



**HAL**  
open science

## Simulation of weather conditions perturbation for an automotive LIDAR sensor

Sylvestre Prabakaran, Youri Mikhail Noutatiem Guiafaing, Ammar Ridzuan, Tchoya Florence Koné, Otmane Attou

► **To cite this version:**

Sylvestre Prabakaran, Youri Mikhail Noutatiem Guiafaing, Ammar Ridzuan, Tchoya Florence Koné, Otmane Attou. Simulation of weather conditions perturbation for an automotive LIDAR sensor. Driving Simulation Conference (DSC), Sep 2024, Strasbourg, France. hal-04720367

**HAL Id: hal-04720367**

**<https://hal.science/hal-04720367v1>**

Submitted on 4 Oct 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Copyright

# Simulation of weather conditions perturbation for an automotive LiDAR sensor

Sylvestre Prabakaran<sup>1,2</sup>, Youri Noutatien<sup>2</sup>, Ammar Ridzuan<sup>1,2</sup>, Tchoya-Florence Kone<sup>3,2</sup>, Otmane Attou<sup>3,2</sup>

(1) AVSimulation, 1 Cr de l'île Seguin, 92100 Boulogne-Billancourt, France, e-mail: {sylvestre.prabakaran, ammar.ridzuan}@avsimulation.com

(2) IRT SystemX, 2 Bd Thomas Gobert, 91120 Palaiseau, France, e-mail: {sylvestre.prabakaran, youri.noutatien, tchoya-florence.kone, ammar.ridzuan}@irt-systemx.fr

(3) Valeo, 6 rue Daniel Costantini, 94000 Créteil, France, e-mail: {tchoya-florence.kone, otmane.attou}@valeo.com

**Abstract** – Perception algorithms based on LiDAR sensors are more and more widely used in intelligent transport systems, such as ADAS and autonomous vehicles. As new technological advancements evolve, industrial LiDAR sensors are becoming much more affordable and equipped on modern vehicles. Despite their accuracy for mapping the environment and estimating precise distances, the operation of LiDAR is known to be affected by adverse weather conditions, thus reducing the operational design domain of vehicles equipped with this type of sensor. Tests in simulation of intelligent transport systems have been widely used in the industry throughout the whole cycle of development, to accelerate R&D and reduce time to market costs. It is thus important to improve the virtual models of LiDAR sensors to ensure the reproduction of realistic and adverse weather conditions in simulation test. This is the aim of project CVH (Tool-based chain for the validation and certification of the connected automated vehicle). This paper introduces a methodology of acquiring real world data and through data analysing and processing be able to create and improve the current model of perfect sensors by implementing the effects of perturbations, whether it be hardware or weather caused.

**Keywords:** LiDAR, Fog, Adverse Weather, Simulation, AD/ADAS

## Introduction

LiDAR (Laser Imaging Detection And Ranging) technology is well known as a perception technology which is essential for vehicles automatization. Even though this sensor is considered to be more accurate and robust than other sensors such as radars and cameras, LiDAR sensors also have limitations caused by the complexity of the environment they are working in. That is why there is a need to validate the sensors. Simulation is known to be a reliable solution in order to validate the future autonomous vehicles and therefore, LiDAR sensors technology must too be modelled in simulation.

Several ray tracing models exist for the virtual model of LiDARs, however perturbation effects on these models have yet to be well implemented in order to have a realistic enough simulation. This is the aim of the research project CVH (in French: Chaîne outillée pour la Validation et l'Homologation des véhicules automatisés) at IRT SystemX. Amongst the goal of this project is the modelling of the behaviours of sensors when confronted with perturbations. Several tools and sensor models exist and are already well known in the automotive simulation industry but most of them are modelled as perfect sensors without taking into account error and disturbances due to perturbation.

The aim of this paper is to present a model that has been developed and used with AVSimulation's SCANer Studio software, with the use of a specific semi-rotative LiDAR.

This overall methodology has been applied and implemented as a Proof of Concept for fog simulations. This paper summarizes the scientific reasoning which has been made in order then to generalize it for other use cases of modelling automotive sensors, as well as other types of adverse weather conditions.

## Overview of LiDAR sensor modeling

### LiDAR Technology in the Automotive Industry

LiDAR is a sensor that is used to measure distance of points in the environment. In the automotive field, it is used as a 3D Cartography sensor that can detect any object in the environment with an accurate estimation of the distance. By sending laser rays in several directions, a LiDAR is able to output a 3D point cloud representing the environment around the vehicle.

## Types of LiDARs

Different technologies are used depending on the sensor, but they all rely on having laser ray emitter, a directing device (mostly mirrors that can aim at given angles) and receivers that receive the emitted rays that have been reflected by the environment. By computing the delay between the emission and the reception, LiDAR is able to estimate the distance.

The main types of LiDARs that are used in the automotive industry are:

- Rotative LiDAR: A motor is used in order to mechanically rotate the laser emission beam to scan with a specific pattern in every direction.
- Semi-rotative LiDAR: Electro-mechanical parts (such as mirrors or MEMS) are used in order to scan the environment in different directions. Usually, the field of view of this LiDAR is restricted to specific angles.
- Solid-state LiDAR: A laser beam is sent with specific phases and patterns in order to broadcast rays in a given direction.

## Sensor outputs

Outputs of LiDAR sensors can be described as two layers.

The first layer is a raw data represented as a three-dimensional point cloud. Whatever the type of technology that is used in the LiDAR, the sensor is able to output a 3D point cloud of the environment. For each specific direction of a laser ray, there will be one or several points (in the case of multi echo) that will be added to the point cloud. The receivers will quantify the attenuation of the ray between the emission and the reception of the laser ray, this measurement leads to an intensity value for each point of the point cloud. This intensity depends on the reflectivity of the target's material, but also on the incident angle of the ray, the distance, etc. This intensity data gives more information on the type of detection for the software to recognize the object in the environment.

The second layer is optional but for some LiDARs, a detection software layer can be added, for object detection and classification. That is why modeling the perturbation in simulation can also be done directly in this second layer of outputs which is the detections.

## Simulation tools

### The need for simulation tools

Simulating LiDARs in an automotive environment relies on using tools that reproduce three characteristics:

- The environment in which the vehicle is driving: the terrain comprising the road, the infrastructures and its condition has to be modelled.

- The actors such as the ego vehicle, the surrounding cars and pedestrians: they must be described including their geometry, materials and positions during the scenario.
- The sensors: The sensor model, in our case the LiDAR, is implemented regarding its position in reference to the vehicle, and its defining parameters (laser emission patterns, rotation frequency, field of views, power, etc.)

## Existing tools for Lidar modelling

Most sensor simulation tools use ray-casting/ray-tracing techniques or projection methods such as "Z-buffer" to create synthetic point clouds. These works have been previously described and compared in (Rosenberger, 2019). Virtual rays are drawn between the emitter location and the open space of the geometrically represented virtual environment. If an intersection with any surface is detected, a point is created and reflected intensity associated to it will be related to the range, the incidence angle and to surface characteristics, described by their bidirectional reflectance distribution function (BRDF).

Among the simulation tools used to simulate LiDARs are for example: Unity, Unreal, IPG Carmaker, Gazebo, Pro-SiVIC, SCANer Studio

In this work, SCANer Studio is used for LiDAR modelling. This is executed using an integrated module called LASERMETER that will be described after.

## Perturbations

Numerous noise factors can affect the operation of a LiDAR sensor. (Chan, Dhadyalla, & Donzella, 2020) proposes a list of 16 factors, grouped in 5 categories. They can be related to interactions with others systems (coexistence with other lidars, malicious attacks), to degradation over time of electronic or mechanical components, among many factors. Particularly, adverse weather conditions, such as rain, snow, and fog, have a pronounced impact on point clouds. The presence of airborne particles (e.g. snow flake or rain drop) may impact the way LiDAR signal propagates on the atmosphere, due to absorption, reflection and scattering phenomena. Likewise, dry and wet surfaces interact differently to incident lidar rays. As a result, when operating in adverse weather conditions, such as rain, snow or fog, lidar point clouds are disturbed on the number of detected points, their intensity, as well as on the appearance of noisy points (Zhang, Ang, & Rus, 2018)

Several research groups have studied adverse weather effects on lidar sensors. (Filgueira, Gonzales-Jorge, Lagüela, Diaz-Vilariño, & Arias, 2017) conducted outdoors experiments with a 360°, 16-layers lidar and were able to collect data for different rain intensities. Through the analysis of

segmented portions of point clouds, they concluded that rainy conditions cause losses in the number of points and the attenuation of the returning signal. (Kutilla, Pyykönen, Holzhüter, Colomb, & Duthon, 2018) collected data in a controlled environment, able to produce artificial rain and fog. In particular they compared two distinct wavelengths (905nm and 1550nm). They observed that lidar signal intensity decreases lightly in the presence of rain and strongly in the presence of intense fog (< 40 meters of visibility).

Operation of perception systems based on lidar sensors embedded on intelligent vehicles is, as a consequence, hindered by adverse weather conditions. It is important to reproduce such behaviours in simulation sensors and render tests in virtual environments more realistic and reliable.

Many authors have proposed models to noise factors related to the presence of airborne particles in the atmosphere. (Goodin, Carruth, Doude, & Hudson, 2019) proposed a mathematical model that introduces noise in the range measurement and reduces the intensity of points as a function of the scattering coefficient and the rainfall rate. (Byeon & Yoon, 2021) adopted a microscopic approach and incorporated into their model raindrop characteristics specific to different regions in the globe, such as size and shape. (Espineira, Robinson, Groenewald, Pak, & Valentina, 2011) developed their lidar model using a ray cast method based on the Unreal Engine that adds false positive points to the point cloud, as a simulation of backscattering effects related to rain.

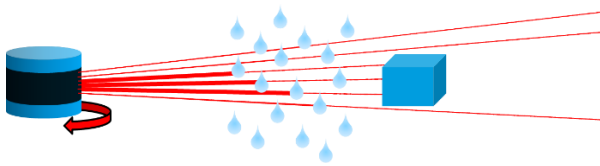


Figure 1: A simplified view of why perturbation affect LiDAR sensor

## Simulation model improvement methodology

The methodology that is presented in this paper relies on using a simulation tool with its existing models, and then improving it in order to consider lacking phenomenon, especially perturbations. The existing perturbations and the interferences with LiDAR's outputs are obtained after performing real data's measurement with our LiDAR, and then by defining characterization metrics for data's analysis.

## Real Measurements

In the scope of the 3SA project at IRT SystemX, the aim was to collect real data corresponding to our requirements: having disturbed sensor data with an accurate enough description of the perturbation and a controlled scenario in order to be able to simulate

it. Therefore, an existing LiDAR has been used to collect the data in specific environments that are described in the following subchapters.

## Outdoor measurements

The first attempt of collecting data has been done in an outdoor environment at IRT SystemX in Palaiseau (France). Measurements have been done in the same spot in different days in order to have different weather conditions.



Figure 2: The environment that has been recorded with our LiDAR during a foggy day

## Platform of simulation of degraded climatic conditions

In order to characterize the LiDAR sensor and to propose a realistic model of the sensor we have in addition to the first tests outside the IRT-SystemX realized other campaigns of data collection by integrating the LiDAR sensor in a platform of simulation of degraded climatic conditions based in Clermont Ferrand within the CEREMA laboratory. This simulation equipment used is referenced within the LABEX ImobS3 "innovative mobility: intelligent and sustainable solutions" among the regional platform "PAVIN B-P", Auvergne platform for intelligent vehicles Fog - Rain.

The installation is in the form of a covered track about thirty meters long, specifically equipped and instrumented with various materials (artificial vision systems, transmissometers, rain and fog generators, photometers, granulometers, radiometers, video-photocolorimeters, spectropluviometers), with the possibility of carrying out tests, day and night.

The CEREMA laboratory linked to the platform is also equipped with materials allowing the measurement of the photometric characteristics of the studied objects, namely luminance and colorimetry. As an object we have realized some test scenarios by placing in the scene road signs, road markings.

The PAVIN test platform consists of a tunnel section and a greenhouse section, the whole of which is 30 m long and under which meteorological disturbances can be generated. This space constitutes the measurement and acquisition area for the Lidar. The entire platform is made up of a draining surface typically used on motorways.



Figure 3: The objects inside the PAVIN platform



Figure 4: Example of rain generation inside the platform

## Metrics of characterization

In order to compare simulated point clouds with real point clouds. There is a need of having metrics of characterization of these point clouds and the associated perturbation we want to simulate.

In the case of the fog simulation, the focus will be on quantifying the fog and then quantifying the implied backpropagation to be able to model it.

## Fog characterization

In the meteorology domain, the fog density is quantified using a value of meteorological visibility distance that is defined by the International Commission on Illumination (CIE) (Commission, 1987) as the distance beyond which a black object of an appropriate dimension is visible with a specific contrast limit.

The CEREMA Pavin platform contained sensors that is able to measure in real time the fog density by giving this meteorological visibility distance. In order to categorize the fog, the LiDAR data that was collected in the platform were acquired during scenarios that ranged from 10 to 30 minutes. In these scenarios, the fog density gradually changed, increasing in visibility, starting with a visibility of 10 meter and gradually ended at a visibility of 1000m. The gradual change in the visibility of the fog is slow enough that we can consider it as a pseudo-static perturbation.

That is why for each measured point cloud, a visibility distance can be associated, representing a fog characterization metric.

## Density distribution

For each point cloud, there is a precise number of points (variations of this number of points occurs

because of the effect of multi-echoes). These points can be counted to define a first metric.

Nevertheless, there is a need of differentiating points that represent real targets (that will be called “hard target” in the following chapters) than points that are caused by the fog’s backpropagation (that will be called “soft target” in the following chapters). That is why a metric has been defined to count the soft target points density depending on the range in front of the LiDAR by using a ray tracing method.



Figure 5: The method of measuring fog points density in the LiDAR point cloud

## Ray extraction

A sub point cloud from the main point cloud is extracted base on a set of selected rays (noted  $R$ ). The main reason of having ray selection is to be sure that only points that are characterize a single object are selected in order to have only representation of fog perturbation and the corresponding object visibility. Any ray that collides with the roof, the ground or the sides of the tunnel is then not selected. Usually, rays that are directed towards the object at the end of the tunnel are selected.

## Ray slicing

After extracting the points included in the selected rays, the rays are then sliced into parts for a given step of range from the LiDAR. The goal of having these slices is to measure the quantity of points in each range step.

Having such a method leads to measure the density for each step of range. The points that are located in the near range can help characterizing the backpropagation and the points located further can help characterizing the visibility of the object (depending on its type, its position, etc.).

For each ray (noted  $i$  in a direction defined by an elevation angle  $\theta_i$  and an azimuthal angle  $\varphi_i$ ), zero to three echoes can be detected at different ranges or distance from the LiDAR, also containing the intensity of each echo:

$$echoes(\theta_i, \varphi_i) = \begin{cases} echo_1(\theta_i, \varphi_i) \\ echo_2(\theta_i, \varphi_i) \\ echo_3(\theta_i, \varphi_i) \end{cases}$$

With each  $echo_1(\theta_i, \varphi_i)$  being the first echo or the nearest distance corresponding to the laser ( $\theta_i, \varphi_i$ ) and each  $echo_3(\theta_i, \varphi_i)$  being the third echo or the farthest distance corresponding to the laser ( $\theta_i, \varphi_i$ ).

## Density repartition representation

For each point cloud, for the set of selected rays  $R$ , we can then define 3 cumulative distribution functions corresponding to each echo (first, second and third) that are defined as follow:

$$\forall j \in \{1,2,3\}, \quad F_j(x) = \frac{1}{|R|} \sum_{i \in R} [echo_j(\theta_i, \varphi_i) < x]$$

For a given fog visibility, these 3 cumulative distribution functions can be computed and even a mean value can be computed for a set of several point clouds (to get smooth curves). This work has been done with the points clouds of CEREMA.

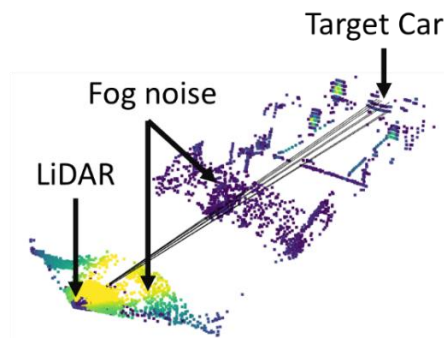


Figure 6: Example of measured point cloud with the selected rays oriented towards the target car

Figure 7 shows the 3 echoes mean cumulative distribution functions are given for 3 levels of visibility (296 point-clouds corresponding to 20m visibility, 56 point-clouds corresponding to 100m visibility, and 15 point-clouds corresponding to 1000m visibility).

### Intensity Scaling

Similar to the distribution of the density of the points from the distance of the lidar, we can also extract the intensity of the lidar points due to the fog noise.

For each point cloud, and every occurrence of the selected rays with a hit (points that for a given ray in a given time stamp exist) *Roccurance*, we can also define the 3 cumulative intensity functions for each echo.

$$\forall j \in \{1,2,3\}, \quad F_j(x) = \frac{1}{|Roccurance|} \sum_{i \in R} [echoIntensity_j(\theta_i, \varphi_i) < x]$$

Figure 8 shows the 3 echoes mean intensity function for each given echo, at 3 levels of visibility.

## Perturbation model construction

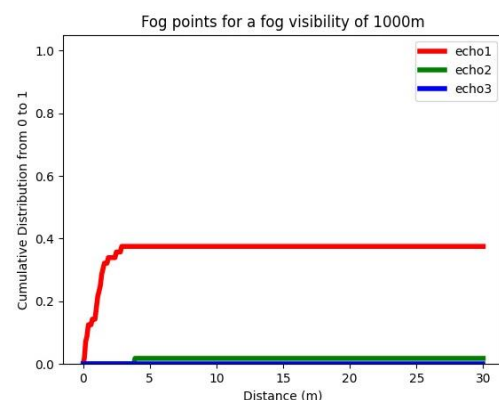
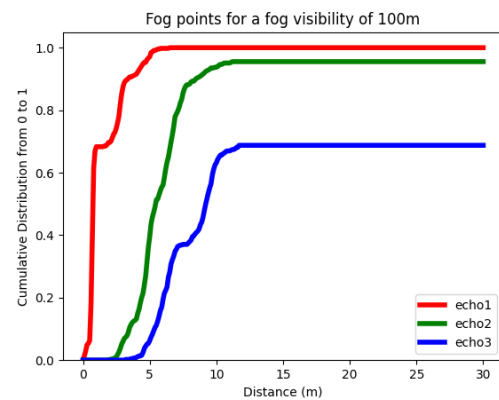
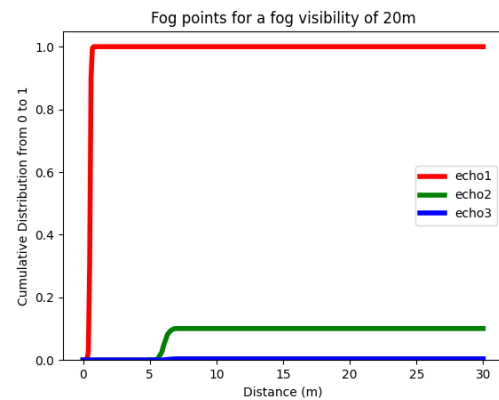
After having analysed real data, the goal of the study is to create from it a model simulating fog backpropagation phenomenon (soft target).

### The LASERMETER model

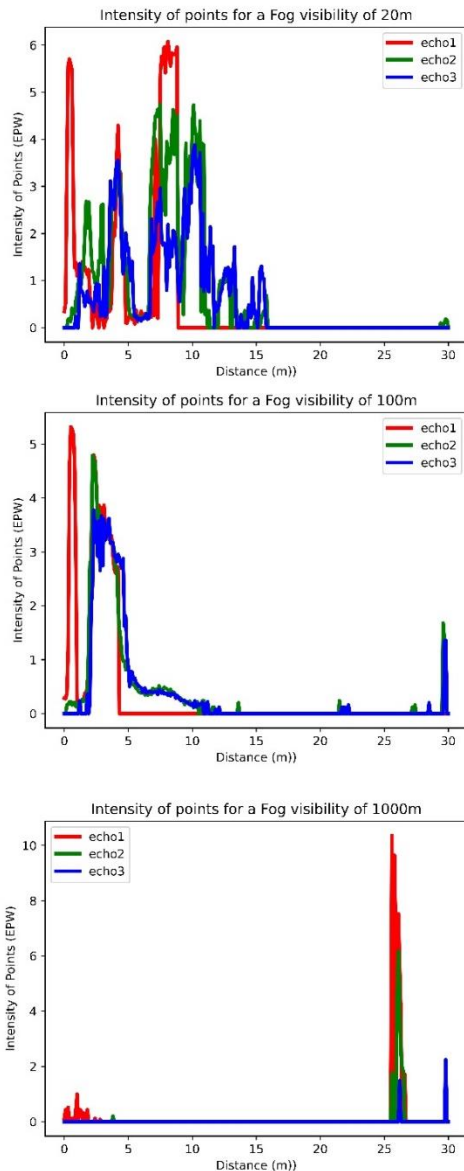
The simulation tool software that is used is AVSimulation’s SCANer Studio software. This software relies on a modular architecture with each module assuring a task of the simulation (one for the vehicle dynamics, one for the traffic management, one for the pedestrians, one for the sensors, etc.).

One of the modules, LASERMETER oversees LiDAR modelling, relying on ray tracing. The ray tracing pattern is configured depending on the laser emitting patterns of the LiDAR that can be parametrized in the model.

The actual version of LASERMETER models is a perfect LiDAR model that is not affected by surrounding anomalies. Any target in the field of view of the simulated laser beam would be detected regardless of weather conditions. The aim of the study is to improve the current LiDAR model to take into account these perturbations, as shown in the following proof of concept implementing fog effects in simulation.



**Figure 7: Mean cumulative distribution function for each echo and each fog visibility**



**Figure 8: Mean intensity functions for each echo and each fog visibility**

### Perturbation generation

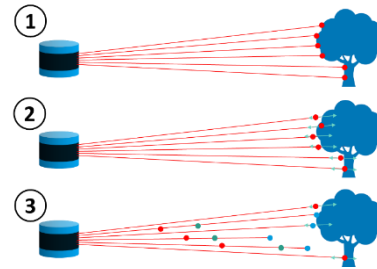
The improvement of the LASERMETER model is done via a Plugin as a layer on the current model. This include correcting the obtained point cloud to create the effect of the fog perturbation.

**First step: Getting LASERMETER output:** The output is a point cloud with one, two or three points for each laser beam. The points coordinates are generated in reference to the sensor frame.

**Second step: Adding precision noise:** The distance for each point can be changed with an error defined by a probability distribution. An example of a simple model is using gaussian error to add a percentage of error in the distance estimation.

**Third step: Adding soft target (point of backpropagation):** The backpropagation effect is happening when points are generated between the LiDAR and the target. The same effect can be applied by generating point for each laser beam. The backpropagation points position is defined regarding a probability distribution. If this random position is further than the target object, then no modification is done. If this random position is between the LiDAR and the object, then this position is defined as the new point for the corresponding laser beam and the target point is ceases to exist.

The different steps are illustrated in figure 9



**Figure 9: Different steps from the LASERMETER output (1) to add precision noise (2) and then backpropagation due to fog (3)**

### Distribution of backpropagation

The third step presented in the previous subchapter must be parametrized through a probability distribution. Indeed, the backpropagation occurrence is determined by a randomly picked position.

#### Using an exponential distribution

A simple probability distribution that determines the position of the backpropagation point for a given laser beam could be the exponential distribution defined by the following equation for a given range  $x$ .

$$P(X > x) = 1 - e^{-\lambda x}$$

Such a distribution law would be representative of a fog where most of the points are in the very near field of the LiDAR with less and less points in the further field.

#### Using the cumulative distribution function

The results of the CEREMA measurements have shown that soft target representing fog backpropagation can be more complex in terms of positions and multi-echoes, depending on the fog density. That is why we can define a probability distribution with the density distribution obtained from the metrics. By taking them as a probability law, a new way of generating soft target can be defined.

$$\text{For each echo } j, P(X > x) = F_j(x)$$

Intensities of the soft target points take into account the same probability law and is extracted from the mean intensity function. In addition to positioning the soft target points depending on multi-echoes,

intensities of the generated points are scaled to correspond to the real data used in the simulation.

## Simulation with the perturbation

### Simulated scenario

In order to compare simulated data with real data, the CEREMA Pavin platform has been modelled to have a similar scenario running in SCANer Studio. Moreover, the correct parameters have been chosen to model the simulated sensor according to the datasheet of the used LiDAR.



Figure 10: The simulated environment in SCANer Studio

### Perturbation model construction

A module is used in SCANer to add the perturbation model generated previously into the point cloud that is generated by the LASERMETER model (through the communication bus of SCANer Studio which is known as the "Network") and publishes modified point cloud as PCD files respecting the same format as the LiDAR's point cloud outputs.

### Obtained point clouds

By applying the presented methodology with the exponential law fog perturbation. The obtained point cloud is given in Figure 11.

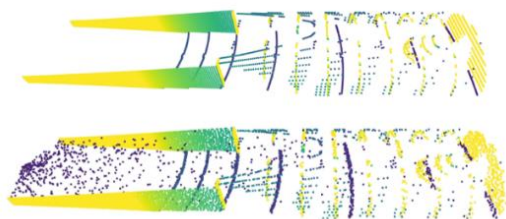


Figure 11: The simulated perfect point cloud on top and the obtained point cloud after using the noise model (exponential law)

## Results and discussion

### Comparison

#### Qualitative analysis

The presented methodology uses three parts to generate the noisy point cloud; the collected data with the method to analyse the density, the simulation environment with the virtual LiDAR and

the virtual CEREMA Pavin Platform, and the noise model relying on a defined probability law.

Merging these three elements lead to have a scaled model of noising based on the collected data. The obtained result is presented in the figure below.

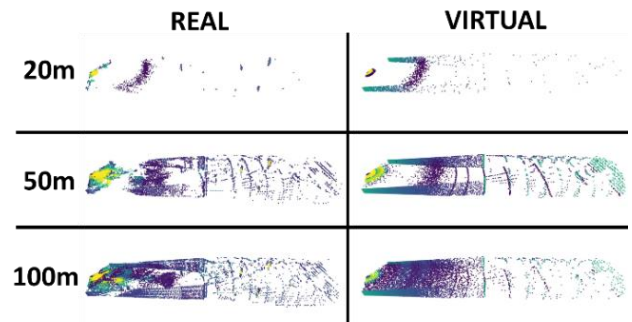


Figure 12: Obtained point clouds for real and virtual values

Both the point clouds presented have a significant similarity that is visible, and the presence of the fog model adds a lot more realism. However, improvements can be done in removing the isotropy of the model.

### Quantitative analysis

A more quantitative method can be used in order to compare obtained point clouds. The goal of such metrics has multiple reasons: having a visually coherent point cloud, having precision in representing objects and having a similarity between point clouds in terms of software detection. A first analysis based on computing point cloud density has been introduced.

This metric is designed to consider the whole point cloud by splitting it into voxels, with each containing the number of points to compute the density per voxel (as shown in Figure 13). By comparing the densities between real point cloud and simulated point cloud, the difference can be computed and a metric has been established using two formulas that count the numbers of points  $N_{i,real}$  or  $N_{i,simulated}$  in each voxel  $i$ .

$$Metric1 = \frac{\sum_i (N_{i,real} - N_{i,simulated})}{\sum_i (N_{i,real})}$$

$$Metric2 = \frac{1}{Nb\_voxels} \sum_i \frac{N_{i,real} - N_{i,simulated}}{N_{i,real}}$$

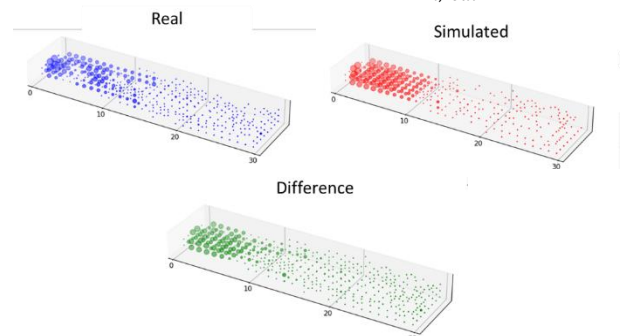


Figure 13: Representation of the comparison metric



The metrics has been computed for some of the obtained point clouds and the following table has been obtained.

Table 1: Obtained metrics for different level of fog

Fog	With fog model		Without model	
	Metric1	Metric2	Metric1	Metric2
20m	20%	1%	182%	10%
50m	50%	11%	131%	14%
100m	59%	15%	111%	17%

The obtained metrics are different regarding the used formula and these can be improved by taking into account intensities, and size of cells of the grid. The results show that the noise model reduces the difference between the virtual and real data.

## Future improvements

The model presented in this paper let users of the simulation have a first representation of degraded fog measurements, with room for improvements.

## Non homogeneous soft targets

To ensure a more realistic model, the need to implement the soft target via spot or cloud forms are needed, to avoid the problems of homogenous soft targets.

## Azimuth dependence

The current model does not yet implement the different azimuthal direction as had been collected at PAVIN, which could improve the model's precision.

## Dynamic study

The data collected has been done for static scenarios corresponding of fixed targets and a fixed vehicle. Weather perturbation (especially fog) can have important variation when the car is moving and dynamic studies allow a more realistic scenario.

## Metrics of comparison

The work of comparison of the point clouds has to be improved with other metrics and to be applied for other point clouds. Moreover, a comparison with other scenarios that have been acquired outside the CEREMA platform can enable a richer validation of the methodology of modelling.

## Conclusion

This paper proposes a methodology that is able to characterize and implement adverse weather conditions into a simulated LIDAR sensor model, aimed to be used for the validation and homologation of autonomous vehicles. The approach is based on collecting and recording real life LIDAR data under various adverse weather condition, and from that generating the perturbation model. This perturbation

model is then implemented into the perfect sensor model to provide a more realistic sensor model to be used and tested against. It is important that the collected real data contains precise information about the weather condition to be able to correctly identify the type of perturbation to be simulated. This approach using real life recording can be implemented to various other types of weather perturbations, and is also not just limited to LIDAR sensors, but also in the case of camera, radar and even ultrasonic sensors. This approach is also less resource consuming compared to other physical based sensor models, and allows for its implementation in real-time simulations.

## Acknowledgement

This work has been supported by the French government under the "France 2030" program, as part of the SystemX Technological Research Institute.

## References

- Byeon, M., & Yoon, S. W. (2021). Analysis of automotive lidar sensor model considering scattering effects in regional rain environments. *15th European Conference Antennas and Propagation*.
- Chan, P. H., Dhadyalla, G., & Donzella, V. (2020). A Framework to Analyze Noise Factors of Automotive Perception Sensors. *IEEE Sensors Letters*.
- Commission. (1987). *International Lighting Vocabulary* (Vol. 17.4). Vienna, Austria.
- Espineira, J. P., Robinson, J., Groenewald, J., Pak, C. H., & Valentina, D. (2011). Realistic LiDAR With Noise Model for Real-Time Testing of Automated Vehicles in a Virtual Environment. *FOurth IEEE International Conference on Software Testing, Verification and Validation*.
- Filgueira, A., Gonzales-Jorge, H., Lagüela, S., Diaz-Vilariño, L., & Arias, P. (2017). Quantifying the influence of rain in LiDAR performance. *Measurement, Volume 95*, 143-148.
- Goodin, C., Carruth, D., Doude, M., & Hudson, C. (2019). Predicting the Influence of Rain on LIDAR in ADAS. *Electronics*.
- Kutilla, M., Pyykönen, P., Holzhüter, H., Colomb, P., & Duthon, P. (2018). Automotive LiDAR performance verification in fog and rain. *21st International Conference on Intelligent Transportation Systems*.
- Rosenberger, P. a. (2019). Benchmarking and Functional Decomposition of Automotive Lidar Sensor Models. doi:10.1109/IVS.2019.8814081
- Zhang, C., Ang, M. H., & Rus, D. (2018). Robust lidar localization for autonomous driving in rain. *IEEE/IROS*.