GUIDE FOR CREATION OF SIMULATION HANDBOOK

Sim4CAMSens project deliverable 3.4

1. Overview

The aim of this guide is to outline the process and expected content of the simulation handbook for a perception sensor model. The guidelines for the creation of a sensor model simulation handbook are based on current interpretations of draft and existing legislation relating to Advanced Driver Assistance Systems (ADAS) and Automated Driving Systems (ADS). The following documents from the UNECE Working Party on Automated/Autonomous and Connected Vehicles (GRVA) have been referred to:

- https://unece.org/sites/default/files/2024-01/GRVA-18-50e_1.pdf
- https://wiki.unece.org/download/attachments/286097494/ADS-10-05.pdf?api=v2

Therefore, the goal of a simulation handbook for a sensor model is to establish the credibility of the sensor model in the context of the simulator that it is used in. It is expected that similar technical documents will be produced for all other aspects of the simulation toolchain to satisfy the credibility requirements.

According to GRVA-18-50e, it is recommended that credibility is achieved by investigating and assessing five Modelling and Simulation (M&S) properties:

1. Capability – what the M&S can do, and what are the associated risks.
2. Accuracy – how well M&S does reproduce the target data.
3. Correctness – how sound & robust is the M&S data and the algorithms in the tools.
4. Usability – what training and experience is needed and what is the quality of the process that manage its use.
5. Fit for Purpose – how suitable is the M&S toolchain for the assessment of the ADS within its Operational Design Doman (ODD).

However, not all of these criteria are appropriate to a sensor model, so we determine that the simulation credibility for a sensor model is established through 3 main criteria:

1. Documentation of the computational model used in the sensor model including its limitations and uncertainties – determines capability and fit for purpose
2. A set of unit tests that demonstrate the sensor model works as documented – determines correctness
3. Validation report showing the sensor model performance compared to real world test data – determines accuracy

The simulation handbook should be delivered to the customer. It is expected that this level of documentation forms part of a qualification kit for the complete simulation environment and not supplied as part of the basic license. It is also assumed that this is additional documentation and should be used together with the user guide documentation.

2. Contents

3. Document History

Date	Version	Comment
August 2025	1	Initial version

4. About the Sim4CAMSens project

Sim4CAMSens was a CCAV funded collaborative R&D project that developed methods to quantify and simulate camera, radar and lidar sensor performance under all conditions. It brought together simulation companies (rFpro, Claytex), systems engineers (Syselek), sensor developers (Oxford RF) and research and academic institutions (NPL, WMG and the Compound Semiconductor Applications Catapult) on a project coordinated by AESIN (led by Claytex) to investigate perception sensor modelling and simulation.

The project was part of CCAV's Commercialising CAM Supply Chain Competition (CCAMSC).

The Commercialising CAM programme is funded by the Centre for Connected and Automated Vehicles, a joint unit between the Department for Business and Trade (DBT) and the Department for Transport (DfT) and delivered in partnership with Innovate UK and Zenzic.

The CCAM Supply Chain competition was launched in October 2022 to support the delivery of early commercialisable Connected and Automated Mobility technologies, products and services and is part of the Government's vision for self-driving vehicles. Connected and automated mobility 2025: realising the benefits of self-driving vehicles.

5. Document Structure

The document is expected to be used alongside the standard users guide and have the following sections:

1. Overview of the real sensor
2. Overview of the sensor model
3. Breakdown and details of the sensor model behaviour
4. Validation of the sensor model

If the information required in these sections is already provided in the standard users guide then a reference to the appropriate document should be included. The information must be covered in either the standard users guide or this simulation handbook.

Section 1 must provide a description of the real sensor and include references to the supplier and documentation provided by that supplier. It should clearly identify the key principles involved in how the sensor works e.g. mechanically spinning 360-degree TOF lidar at 905nm or FMCW radar at 79-81GHz.

Section 2 must provide an overview of the sensor model and identify the main features of the sensor model that need to be documented, verified and validated in Section 3. This overview of the sensor model must identify the computational steps in the model, ideally with a diagram to help explain and illustrate the process. The following key points must be covered:

- How energy is transmitted into the environment and received back at the detector
  - Propagation of energy through the atmosphere
  - Reflection of energy off objects
- Details of noise factors and how they affect the energy propagation/reflection
- Details of any signal processing steps and documentation on the output format of the sensor model

Section 3 must cover each aspect of the sensor model behaviour. For all sensor models there are many complex and interdependent behaviours that are being modelled. Each of these should be described, verified and where possible validated. A number of these sub-sections will be covering some of the fundamental physics being simulated for multiple sensors and may therefore be separated into multiple simulation handbooks that are referenced by each specific sensor model. See

Documenting the Computational Model Guidelines for detailed guidance on how to structure these sub-sections.

Section 4 must cover the validation of the complete sensor model against real world test data. This section is needed to prove that the sensor model as a whole works correctly and that the interaction of all the features in the sensor model add up to correct overall behaviour. See ASSESSING THE COMPLETE SENSOR MODEL.

5.1. Documenting the Computational Model Guidelines

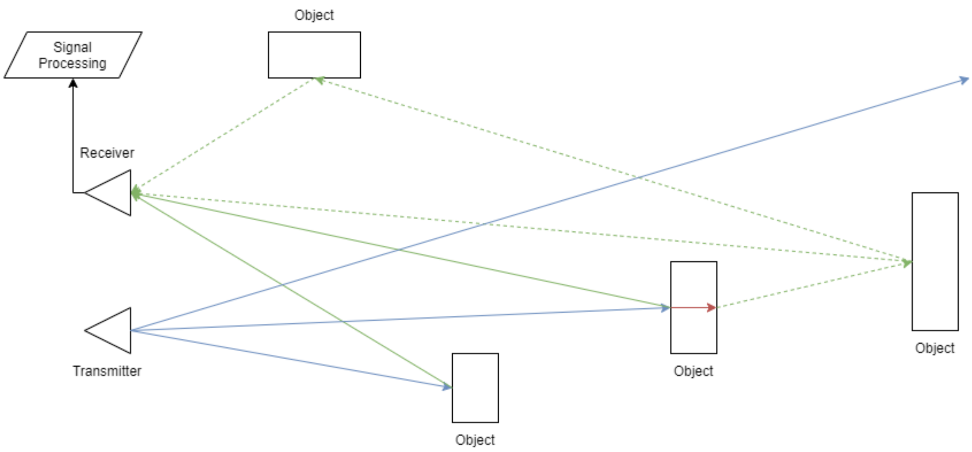
Each section that covers a feature of the sensor computation model should include:

- A general description of the feature. It might be appropriate to refer to the standard product documentation for this feature but in most cases it is expected that more detail is included in the simulation handbook
- Documentation of the computational model including any assumptions made in the implementation and references to any papers/books used to provide the theory
- Documentation and description of the effect of any user controllable parameters including typical values and valid ranges. This might be covered in the users guide.
- Definition of the unit tests that prove the model matches the computational model (verification), see VERIFICATION GUIDELINES
- Detailed validation against real world test data, see VALIDATION GUIDELINES
- The uncertainties, limitations and valid ranges of operation for each feature, see RANGE OF VALIDITY, LIMITATIONS AND UNCERTAINTIES

For each feature, the documentation should describe how it works and include a step-by-step guide to the mathematical model including relevant formula. The following example documentation is for a generic ray traced sensor to use as further guidance on how to structure this section.

Example Documentation for a Computation Model

The generic sensor uses ray tracing to determine how energy propagates through the environment from the transmitter to the receiver. The diagram below shows the overall scheme of how the computational model works and the steps involved.



1. At the transmitter, the model determines the number of rays to be transmitted, their starting point and direction of travel. It also determines the amount of energy in Joules that should be associated with the origin of each ray (blue arrows).
2. For every ray that propagates through the environment the attenuation of energy is calculated taking into account the following atmospheric models:
    a. Propagation through the atmosphere under ideal conditions will result in energy loss described in XX.
    b. The rain model will calculate the energy loss and determine if there are any reflections off rain drops as described in XX.
3. When a ray hits an object some energy might be reflected (green lines) or transmitted (red lines) through the object as described in XX.
4. For every ray that is reflected, or transmitted after passing through an object, steps 2 and 3 should be repeated until the rays reach the receiver or are lost to the environment (green dotted lines). When the power associated with the origin of a ray drops below a threshold of XX Joules the ray will be discarded.
5. At the receiver the received rays are collated and an overall power distribution function is calculated as described in XX.
6. The power distribution function is then passed to the signal processing block which calculates and produces the output messages from the sensor as described in XX.

5.2. Verification Guidelines

Verification in this context means that unit tests are carried out to demonstrate that the implementation in the sensor model matches the described computational model under the relevant range of conditions. Results should be compared against the theory used and any references to standards or publications should be included.

It is expected that each feature will require several different unit tests to verify that the sensor model behaviour matches the documented computational model. The unit tests should cover the valid range of parameterisation for the model.

The setup of the unit tests using the supplied simulation toolchain should be described so that the customer is able to recreate the tests using the delivered sensor model and simulation toolchain. The expected results of the unit tests should be included.

If the test results show any discrepancies to the those expected from the documentation of the computational model, these should be explained.

The unit tests should demonstrate that the simulation can be deterministic even if the default operating mode is to be non-deterministic to allow the simulation of stochastic noise factors.

5.3. Validation Guidelines

Validation in this context means a comparison between the simulation data and real world test data to determine that the sensor model is sufficiently accurate. Where possible, validation of each part of the computational model should be done using the simplest possible test that demonstrates the behaviour being validated. If validation of an individual feature is not possible, then the simplest possible test that covers multiple features should be used to validate the behaviour. The validation should then continue to build on validated features, adding more complexity, to demonstrate the validation of the complete sensor model.

Validation of every feature may not be possible as some processes may be near-impossible to replicate due to chaotic or stochastic effects (such as rain on a droplet-by-droplet basis). In these cases, an alternative description of the physical effect will have been implemented and the validation section must demonstrate why this implementation is appropriate.

The validation report for each feature should include a description of the test setup so that a customer could recreate the validation tests. The report should detail the metrics used to demonstrate that the sensor model matches the real-world data and include evidence of why the accuracy bounds used to establish the validity of the model are reasonable and appropriate. A definition of some metrics that might be appropriate for comparing different types of data are included in VALIDATION METRICS.

5.4. Range of validity, limitations and uncertainties

An important detail when modelling real-world physics is that the software allows the implementation and testing of scenarios which are not possible in reality. These unrealistic use cases can often give insights into the underlying physics but clearly do not represent a realistic use case.

In this step, the best- and worst-case extremes of the operating conditions for the sensor model are to be explained. This includes describing the robustness and stability of parameter ranges, which dictate whether small changes in a parameter cause large or hard-to-predict changes to the overall simulation scenario.

Parameter ranges which are suitable for normal use cases, and extended to testing use cases are therefore provided, along with parameter combinations which may provide unphysical behaviour.

Both aleatory and epistemic uncertainties associated with the feature should be described and characterised. Within the relevant regulations it is required that the uncertainties within the models are documented and that their propagation within the model and through the simulation toolchain can be understood and quantified.

5.5. Assessing The Complete Sensor Model

The section that covers the validation of the compete sensor model must cover the following:

- Definition of the unit tests that demonstrate the complete sensor model behaviour
- Detailed validation against real world test data, see Validation Guidelines
- The uncertainties (aleatory and epistemic), limitations and valid ranges of operation, see Range of validity, limitations and uncertainties

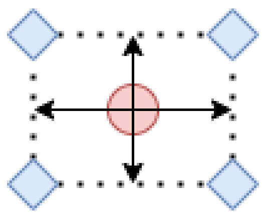
The unit tests for the complete sensor model should demonstrate that the sensor model works as expected. Whilst it is not expected that the unit tests for the complete sensor model can be matched against theoretical calculations, a set of repeatable tests are required that demonstrate the complete sensor model behaviour over the expected valid range of operation. This section should, therefore, also define the valid range of operation for the sensor model.

6. Validation Metrics

Within this set of guidelines it is not feasible to specify which validation metrics are appropriate for every type of data. The most appropriate validation metric for each variable will depend on the type and nature of the data. The following guidelines are to help understand some of the most common metrics that could be used.

6.1. Time varying data

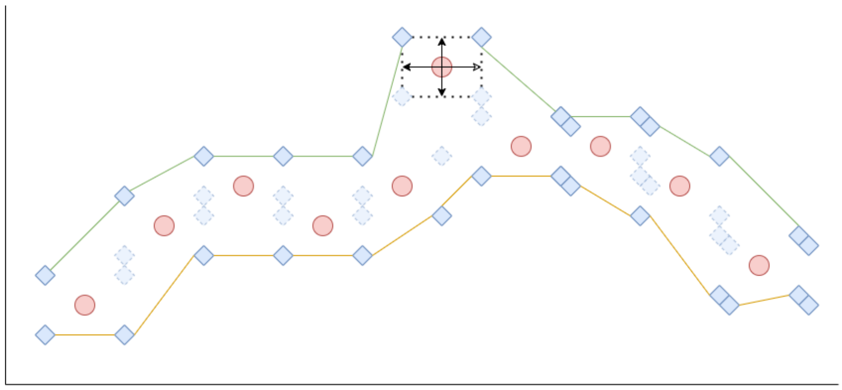
Comparison of time varying data can be done by constructing a tube from the reference data and then checking that the result data lies within this tube. There are many different ways that you could determine the upper and lower bounds of the tube and you will need to include a justification for the method used within the validation discussion.



One approach to constructing the tube is that for every point you apply x and y offsets to create 4 boundary points around the data point. In the figure to the right, the data point is the red circle and the 4 boundary points are the blue diamonds. The offsets would usually be an absolute value. For high frequency sampled data an appropriate x offset might be half the time between samples. The magnitude of the y offset depends on the data and knowledge of the expected spread and uncertainty in the measurement data. As a result, the y offset may not be constant across the measurement range.

After constructing boundary points around all of the data values, points are selected to define the upper and lower tube boundaries. This should usually result in many of the boundary points being discarded as shown in the figure below where the green line forms the upper boundary between boundary points and the orange line is the lower boundary. The data that is being compared should now be checked to make sure that it is entirely within the tube boundaries.

A simplification of this approach that might be better for low frequency data is to only use the y offset to construct the tube boundaries.



6.2. Noisy data

A good approach for comparing noisy data where there is an expected distribution of values for the same measurement is referred to as the double validation metric[1]. In this

[1] Rosenberger: *Metrics for Specification, Validation, and Uncertainty Prediction for Credibility in Simulation of Active Perception Sensor Systems*, PhD Thesis, TU Darmstadt, Darmstadt, Germany, 2023

approach both the bias and distribution spread for the reference and comparison data are calculated and compared and allow a quantification of how close the distributions match between the two sets of data.

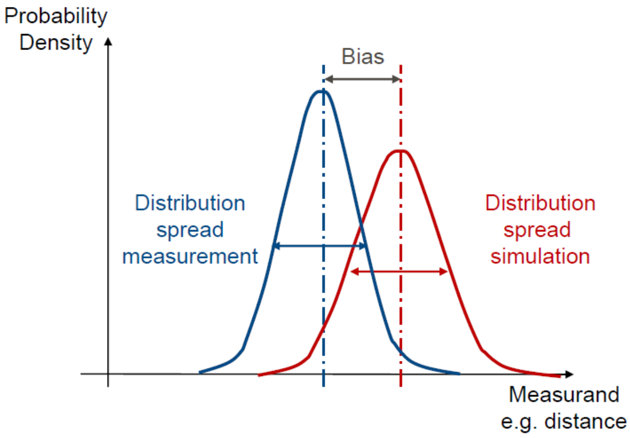


Image source: Presentation at AVT Stuttgart 2025 by Clemens Linnhoff, Persival GmbH

6.3. Point cloud data

An approach proposed by Chan et al[2] is a holistic approach to quantify the quality of collected and/or simulated pointclouds, combining several new and existing pointcloud metrics:

(i) a newly proposed 'dissimilarity metric', hereby named Shahbeigi-Donzella Metric (SDM), as a robust bounded pointcloud metric;
(ii) six adapted geometrical pointcloud metrics, e.g. Hausdorff distance (HD), Root-mean-square deviation (RMSE), etc.;

TABLE I
HIGH LEVEL COMPARISON OF DIFFERENT PCQA METRICS SELECTED IN THIS WORK

Metric	Importance	Pros	Cons
SDM	PC shape signature and distance between signatures	View and size invariant	Can be computationally intensive
RMSE	Root Mean Square Error (RMSE) point-to-point difference	Overall assessment of differences	Needs PCs of same size
MDE	Euclidean distances between all nearest neighbour pairs	More robust to outliers than RMSE	More focused on spatial similarity
MMDE	Emphasise the maximum average distance between point pairs	Emphasise detection of anomalies	Sensitivity to Large Differences
HD	Maximum distance error of corresponding points	Anomaly Detection	May introduce redundancy
HD95	Similar to HD but selects the 95% percentile distance	Robust Outlier Handling	May affect the sensitivity to certain outliers

[2] Pak Hung Chan, Daniel Gummadi, Abu Mohammed Raisuddin, et al. LIDAR De-Snow Score (DSS): combining quality and perception metrics for optimised data filtering. *TechRxiv.* April 08, 2024.
DOI: 10.36227/techrxiv.171259681.14859329/v1

6.4. Camera Images

A comparison of real and simulated camera images is complicated due to the interaction of many different factors that affect exactly what image is captured. Pixel-to-pixel correlation is not feasible due to the shear number of variables that would all have to be aligned within the vehicle dynamics model, world model, camera model and ISP. Therefore a number of other metrics should be used to compare the outputs from real and simulated cameras to demonstrate that the image quality produced by both is comparable and appropriate.

A large range of image quality metrics exist but these methods have focused on evaluating or trying to quantify how realistic and satisfying an image is for human consumption. For ADAS and ADS, the images are fed to a perception algorithm and so we need to select metrics that are best suited to quantifying quality for this purpose.

TABLE I
SUMMARY OF IQA METRICS - FULL-REFERENCE (1-14) AND NO-REFERENCE (15-17). KEY FEATURES EXTRACTED BY METRICS: LM = LOCAL MEAN, LC = LOCAL CONTRAST, EG = EDGE GRADIENTS, CH = CHROMINANCE, VS = VISUAL SALIENCE, LBT = LOCAL BLOCK THRESHOLDS. UP ARROW: HIGH VALUE = GOOD IMAGE QUALITY, DOWN ARROW: LOW VALUE = GOOD IMAGE QUALITY.

	Frequency Features		Spatial Features					
IQA Metric:	Fourier	Wavelet	LM	LC	EG	CH	VS	LBT
1) PSNR↑								
2) SSIM↑			✓	✓				
3) CW-SSIM↑		✓						
4) DSS↑	✓							
5) HAAR-PSI↑		✓						
6) VIF↑	✓							
7) VSI↑					✓	✓	✓	
8) MS-SSIM↑			✓	✓				
9) SR-SIM↑					✓		✓	
10) IW-SSIM↑			✓	✓				
11) FSIM↑	✓				✓	✓		
12) GMSD↓					✓			
13) MS-GMSD↓					✓			
14) MDSI↓					✓	✓		
15) BRISQUE↓			✓	✓				
16) NIQE↓			✓	✓				
17) PIQUE↓				✓				✓

Gummadi et al[3] have analysed a range of metrics and proposed a subset that are most appropriate for assessing image quality for perception algorithms. The results demonstrate a strong correlation between most of the traditional IQA metrics and the object detector (i.e. a Faster R-CNN) performance across various compression levels. Notably, metrics like IW-SSIM and retrained BRISQUE exhibited extremely high positive correlations.

[3] Daniel Gummadi , Pak Hung Chan , Hetian Wang , et al. Correlating traditional image quality metrics and DNN-based object detection: a case study with compressed camera data. *TechRxiv.* November 22, 2023. **DOI:** 10.36227/techrxiv.24566371.v1