Analysis of Real World Sensor Behavior for Rising Fidelity of Physically Based Lidar Sensor Models

Philipp Rosenberger, Martin Holder, Marina Zirulnik, and Hermann Winner

Abstract— Safety validation tests of automated driving (AD) use simulated environments and perception sensor models. To achieve the level of fidelity needed for safety approval, such sensor simulations can be physically based. For formulating requirements for sensor models to be used in such test frameworks, the extent to which they must include physical effects should be determined. One approach is to clarify their relevance for following processing steps like object detection or mapping. But at first, an analysis is needed to determine, which effects are relevant and if they can be implemented at all. In this work, we focus on one lidar sensor and analyze its observable real world sensor behavior to derive the possible effects, physically based lidar sensor models can include. Consequently, we describe environmental parameters that could be considered to influence physically based lidar sensor models. By investigating the specifications given by the manufacturer with own measurements, we show that some of them should be implemented in a dynamic manner. In conclusion, we enable to formulate detailed requirements for sensor models, as their actual possible fidelity is presented.

I. INTRODUCTION

Since it would take billions of kilometers in road tests to prove safety of automated driving [1], virtual test methods are considered to be the more efficient solution in automotive industry [2]. The German project PEGASUS has the goal to develop and implement such efficient test methods for the automotive use case called *highway chauffeur* [3]. One part of virtual test environment consists of perception sensor models and their interaction with the simulated environment and the following processing steps and functions. Testing of automated driving systems [4] requires high accuracy and therefore virtual testing of automated driving functions have to use accurate models of the simulated environment as well as of the vehicle's environment perception sensors. However, it is still unclear how requirements for sensor models usable in such validation tests can be defined. Even if there are several approaches how to model automotive lidar and some work has been published about validation of such, there is no methodology or guideline what extend of physical effects or stochastic behavior one should consider when simulating lidar. When the intended function of a model is clarified, we see the main contribution of our work in enabling formulation of requirements. This is provided by giving insight into the physical processes during measurement and by stating possible implementations. At the end, their impact, complexity and effort of inclusion can be estimated.

If it comes to safety test by simulation, critical scenarios are of interest, and by critical not only critical movements of involved dynamic objects but especially situations critical from sensor perspective should be considered. To be able to simulate degraded sensors, physically based sensor models offer the flexibility to inject failures based on their respective causation. Ideal, probabilistic, or phenomenological models as in [5] or [6] have a disadvantage in generalization ability for unseen situations. All of them are built using a-priori-knowledge and possibilities based on collected data. Recently, also machine learning approaches have been reported, [7] however, their performance on unseen scenarios remains unclear. No expert or algorithm that puts his or data-driven knowledge into such models can be aware of every combination of relevant effects affecting the sensors in any possible situation. Therefore, those models are not fully capable to be used in safety relevant tests. Nevertheless, solving this with fully physical simulation in every detail (like FEM, etc.) is not feasible because of e.g. computation time. So we analyze the real sensor systematically and propose how to implement the effects and aspects determined as necessary in a computational efficient way. Consequently, the model still stays physically based keeping the simulation efficient.

Another important aspect to consider is the output interface of the developed sensor model. The already mentioned ideal, probabilistic and phenomenological models are mostly staying high-level and work on object lists, changing object positions, existences or types, especially when real time capability is required. There is already an ongoing standardization process to cover that special data type and establish a standard for sensor models [8]. Lidar sensors are besides object classification also commonly used for mapping and localization, e.g. [9] as well as in sensor fusion setups [10]. Therefore, we consider the point cloud as universal representation of lidar data as it allows unrestricted processing for different applications. Consequently, physically based sensor models, will need to simulate sensor data on this *low level* interface as it delivers unprocessed data. By simulating point clouds, object lists can still be derived through object detection and classification algorithms. For implementation we propose to use state of the art ray tracing, e.g. like in [11], to apply all the aspects we will describe in the following chapters. The desired solution after all would be a model leveling out the existing dilemma between efficient simulation and high fidelity [12].
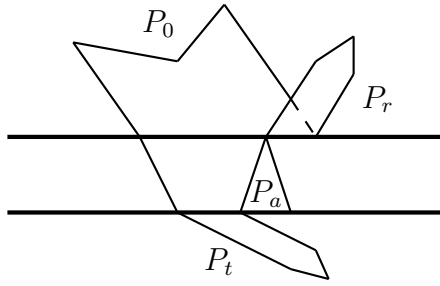
Fig. 1: Sum of Luminous Power on Illuminated Objects [15]

The remainder of this paper is organized as follows: We introduce definitions at first and review basic principles of lidar sensors. Then we discuss environmental parameters to consider and analyze one real lidar sensor. In the course of this, besides the measurement based investigation of the given specification by the manufacturer, we determine existing noise and calculate the temporal effects to explain its impact on simulations that put the complete environment on hold for a whole scan period.

II. DEFINITIONS

Verification and validation of sensor models can only be performed if there are requirements that have to be fulfilled. Requirements for sensor models are formulated to ensure that the model fulfills its intended function [13]. Verification, as a binary decision, means to show the successful implementation of certain aspects or effects that have to be included according to the requirements. For validation of physically based sensor models, there have to exist metrics to compare for example how close synthetic output data is compared to real data. A certain degree of fidelity can be postulated in requirements best, when the full range of includable aspects and effects is given. Then it is measured during validation if the sensor model reaches the required fidelity.

To enable formulating requirements for sensor models, in this work, the necessary fidelity of virtual lidar sensors is derived from an analysis of real world sensor behavior. We do not claim to deliver a complete list of all effects, but every considered section may increase the fidelity of an implemented model. A possible use of this collection is to derive requirements for a sensor model and its step-by-step validation, showing the implementation of all the different aspects as in [14]. Nevertheless, it is already possible to evaluate existing approaches claiming to be physically.

III. LIDAR SENSOR BASICS

Before describing all aspects that could and most likely should be covered by physically based sensor models, this section contains all necessary lidar basics. The abbreviation *lidar* is derived from *light detection and ranging* what is referred to the measurement principle explained in the following lines.

Lidar sensors send out pulsed light beams with a wavelength of $850\,\mathrm{nm}$ to $1\,\mathrm{\mu m}$ [15]. Automotive lidars have a limited transmission power because of required eye safety.

The transmitted infrared light gets reflected on surfaces and the portion that reverts to the sensor's photo diode is measured. Using the *time of flight* principle, the distance is calculated with $r = \frac{ct}{2}$, where $\mathbf{c}$ is the speed of light and t is the measured time of flight [16]. More precisely, t refers to the time interval between emitting of the light pulse and receiving a signal with a signal strength upon a predefined threshold.

Parts of the emitted luminous power P_0 get reflected on objects P_r. Besides the portion that directly finds its way back to the sensor, most of the reflected part of the beam is dissipated into the environment. Reflection is called ideal diffuse or Lambertian, if the light is reflected homogeneously in every direction. In contrast, specular reflection describes inhomogeneous reflection under specific angles. Apart from those reflections, it is possible that the light is partly absorbed P_a and converted into heat or transmitted P_t by the illuminated material. The sum of $P_0 = P_r + P_a + P_t$ [15] combines all the described beam reactions and is illustrated in Figure 1. The measurable sensed intensity after a reflection on a single object P_1 can be calculated with the Laser-Radar-Equation [17]

$$P_1 = K_r T^2 \frac{A_{sensor}}{A_{beam,r}} \frac{A_{refl}}{A_{beam,0}} P_0, \qquad (1)$$

where K_r, T and A_{sensor} denotes the reflectivity of the object, the transmittance of the atmosphere (which is passed twice), and the receiving lens surface, respectively. The reflecting surface

$$A_{refl} = MIN(A_{beam,0}, A_{object}) \qquad (2)$$

depends on the object size A_{object} and the beam size

$$A_{beam,0} = \alpha_v \alpha_h r^2 \pi, \qquad (3)$$

which is calculated with the distance r and the vertical and horizontal beam divergence α_i using the small-angle approximation. The beam size of the reflected beam

$$A_{beam,1} = f(\theta_{total}, \Delta\theta_{div}) r^2 \qquad (4)$$

depends on the effective solid angle towards the lens in surface normal direction from the object and the distance.

IV. ENVIRONMENTAL PARAMETERS TO CONSIDER

A sensor model for testing of AD is depending on the level of detail of the whole simulated environment. The equations in the previous chapter give a list of environmental parameters that could be considered for physically based lidar sensor models and will be discussed below.

The first parameter is the transmittance T in (1) which implies atmospheric attenuation. It is affected by temperature, pressure, humidity [18], and different types of precipitation as rain [19], [20], wet and dry snow [21], as well as fog or haze [22]. Even exhaust gases can have a strong impact on laser scanners at low temperatures causing condensed water drops [23]. Pollution of the sensor or parts of it is also included in T.
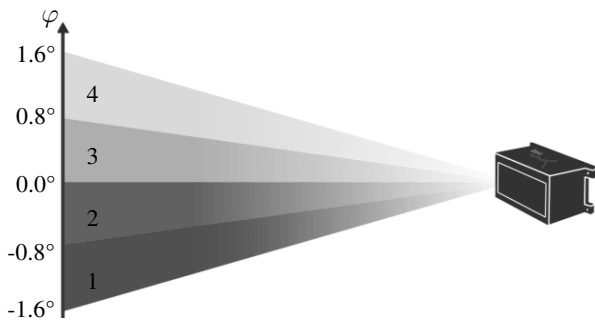
Fig. 2: Scan Layers and Vertical Beam Divergence of Ibeo LUX 2010 [25]

Fig. 3: Angular Resolution of Ibeo LUX 2010 [25]

The second parameter to be modeled is the reflectivity of materials considered in the simulated environment, which is represented by parameter K_r in (1). Materials and their properties are included by look-up tables in various simulation tools. Nevertheless, it can be done in different levels of detail. We suggest using physically based rendering including a complete data base of reflectivities and derived from computation techniques for the visual spectrum of light as described in [24]. As automotive lidars use a wavelength close to the visible spectrum, existing data bases for visible light can be used in a first approximation. Surface normals of the whole environment as part of (4) are used for computation of backscattered rays. Therefore, it is in theory possible to reproduce total reflection and correct multi-path propagation.

As a third environmental parameter, we consider the surrounding amount of light. It is a direct influence to the lidar measurement by inducing a specific amount of intensity into the sensor collected as bias. In consequence, the signal to noise ratio (SNR) is reduced leading to a change of the usually dynamic detection threshold and possibly to a worse performance of the sensor. The solar altitude is of special interest here, as direct sunlight e.g. at sunrise or sunset has a strong impact on lidar sensors that even blinding could occur [15]. The same effect is possible when direct illumination by other lidar sensors occurs.

V. ANALYSIS OF SENSOR BEHAVIOR IN REAL-WORLD

The aspects of physically based lidar sensor models described in the following chapter are derived from an analysis of the widely used Ibeo LUX 2010 [25]. However, the presented method of analysis remains applicable for any automotive lidar sensor.

A. Ibeo LUX 2010 Specification

The Ibeo LUX 2010 [25] uses the 3D scanning principle having four vertical scan layers with 0.8° difference in elevation leading to an elevation range of 3.2° as illustrated in Figure 2. It has an azimuth range from +50° to −60° with selectable angular resolution and scan frequency. At a scan frequency of 12.5 Hz the angular resolution in azimuth can differ by sector. In the central range (1) in Figure 3 the resolution is 0.125°, in the medium range (2) it is 0.25°, and in the lateral range (3) 0.5°. The spatial characteristics of

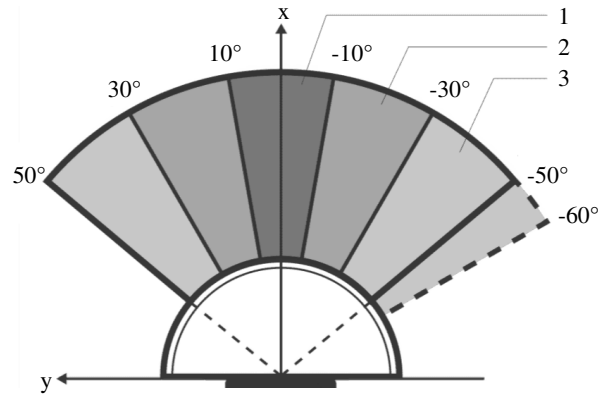the resolution is to be modeled in a physically based model. To facilitate the interpretations, we chose to use a constant angular resolution of 0.25° at a scan frequency of 12.5 Hz for all later described experiments.

The specification lists an object detection range from 0.3 m to 200 m at an elevation angle of 0°. Further the repeat accuracy in distance measurement is 0.1 m (1 σ) and the pulse duration is 4.5 ns. All that describes the field of view of a sensor and is the only adaption an ideal sensor model has to enable. For a sensor model utilizing ray tracing, this corresponds to setting the angle and amount of rays that are transmitted in the simulated environment. The frequency of the scanner gives the frequency that the sensor model has to achieve if real time capability is a requirement, what is not needed in every test architecture, but becomes important if it includes real hardware as in HiL or ViL tests.

B. Sensor Position and Orientation

The output of one single sensor is a so called raw point cloud in relative spheric coordinates including per point its distance, angle, layer, echo no., and echo-pulse width [25]. The frame origin is the sensor itself, what dictates the origin of an ideal sensor model or of rays send out in the virtual environment when ray tracing is applied. The orientation of the sensor defines the direction of the rays that are sent out. It is possible to fuse several sensors e.g. using the Ibeo ECU [25]. As a result, the fused point cloud is given in relative Cartesian coordinates with its origin at the center of the rear axle of the vehicle. To be able to compare such a point cloud with a synthetic one fused of different simulated sensors, they shall have the same origin as well.

C. Different Orientation of Scan Layers

Figure 4, showing measured data from Ibeo [25], visualizes measured raw point clouds. 10 % remission here means 10 % of reflection in contrast to perfect reflector, what implies that the shown field of view displays no hard detection border, but gives a good overview. It illustrates the different angular orientation of the scan layers. Two (no. 1 and 3) are orientated more to the left and the others are looking more to the right. The whole azimuth angular coverage of the sensor is 110°.
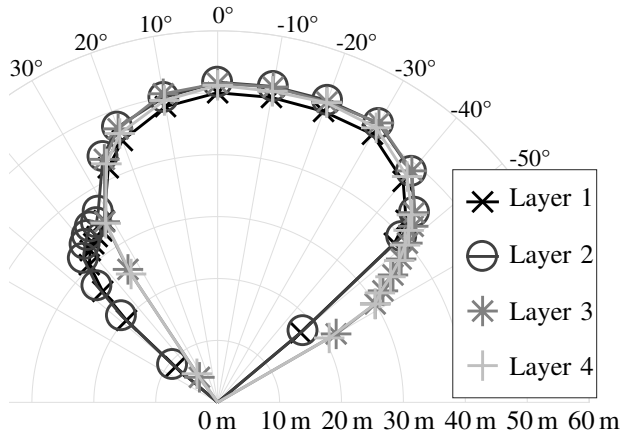
Fig. 4: Ibeo LUX 2010 Angular Dependent Measurement Range to a target with 10 % remission [25]
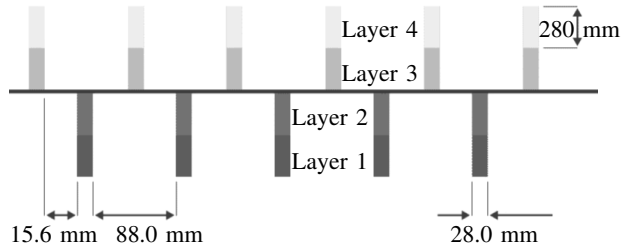

Fig. 5: Zipper-like Layer Orientation and Beam Size at 20 m with an Angular Resolution of 0.25° of Ibeo LUX 2010 [25]

Besides the explained shift for higher angular range, beams are also shifted to reach a higher angular resolution forming a zipper-like shape as shown in Figure 5. Both shifts could be included when programming the ray tracing algorithm in simulated environments by transmitting accordingly.

D. Accuracy, Range and Noise

Ibeo specifies a repeat accuracy (1σ) of 0.1 m in range measurements from 0.3 m to 200 m [25]. Figure 6 shows the repeatedly measured differences $S_i = r_{nom,i} - r_{meas,i}$ for scan layer 1 and nominal distances r_{nom} from 2 to 200 m. Measurements were performed under open sky and different weather conditions from windy and cloudy to sunny, which made no difference. A Trimble 3600 [26] tachymeter served as reference measurement system where an ordinary gray carton was used as reference target. Displaying mean values as well as median and box plots for all distances, Figure 6 shows that with increasing distance, the error rises, showing a linear trend, but the precision holds for all distances. Both should be included in physically based lidar models.

The Ibeo specification gives a maximal range of 200 m [25]. As expected, this is not a static value as it depends on the parameters already described in section III. We show its dynamic with a series of measurements up to 350 m with gray carton in size and reflectivity of a dirty back of a car and with a stop sign. The stop sign can partly be measured up to 325 m, appearing in every scan up to 300 m. Gray
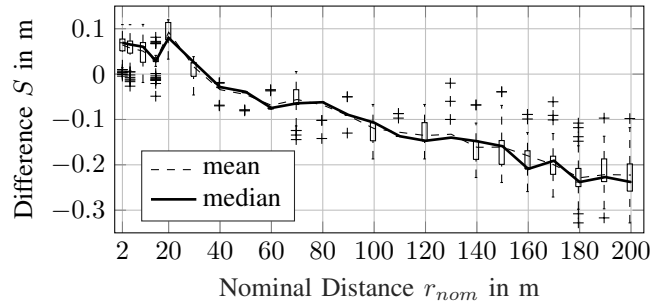
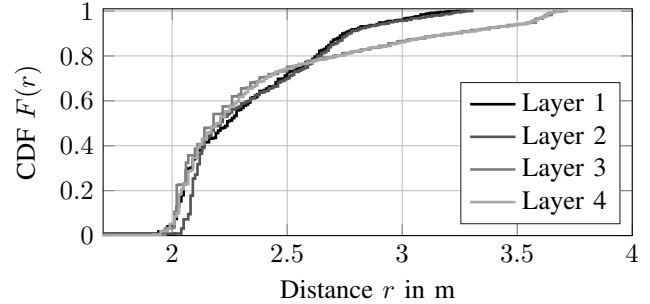

Fig. 6: Accuracy of Layer 1 of Ibeo LUX 2010

Fig. 7: Cumulative Distribution Function from 2 to 3.5 m

carton is recognized in 95 % of all scans at 200 m going down to 8 % at 271 m. These findings should be included in physically based models, complementing the equations for backscattered power in section III as explained in section IV.

Measurement data is naturally affected by different noise sources. Besides randomly distributed parts, in case of the investigated lidar sensor we find deterministic noise caused by its counter resolution. If the distance is continuously and slowly changed and measured as it has been done for Figure 7, the counter resolution is implied in the step height of the graphs. Figure 8 displays the ratio of all occurring step sizes of a performed measurement series targeting gray carton over the distance r from 6.056 m to 6.156 m in 1 mm wide steps, and from 6.156 m to 6.256 m in 5 mm wide steps. It states, that definitely only the four indicated step sizes occur, as their ratios always sum to 1. Resulting in very detailed and expensive investigations, such investigations can most likely not be performed for the whole measurement range, but it should be repeated for more intervals and then the found discrete distribution should be implemented in a sensor model. It should be stated, that no other than the described deterministic noise has been found in our range measurements. We conclude, that this noise can be modeled as quantization noise rather than Gaussian white.

E. Beam Divergence

It has been mentioned that the angular difference in elevation is 0.8°. But as stated in [25], the vertical beam divergence from (3) is exactly the same as the angular resolution of 0.8°. This determines the position of the beam
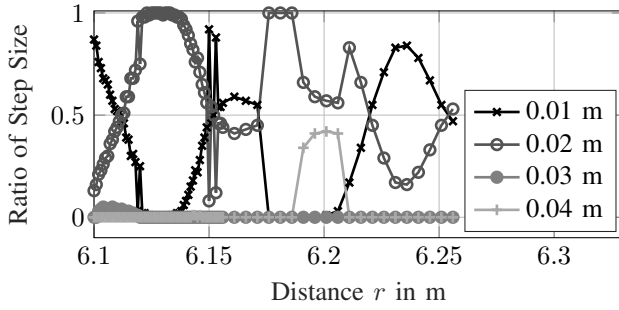
Fig. 8: Counter Resolution of Layer 1 of Ibeo LUX 2010

to the center of the angular ranges shown in Figure 2. The horizontal beam divergence, which is also part of (3), as given in [27], is 0.8°. Both divergences are visualized in Figure 5, forming an idealized rectangle shaped beam, 0.28 m high and 0.028 m wide, at a distance of 20 m.

Beam divergence is critical if the object size is smaller than the beam size like for long distances or when the reflectivity over the surface in (2) shows a step. This happens for example at the change from car paint to the headlights or license plates. For the implementation of beam divergence, one infinitesimal thin ray is not enough. We suggest to shoot more than one ray to approximate the beam divergence and find the closest or most intense hit in that beam volume in a still computational efficient way.

F. Multi-Echo Perception, Echo-Pulse Width and Signal Intensity

In contrast to sensors like Velodyne's HDL-64E [28], which measure the intensity P_1 in (1) directly, Ibeo sensors measure the echo-pulse width of the received signal. The Ibeo LUX detects up to three echoes from one transmitted pulse, which is called multi-echo perception. Multiple echoes can occur if e.g. a laser beam bounces at adherent snow, then at a window and then at the inside of a car. Condition for a multi-echo reception is, that the signal strength falls under the detection threshold after the first echo and rises again.

To be able to compute the intensity out of the echo-pulse width, its height needs to be sampled with sufficient frequency as well. As this is not the case for the Ibeo LUX 2010, the given echo-pulse width cannot be used as intensity indicator. To prove that statement, two perceived objects can have the same echo-pulse width, but if their reflectivity is not the same, they are not causing the same intensity to be measured. Recent work has shown that the implementation of the echo-pulse width will not improve the model quality: As its does not appear to be a highly descriptive feature, typical object classification algorithms do not necessarily benefit from echo-pulse width information [29]. We therefore cannot emphasize the need to model the echo-pulse width. Because of the information it adds to a specific hit point, intensity should be included in a physically based sensor model, if the real sensor has it as an output. Again, the equations and parameters in section III and explained in section IV can be used for physically based rendering.

G. Temporal Order of Transmitted Beams

The last aspect to be considered when simulating a (raw) point cloud is the temporal order of the transmitted beams. As different beams are sent out at different time, the ego vehicle and dynamic objects are moving in the meantime. This can lead to well known effects like rolling shutter and motion blur, which all have in common to suggest wrong shape and size of recorded objects. In contrast, actual simulations stop the environment for the whole scan duration eliminating such effects. Rolling shutter for example is caused when recording different lines of a picture, or scan layers in lidar context, leading to different positions of the same object in different layers causing distorted point clouds. This effect can occur using the Ibeo LUX 2010 as the layers are scanned interlaced with two of four simultaneous layers.

Motion blur in lidar context means a wrong perceived size of an object, caused by the scan duration and the movement of the object in the meantime. For the lidar point cloud, it can lead to wrong positions and amounts of points belonging to one object and therefore to wrong cardinalities of the later clustered points. The error in perceived object width Δw_{obj} by motion blur is equivalent to the way x_{obj} that the object moves relative to the scan direction during its scan time t_{scan}. If we consider that the scan itself and the post processing is fulfilled during one scan period and that the duration needed to scan one object is much smaller than one scan period, the actual scan will only take e.g. $t_{scan,actual} = 0.25 \cdot t_{scan}$. Therefore, the error is calculated by $\Delta w_{obj} = v_{obj,lat} \cdot t_{scan,actual}$. As an example, we consider a static ego vehicle waiting at an intersection and a vehicle passing by in lateral direction with a relative velocity of $v_{obj,lat}$ = 15 m/s and a length l_{obj} = 5.0 m at a distance r = 5.0 m. The lidar sensor is orientated to the front and has a scan frequency of f_{scan} = 12.5 Hz, which leads to an object width error Δw_{obj} = 15 m/s·20 ms = 0.30 m.

Continuing this example, we consider an angular resolution $\Delta\phi_{res}$ = 0.25°. The mentioned object length leads to an angular range ϕ_{range} = 60° covered by the vehicle, what results in an amount of beams targeting the vehicle of $n_{beams,obj} = \phi_{range}/\Delta\phi$ = 240 beams in a static case. In case of the moving object, the theoretical beam error is $\Delta x_{beam} = \Delta w_{obj}/n_{beams,obj}$ = 1.25 mm. But, as this error is summing up on every beam, in the dynamic case, the beam n hits the moving object $n \cdot \Delta x_{beam}$ besides the point in the static case. As a consequence, the beams with high number n could end up missing the moving object if the scanning direction is opposite to the movement of the object. If the object moves in the scanning movement direction, this could result in more reflections on the object as in the static case.

This theoretic example shows the necessity of that aspect in physically based modeling. Implementation of rolling shutter and motion blur in a virtual environment could be done e.g. by an artificial movement of the sensor origin, as a first approximation, that considers only the ego movement between two computation steps of the environment.

Aliasing as another typical sampling phenomenon can happen when objects with periodic movements are scanned with a frequency that breaks the Nyquist-Shannon sampling theorem. The objects then seem static or move with a wrong frequency. For lidar sensor systems used in automotive use cases aliasing should not have an effect, because object detection, localization or mapping should not rely on movements rather than size and shape of surrounding objects. Even though, the implementation of correct aliasing in sensor models comes naturally by selecting the same sampling frequency as the real sensor.

VI. CONCLUSION

The contribution of the presented systematic analysis is that it now enables to postulate a certain fidelity of lidar sensor models when knowing the intended function of the model, having the modeling effort and the impact of the implemented effects in mind. Still, even if there have been reported several implementations and modeling approaches of many effects we describe here, neither have been found implementations of correct temporal scan orders, nor has there been fulfilled a systematic analysis beforehand to formulate requirements about what aspects are really needed in specific cases. Consequently, in future work, development of sensor models with beforehand defined requirements and afterwards performed verification and validation will follow. Thereby, research is needed to provide a better insight in the impact of synthetic data on several functions that use lidar point clouds, such as object detection and tracking, sensor fusion or localization and mapping. Their robustness to small deviations in real data will influence the needed fidelity of the sensor models. Additionally, it will be necessary to find applicable holistic validation methods and metrics as well.

REFERENCES

[1] W. Wachenfeld and H. Winner, "The release of autonomous vehicles", in *Autonomous Driving, Technical, Legal and Social Aspects*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 425–449.

[2] S. Sovani, "Simulation accelerates development of autonomous driving", *ATZ worldwide*, vol. 119, pp. 24–29, 9 2017.

[3] German Aerospace Center. (2017). PEGASUS Research Project, Securing Automated Driving efficiently, [Online]. Available: http://www.pegasus-projekt.info/ (visited on 05/04/2018).

[4] J. E. Stellet, M. R. Zofka, J. Schumacher, T. Schamm, F. Niewels, and J. M. Zollner, "Testing of advanced driver assistance towards automated driving, A survey and taxonomy on existing approaches and open questions", in *International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2015.

[5] S. Bernsteiner, Z. Magosi, D. Lindvai-Soos, and A. Eichberger, "Phaenomenologisches Radarsensormodell zur Simulation laengsdynamisch regelnder Fahrerassistenzsysteme, VDI-Bericht 2169", *Elektronik im Fahrzeug*, 2013.

[6] N. Hirsenkorn, T. Hanke, A. Rauch, B. Dehlink, R. Rasshofer, and E. Biebl, "A non-parametric approach for modeling sensor behavior", in *International Radar Symposium (IRS)*, IEEE, 2015.

[7] T. A. Wheeler, M. Holder, H. Winner, and M. J. Kochenderfer, "Deep stochastic radar models", in *Intelligent Vehicles Symposium (IV)*, IEEE, 2017.

[8] T. Hanke, N. Hirsenkorn, C. Van-Driesten, P. Gracia-Ramos, M. Schiementz, and S. Schneider. (2017). Open Simulation Interface, A generic interface for the environment perception of automated driving functions in virtual scenarios, TUM Department of Electrical and Computer Engineering, [Online]. Available: https://www.hot.ei.tum.de/en/forschung/automotive-veroeffentlichungen/ (visited on 05/04/2018).

[9] K. Zimmermann, T. Petricek, V. Salansky, and T. Svoboda, "Learning for Active 3D Mapping", in *International Conference on Computer Vision (ICCV)*, IEEE, 2017.

[10] N. Kaempchen, M. Buehler, and K. Dietmayer, "Feature-level fusion for free-form object tracking using laserscanner and video", in *Intelligent Vehicles Symposium (IV)*, IEEE, 2005.

[11] A. Schaermann, A. Rauch, N. Hirsenkorn, T. Hanke, R. Rasshofer, and E. Biebl, "Validation of Vehicle Environment Sensor Models", in *Intelligent Vehicles Symposium (IV)*, IEEE, 2017.

[12] P. Cao, W. Wachenfeld, and H. Winner, "Perception Sensor Modeling for Virtual Validation of Automated Driving", *it - Information Technology*, vol. 57, 4 2015.

[13] P. Rosenberger, M. Holder, H. Winner, J. Copps, B. Walker, and E. Voigt, *Intended Function as Key for Perception Sensor Model Validation: A Lidar Object Tracking Perspective*. Submitted to 2018 IEEE 21st International Conference on Intelligent Transportation Systems (ITSC), 2018.

[14] J. T. Wendler, T. Menzel, M. Steimle, and M. Maurer, "Sensormodelle für die simulative Absicherung von automatisierten Fahrfunktionen", in *8. Tagung Fahrerassistenz, Einführung hochautomatisiertes Fahren*, (München), 2017.

[15] H. Gotzig and G. Geduld, "Automotive LIDAR", in *Handbook of Driver Assistance Systems, Basic Information, Components and Systems for Active Safety and Comfort*, Springer International Publishing, 2016, pp. 405–430.

[16] S. Svanberg and W. Demtröder, "Optical and spectroscopic techniques", in *Springer Handbook of Lasers and Optics*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 1097–1169.

[17] K. C. Fürstenberg, "Fahrzeugumfelderfassung und Fussgängerschutz unter Nutzung mehrzeiliger Laserscanner", Fakultät für Ingenieurwissenschaften und Informatik, Dissertation, Universität Ulm, 2009.

[18] M. Berk, H.-M. Kroll, O. Schubert, B. Buschardt, and D. Straub, "Zuverlässigkeitsanalyse umfelderfassender Sensorik, Eine stochastische Methodik zur Berücksichtigung von Umgebungseinflüssen am Beispiel von LiDAR Sensoren", in *32. VDI/VW Gemeinschaftstagung Fahrerassistenz und automatisiertes Fahren. VDI Berichte*, vol. 2288, VDI Verlag GmbH, 2016.

[19] S. Hasirlioglu, A. Kamann, I. Doric, and T. Brandmeier, "Test Methodology for Rain Influence on Automotive Surround Sensors", in *International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2016.

[20] S. Hasirlioglu, I. Doric, C. Lauerer, and T. Brandmeier, "Modeling and Simulation of Rain for the Test of Automotive Sensor Systems".

[21] R. Nebuloni and C. Capsoni, "Laser Attenuation by Falling Snow", in *International Symposium on Communication Systems, Networks and Digital Signal Processing, 2008*, IEEE, 2008.

[22] R. H. Rasshofer, M. Spies, and H. Spies, "Influences of Weather Phenomena on Automotive Laser Radar Systems", *Advances in Radio Science*, vol. 9, pp. 49–60, 2011.

[23] S. Hasirlioglu, A. Riener, W. Ruber, and P. Wintersberger, "Effects of Exhaust Gases on Laser Scanner Data Quality at Low Ambient Temperatures", in *International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2016.

[24] J. Kolesik, "Physically Based LIDAR, OptiX, a real time sensor (LIDAR) simulation powered by GPUs", in *GPU Technology Conference Europe 2017, NVIDIA Corporation*, 2017.

[25] Ibeo Automotive Systems GmbH, *Operating Manual ibeo LUX 2010 Laserscanner*.

[26] ZSP Geodetic Systems GmbH, *Trimble 3600-series User Guide*.

[27] Y. Li, Y. Wang, W. Deng, X. Li, Z. liu, and L. Jiang, "LiDAR Sensor Modeling for ADAS Applications under a Virtual Driving Environment", in *SAE-TONGJI 2016 Driving Technology of Intelligent Vehicle Symposium*, ser. SAE Technical Paper Series, SAE International400 Commonwealth Drive, Warrendale, PA, United States, 2016.

[28] Velodyne LiDAR Inc., *HDL-64E S3 High Definition Lidar Sensor, User's Manual and Programming Guide*, 63-HDL64ES3 Rev J, 2017.

[29] M. Holder, P. Rosenberger, F. Bert, and H. Winner, *Data-driven Derivation of Requirements for a Lidar Sensor Model*. 11th Graz Symposium Virtual Vehicle (GSVF), Graz, Austria, 2018.