely-used framework and sensor simulation engine designed for closed-loop simulations. It is complemented by ReplicaR, which allows users to execute scenarios created within the ReplicaR platform using the Carla Simulation Engine. This integration leverages Carla's dockerized form or manual installation, making it versatile for different user setups. Scenarios composed of Open Simulation Interface (OSI) data and an OpenDRIVE map (both adhering to ASAM standards) are loaded by the ReplicaR co-simulation (co-sim) client. This client orchestrates the scenario within Carla, offering a range of settings and controls as outlined below:

1. **Carla Folder:** Specifies the location of the Carla installation folder (if installed manually).
2. **Scenario Package:** Path to the downloaded scenario zip file from the ReplicaR platform.
3. **Scenario Params:** Opens the Scenario Configuration page, allowing users to modify scenario parameters.
4. **World Settings:** Opens the World Settings page for configuring environmental aspects.
5. **Initialize:** Loads and spawns the scenario agents in the Carla world, preparing them for execution.
6. **Execute:** Runs or replays the loaded scenario.
7. **Initialize Ego:** Option to include or exclude the Ego vehicle in the scenario. The default setting includes the Ego vehicle.
8. **RGB Camera Position:** Provides default positions for Carla cameras. Users can configure the sensor setup independently of the co-sim client.
9. **Attach Collision Sensor:** Enables collision detection during the Carla simulation.
10. **Weather:** Allows setting various environmental parameters provided by Carla.
11. **Network Setting:** Specifies the host address and port for the Carla Engine. If “Execute with Docker” is checked, the Carla Folder is not required, and the co-sim client will wait for Carla to be manually initiated.
12. **World Generation Settings:** If the scenario package map is to be used, ensure the “Use Scenario package map” box is checked (default). Unchecking this box allows for loading a custom map in Carla.

5.2. Future Enhancements:

1. Integration with intelligent traffic module which will enable the use of naturalistic traffic participants developed by AAI.
2. Loading 3D scenes generated by ReplicaR (enhanced by Map-Editor and Scene-Editor) in Carla as World-Scene.

6. Scenario Execution with ReplicaR

The execution of scenarios within the ReplicaR platform encompasses both object list simulation and high-fidelity sensor simulation, developed on the Unity platform. This capability allows for the seamless integration of the System Under Test (SUT) with ReplicaR, facilitating both open-loop and closed-loop scenario executions. These executions can be managed via ROS2 interfaces or OSI3 messages, though the latter is not currently exposed.

6.1. Initialization Services:

1. ROS2 services based.
2. OSI3 messages based.

6.2. Object List Simulation Interface

The object list simulation interface in ReplicaR involves several critical components:

1. Map Interface: Maps are provided in the OpenDRIVE format, ensuring standardized and detailed map data for simulations.
2. Data Interface: The data interface employs any customer specific messages. Currently ReplicaR supports all well-established exchange interface for ROS, facilitating robust data handling during simulations.

3.2.7 Looking Forward

1. Exposure of OSI3 Interface: Enabling existing closed-loop object list-based simulations with OSI3.
2. Phenomenological Sensor Modeling: Integrating phenomenological sensor models (e.g., camera-based, lidar-based, radar-based) will provide more accurate and detailed sensor data, improving the fidelity of the simulations.

6.3. Sensor Simulation Interface

1. Map Handling: Like the object list interface, maps are provided in the OpenDRIVE format.
2. Data Interface: The data interface currently supports open-loop simulation and plans to include radar scans and radar tracks for more comprehensive sensor simulations.

3.2.8 Looking Forward

1. Closed-Loop Simulation: Transitioning to closed-loop simulations will enable more dynamic and responsive scenario executions.
2. Radar Sensor Simulation: Incorporating radar sensor simulations, including radar scans and tracks, will enhance sensor coverage by ReplicaR.

Figure 6 A snippet of sensor simulation in AAI ReplicaR, covering all aspects of simulation-based testing, from synthetic environment creation to realistic traffic simulation.

7. Synthetic Data for Open-Loop

Synthetic data generation for open-loop validation is one of the primary use cases that ReplicaR is designed to support. During each execution run, users have the option to generate extensive synthetic simulation data directly from ReplicaR. This feature enables the creation of highly detailed datasets that include billions of unique combinations of vehicles and pedestrians. Additionally, ReplicaR can simulate a wide range of environmental conditions, incorporating various weather scenarios and different times of day. Figures 7 and 8 illustrate the comprehensive capabilities of ReplicaR in producing this synthetic data, highlighting its potential for diverse and complex simulation environments.

7.1. Interface

1. Data available as ROS2 mcap file with topics mentioned.
2. Data available as OSI3SensorData (according to SETLEVEL).

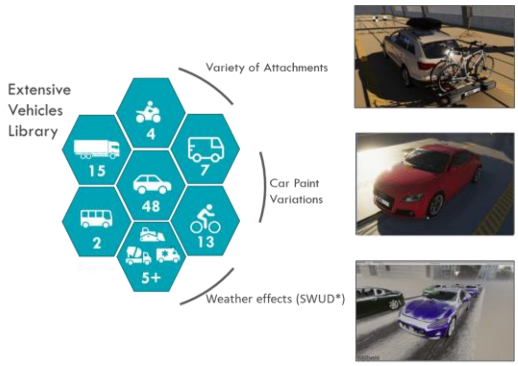

Figure 7 Vehicle variations in ReplicaR (SWUD = Snow, Wetness, Use, Dirt)

7.2. Looking Forward

1. Ability to drive any listed vehicle from ReplicaR's portfolio of VUT in ReplicaR generated environment.
2. Sensor Level Annotation (including but not limited to instance segmentation and classification of 'Road Artifacts' and 'Traffic participants').
3. Provision of data in user required specifications.

8. AAI Versatile (Map and Scene Editing Tool)

AAI offers a tool known as Versatile [8], which natively supports the creation and editing of OpenDRIVE maps. Versatile is intuitive and ensures validation checks needed for ASAM OpenDRIVE compliance. The map can be converted into a 3D scene, and any additions in the 3D scene are reflected to the OpenDRIVE map.

9. Conclusion

The AAI ReplicaR platform is an indispensable tool for the V&V of autonomous driving systems. Its comprehensive scenario generation capabilities, detailed simulation execution, and advanced data generation

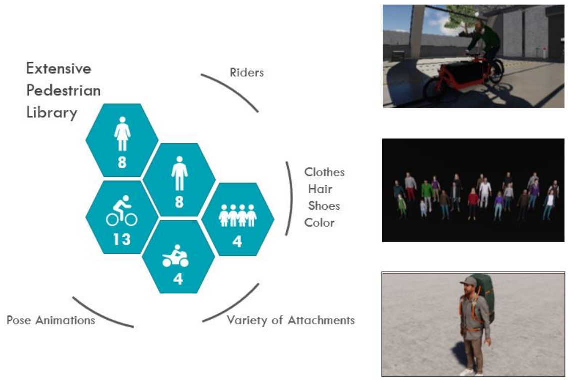

Figure 8 Pedestrian variations in ReplicaR for synthetic data

features ensure ADS meet stringent safety and regulatory standards. The platform's support for Software-in-the-Loop (SIL) testing and seamless integration with popular simulation tools like Carla further enhances its utility. As the platform evolves, it will continue to provide sophisticated tools, ensuring the reliability and safety of autonomous vehicles in real-world conditions.

References

[1] 1001 Ways of Scenario Generation for Testing of Self-driving Cars: A Survey, https://arxiv.org/pdf/2304.10850
Request access to the white paper, contact info@automotive-ai.com

[2] Pegasus Scenario Description, https://www.pegasusprojekt.de/files/tmpl/PDF-Symposium/04ScenarioDescription.pdf

[3] Critical Scenario Generation for Developing Trustworthy Autonomy, https://arxiv.org/pdf/2305.00339

[4] Pegasus Method, https://www.pegasusprojekt.de/en/pegasus-method

[5] replicar-lite traffic library, https://github.com/automotive-ai/replicar-lite

[6] OSI3, https://opensimulationinterface.github.io/open-simulation-interface/

[7] Case Study Automotive Artificial Intelligence (AAI) | TÜV SÜD (tuvsud.com)

[8] AAI Versatile (youtube.com)