

A LiDAR system simulator based on raytracing, modeled with metrological parameters and environmental noise

Guilherme Ferreira Gusmão
dept. of Metrology & Tecgraf
PUC-Rio
Rio de Janeiro, Brazil
gusmaof@tecgraf.puc-rio.br

Carlos Roberto Hall Barbosa
dept. of Metrology
PUC-Rio
Rio de Janeiro, Brazil
hall@puc-rio.br

Alberto Barbosa Raposo
Tecgraf
PUC-Rio
Rio de Janeiro, Brazil
abraposo@tecgraf.puc-rio.br

Abstract— The widespread use of point clouds in many applications is prompting many researchers to find new ways to generate such datasets, as the technology is still costly and slow depending on the experiment’s parameters. Virtual scanning, a computer simulation of scanning a real target, is one of these solutions. This paper is a short presentation of the current development of a light detection and ranging (LiDAR) system simulator based on metrological parameters, sensor error model and light scattering treatment.

Keywords— virtual sensor, LiDAR, raytracing, remote sensing, synthetic point cloud

I. INTRODUCTION

3D imaging technologies are changing how the world around us can be captured. One such device is the Light Detection and Ranging system (LiDAR), which allows generation of dense datasets containing georeferenced points called Point Clouds [1]. Unfortunately, depending on the application and size of desired target, the data acquisition can be very costly and slow [2]. This prompted the development of simulators based on the behavior of 3D imaging sensors for scanning virtual scenes, generating as many synthetic point clouds as the processing capacity of the computer permits [2,3]. This solution is being especially researched for the creation of dataset models for machine learning training in autonomous vehicles [4].

II. MOTIVATION

As most of the literature focus on autonomous vehicles applications, there is a lack of general simulators that enable a wider range of experiments. In addition, the literature indicates that there is a need to keep improving the realism of the LiDAR simulations, as many of these simulators ignore or simplify the issues that happen during real scanning.

III. RELATED WORK

Reference [3] shows “a sensor measurement model for an automotive LiDAR sensor”. They used raytracing to simulate the sensor measurements, as the working principle of the sensor is very similar with the rendering algorithm. This “enabled real-time generation of a point cloud within the framework of an automotive simulator” [3].

Reference [4] published a “novel open-sourced method to extract point clouds with ground truth annotations from a simulator automatically”. The simulation sensor can be configured to simulate many real 3D imaging devices. It also justifies the use of synthetic data in deep learning training.

The base software used in this paper is found in [5]. It presents a LiDAR system simulator with parallel raytracing modelled with metrological parameters, being part of the proposed solution here.

IV. PROPOSED SOLUTION

To fill the gap in the literature, a LiDAR based simulator built upon the Nvidia OptiX raytracing engine is being developed [5]. The parallel raytracing algorithm was repurposed to behave following metrological parameters of a real LiDAR. In this way, the virtual sensor, a mathematical model of a sensor behavior, shoots rays with the same angle, resolution and detection range.

The OptiX engine also permitted the creation of a scene builder. The scene is built with 3D models of designed target object/area and positioning parameters files. The user defines the scanning positions of the virtual sensor as well [5]. The software then tracks the rays’ collisions with the scene and generates a point cloud for each defined sensor position in the virtual world.

Each point in the point cloud carries information about 3D position, normal vector, color and ID (a unique number that identifies from which 3D model belongs the point). The ID information is especially useful for segmentation procedures, as different objects in the cloud are already highlighted [6], as illustrated in Fig. 1.

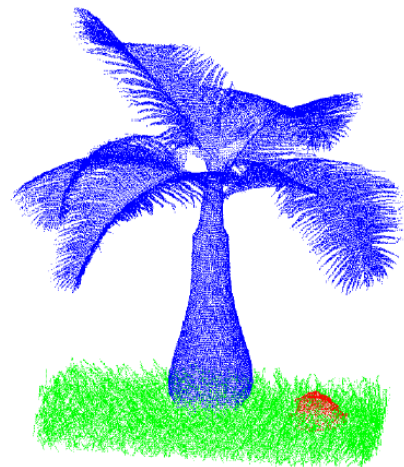


Fig. 1. A synthetic point cloud generated with the LiDAR simulator. The color is based on which object the point belongs to [5].

To improve the realism of the generation of point clouds two approaches are being researched:

A. Sensor Error Model:

As mentioned in [3], this method “aims to reproduce the statistical characteristics of errors, i.e. deviations between the perceived and true values” of a sensor system.

To obtain the statistical characteristics of errors a series of scanning experiments are presently being made. From the data obtained, the mean and standard deviation can be found for each ray shooting angle.

The next step will be decided on how to best construct the error distribution given the information acquired. After this, the statistical characteristics of errors will be able to influence on how the rays shot in the simulator will be registered by the software, adding random and systematic errors in the synthetic point cloud.

B. Light Scattering Model:

The LiDAR sensor is a time-of-flight device that uses a laser and photodetector to detect information from the designed target object/area [1,2]. As the laser is a light source, its trajectory is also dependent of the optical properties of the media and objects in its way. These properties can cause reflection, scattering and/or absorption of photons, influencing the sensor output [1]. For example, the sensor could register an artifact (a false measurement of a point) or even holes (no information) in the point cloud.

These phenomena, if modeled in the simulator, would enhance the synthetic point cloud realism during generation [2,7]. The literature shows two methods in this case:

a) Mie scattering theory: As in most situations of LiDAR scanning, the scattering and absorbing particles are much larger than the light wavelength, the Mie solution for the Maxwell equation is the best way to solve this problem. The solution is very lengthy and will not be explained here. Software based on this method try to perfectly recreate photon propagation in a media. Although a very powerful tool for solving the related problem, it has a high computational cost [7].

b) Monte Carlo (MC) method: Using randomly generated samples, the MC statistical technique calculates numerical results from both deterministic or non deterministic nature. In the case of light scattering problem, the MC can be used as well to calculate the photon propagation, given random samples of the photon initial position in a defined time gate, for example. This type of solution can be calculated faster than the physical one, especially if using parallel programming (GPU) in the algorithms [8].

The MC method is being researched for light scattering treatment in this work.

V. PRELIMINARY RESULTS

Reference [5] shows most of the preliminary results of this work thus far. A comparison with a real sensor validates that the software can generate synthetic point clouds following the characteristics of a chosen LiDAR. In Fig. 2, the plotted graph demonstrates the sensor hits on the surface of a can, where ‘x’ is the virtual one and ‘o’ the real one.

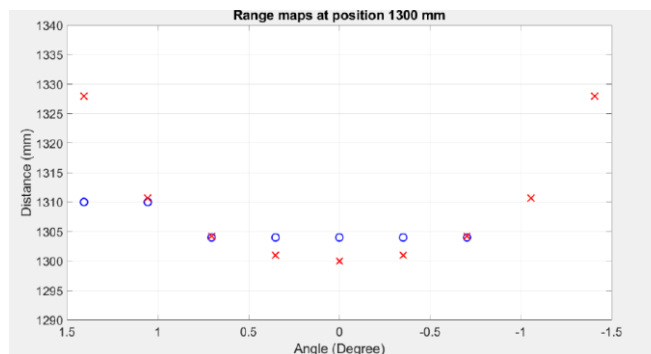


Fig. 2. The rays hit detection from the real LiDAR sensor (‘o’) and virtual sensor (‘x’) at position 1300 mm (distance vs ray shooting angle) [5]

It was also noted that, as the simulator had no noise added yet, the difference between real and virtual scanning were inside the uncertainty error range specified in the LiDAR sensor datasheet. This is a good indication of the virtual sensor modelling.

VI. CONCLUSION

The work in this paper already has achieved the development of an adequate virtual sensor modeled after metrological parameters of a LiDAR sensor. This resulted in synthetic point clouds generated with some level of realism, as the points detected in the simulation have the same angle of the real scanning.

The next step is to integrate the sensor error model and MC scattering simulation in development to the simulator.

ACKNOWLEDGMENT

The authors thank for the financial support provided by the Brazilian funding agencies CNPq, CAPES, FINEP and FAPERJ.

REFERENCES

- [1] M. Massot-Campos and G. Oliver-Codina, “Optical Sensors and Methods for Underwater 3D Reconstruction,” *Sensors*, vol. 15, no. 12, pp. 31525–31557, 2015.
- [2] A. Tallavajhula, “Lidar simulation for robotic application development: modeling and evaluation,” Ph.D. dissertation, The Robotics Institute, Carnegie Mellon University, Pittsburgh, 2018
- [3] T. Hanke, A. Schaermann, M. Geiger, K. Weiler, N. Hirsenkorn, A. Rauch, S.-A. Schneider, and E. Biebl, “Generation and validation of virtual point cloud data for automated driving systems,” 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), 2017.
- [4] F. Wang, Y. Zhuang, H. Gu and H. Hu, “Automatic Generation of Synthetic LiDAR Point Clouds for 3-D Data Analysis,” in *IEEE Transactions on Instrumentation and Measurement*, vol. 68, no. 7, pp. 2671-2673, July 2019.
- [5] G. Gusmão, “A LiDAR system simulator using parallel raytracing and validated by comparison with a real sensor,” to be present at CBM 2019.
- [6] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, “PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation,” 2017 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [7] N. Hirsenkorn, H. Kolsi, M. Selmi, A. Schaermann, T. Hanke, A. Rauch, R. Rasshofer and E. Biebl, “Learning Sensor Models for Virtual Test and Development,” in *Workshop Fahrerassistenz und automatisiertes Fahren*, 2017.
- [8] Q. Fang and D. A. Boas, “Monte Carlo Simulation of Photon Migration in 3D Turbid Media Accelerated by Graphics Processing Units,” *Optics Express*, vol. 17, no. 22, p. 20178, 2009