Data Simulation of an Airborne LIDAR System

Seongjoon Kim *[a], Seonghong Min[a], Geunhan Kim[a], Impyeong Lee [a], Chulmin Jun [a]
[a] Dept. of Geoinfomatics, The University of Seoul, 90 Jeonnong-dong Dongdaemun-gu Seoul, Korea

ABSTRACT

An airborne LIDAR (LIght Detection And Ranging) system can rapidly generate 3D points by densely sampling the terrain surfaces using laser pulses. The LIDAR points can be efficiently utilized for automatic reconstruction of 3D models of the objects on the terrain and the terrain itself. The data simulation of such a LIDAR system is significantly useful not only to design an optimal sensor for a specific application but also to assess data processing algorithms with various kinds of test data. In this study, we thus attempted to develop data simulation software of an airborne LIDAR system generally consisting of a GPS, an IMU and a laser scanner. We focused particularly on the geometric modeling of the sensors and the object modeling of the targets and background. Hence the data simulation software has been developed using these models. For the geometric modeling, we derived the sensor equation by modeling not only the geometric relationships between the three modules, such as a GPS, an IMU and a laser scanner but also the systematic errors associated with them. Moreover, for rapid and effective simulation, we designed the data model for both targets and background. We constructed the model data by converting the VRML formatted data into the designed model and stored these data in a 3D spatial database that can offer more effective 3D spatial indexing and query processing. Finally, we developed a program that generates simulated data along with the system parameters of a sensor, a terrain model and its trajectories over the model given.

Keywords: Simulation, LIDAR, Multi-Sensor System, Sensor Model, Geometric Model, Sensor Equation

1. INTRODUCTION

A LIDAR system is a well-accepted tool that can acquire 3D point clouds of the terrain surfaces fast and precisely. As an active sensor, LIDAR measures the ranges from the sensor to the dense points on the terrain surfaces using laser pulses. Even though LIDAR data have some drawbacks; for example, hazy breaklines, relatively low horizontal accuracy and no inherent redundancy, it has been widely used because of its time and cost-effectiveness, and its reliability. Many studies have been performed to develop more efficient algorithms to generate 3D spatial information using LIDAR data. Many algorithms have been developed for digital surface model (Schenk, 2001), segmentation (Lee, 2002), building/road reconstruction (Haala and Brenner, 1999; Rottensteiner, 2003; Vosselman 2002), forest management (Perrson et al, 2002) and so on. Recently, many researchers have tried to extract more information from the full-waveforms recorded by a LIDAR system (Wagner, 2006).

Data simulation offers a more easy and reliable method to verify these algorithms using simulated LIDAR data with various properties in diverse environments. For instance, NASA has used the simulation software to design and verify the hardware of GLAS (Geoscience Laser Altimeter System) on ICESat spacecraft (Filin et al, 2000). In a similar way, CAIL (Center for Advanced Imaging LADAR) in University of Utah developed simulation software, "LadarSIM", and performed various experiments related with LADAR (Pack et al, 2006). The earlier studies on the simulation of LIDAR data had been performed by only a few researchers. These literatures focused on full-waveform of a returned beam and, in the scope of geometric sensor modeling, the methods were not fully satisfactory (Liviue, 2007; Rastislav, 2004), for example, the applied input data was defined in the raster formatted (Antero 2007).

In this paper, we attempted to develop data simulation software for LIDAR systems and data modeling for the target objects. Although it is essential to perform modeling its geometric, radiometric and electronic aspects for data simulation, we mainly focused on the geometric modeling related with three main subsystems such as GPS, IMU, and Laser scanner.

*sinus7953@uos.ac.kr; phone +82-2-2210-5422; lsm.uos.ac.kr

2. OVERVIEW

In generally, a LIDAR simulation program comprises several components, as shown in Fig. 1. *Sensors*, which consist of GPS, IMU and laser scanner, can compute the origin and the direction of laser beams by geometrically integrating the data from the individual sensors. *Target/Terrain* is used as the input data for the simulation program. This module obtains not only the shapes of objects but also the surface material properties. *Atmosphere* provides attenuation when the beams travel in atmosphere. *Beam Interaction* operates by finding the surface of the target which intersects with the beam ray, then computing the intersection position and the flight time of the beam, and thus determining the beam power with the target reflective properties and attenuation due to the atmosphere. Finally, the models of the returned pulses collected by the receiver in the time domain are generated. The *Data processing* module deals with full-waveforms–signal processing and image processing.

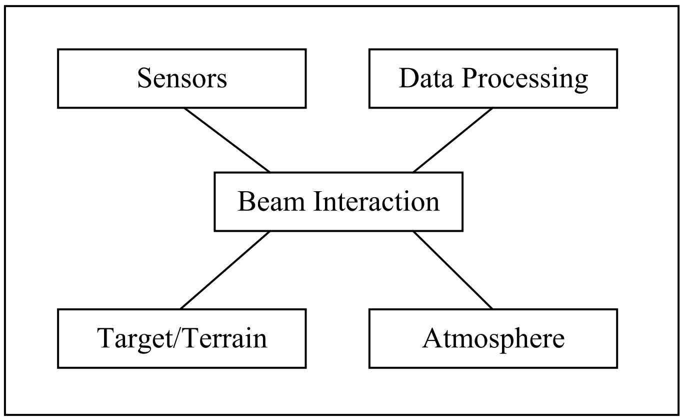

Fig. 1. Essential components in LIDAR simulation.

To develop a simulation program which can perform the tasks of all modules stated above, three stages of modeling are required as depicted in Fig. 2. In the first stage, we establish the relationship between sub-modules of the LIDAR system: GPS, IMU, laser scanner and the target by considering the systematic errors. In the second stage, we calculate the power of the beam ray when it travels in the atmosphere and reflects from the target surface due to the atmospheric attenuation and the target reflective properties. In the third step, we model the full-waveform and the sum of the returned pulses in the time domain using the flight time of the beam computed by geometric modeling and the power by radiometric modeling.

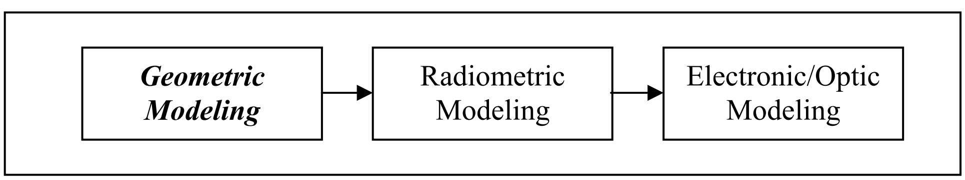

Fig. 2. Modeling parts for LIDAR simulation.

Since we are focusing on geometric modeling to develop the LIDAR simulation program and now in progress of performing the radiometric and electronic modeling, in this paper, we will not discuss these modeling but geometric modeling. Fig. 3 illustrates three main processes for the geometric modeling and the detailed process of each main process.

Sensor modeling: We derived the sensor equations based on the geometric relationship between sensors and the systematic errors associated with the sensors.

Target modeling: There exist several methods for object modeling in the scope of GIS. Among them, we selected one that suits our simulation method. We then converted the terrain/building objects into the VRML format and stored in a 3D Spatial DBMS.

Beam ray modeling: The sensor modeling offers the ability to determine the origin and the direction of a beam. We search the surface of object which intersects with the beam ray using the 'ray-tracing' algorithm.

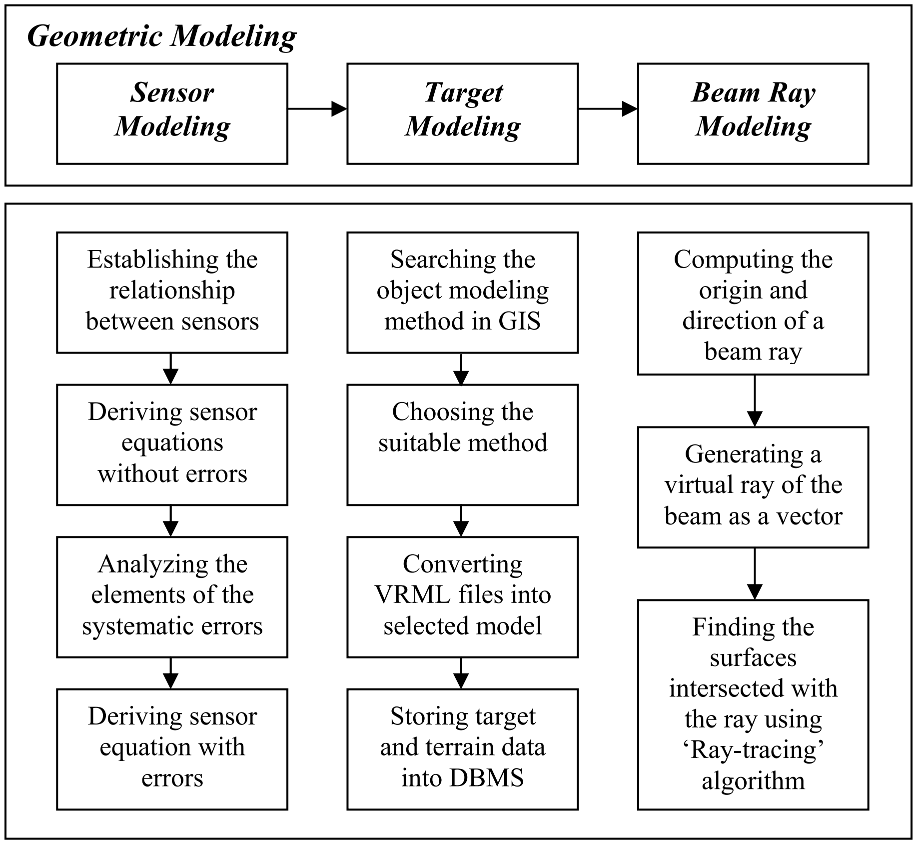

Fig. 3. Modeling and processing decomposition

This paper is organized into separate sections to describe the detailed description of the proposed method. In Section 3, we derived the sensor equation representing the geometric relationships between the sensors and the systematic errors. Section 4 describes the design of the input data model of the background and objects on the terrain in a vector format to offer fast and effective performance to the simulation. In this study, we used a 3D spatial database that can store the data and offer more effective 3D spatial indexing and querying processes. The detailed method of the data simulation is explained in Section 5. Section 6 discusses the experiments that applied the designed terrain model to the suggested simulation method. Finally, we discussed our study and future works in Section 7.

3. SENSOR MODELLING

To generate the simulated LIDAR data, it is necessary to model the operation of LIDAR system. The system consists of GPS, IMU and laser scanner. We can find the position and the direction where the laser pulse triggered with GPS and IMU, and the range from the origin to the terrain by detecting the flight time of the pulses. The modeling of the LIDAR system has to include these sensors' activities. To model the system, we derived a sensor equation. It is a mathematical representation of geometric and timely relationship between the sensors and the position of a 3D point on a surface of a target, where a laser pulse transmitted from the sensor is reflected. The geometric correspondence means that the coordinate systems of GPS, IMU and laser scanner in a LIDAR system should be redefined in a common coordinate system. The timely correspondence is that the time of each module should be synchronized with each other (Lee, 2003). Fig. 4 shows each coordinate system of the three modules and their geometric relationships.

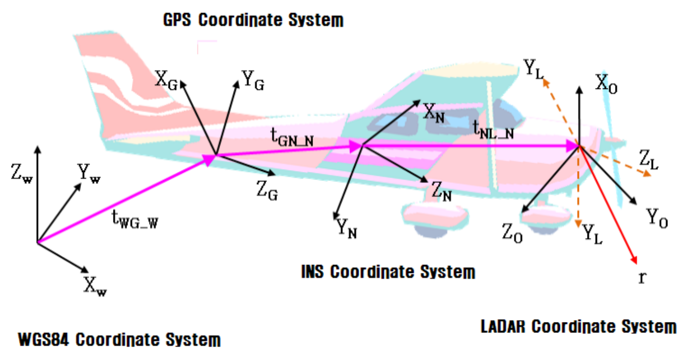

Fig. 4. Geometric relationship among the coordinate systems of the individual sensor (Min et al., 2008).

Based on these relationships, the sensor equation of a LIDAR system without considering the systematic errors can be derived as Eq. (1). The parameters are explained in Table 1. '0', 'L', 'N', 'G' and 'W' are used as subscriptions to indicate the corresponding coordinate system while R and t mean a rotation matrix and a translation vector, respectively.

$$P_W = R_{GW} R_{NG} (R_{LN} R_{0L} u_z r + t_{NL_N} + t_{GN_N}) + t_{WG_W} \quad (1)$$

Table. 1. The definition and description of the variables embedded in the sensor equation.

Variables	Definition and Description
P_w	The true values of laser pulse's reflected point with respect to the WGS84 coordinate system
u_z	The unit vector (0,0,1) along the z-axis with respect to the initial LIDAR coordinate system
r	The range from the starting point of the transmitting laser pulse to its reflected point on the target surface
R_{0L}	The rotation matrix used for the transformation from the initial LIDAR coordinate system to the LIDAR coordinate system
R_{LN}	The rotation matrix used for the transformation from the LIDAR coordinate system to the INS coordinate system
R_{NG}	The rotation matrix used for the transformation from the INS coordinate system to the GPS coordinate system
R_{GW}	The rotation matrix used for the transformation from the GPS coordinate system to the WGS84 coordinate system
t_{NL_N}	The translation vector connected from the origin of the INS coordinate system to the origin of the LIDAR coordinate system represented in the INS coordinate system
t_{GN_N}	The translation vector connected from the origin of the GPS coordinate system to the origin of the INS coordinate system represented in the GPS coordinate system
t_{WG_w}	The translation vector connected from the origin of the WGS84 coordinate system to the origin of the GPS coordinate system represented in the WGS84 coordinate system

From *a priori* knowledge, all sensors in LIDAR systems possess some systematic errors. However, Eq. (1) does not consider any errors. As a result, we need to identify the error sources and derive their parametric models. There are two types of the errors related to LIDAR. One is caused from each individual sensor errors and the other is from the integration errors. The examples of the individual sensor are the drift errors in IMU, bias errors in GPS and range errors in laser scanner, which are the systematic errors of each sensor. The integration errors come from geometric and timely integration among sensors, GPS, IMU and laser scanner. The examples of the integration errors are the mounting error (variation of positions or directions of sensors from external shocks) and interpolation error (combining data among sensors that have different time cycles). Schenk (2001) and Lee (2003) described more details of these two errors in their researches.

Based on the sensor equation, those six variables may possess their own error as presented in Table 2. By considering the major errors, we derived the sensor equation with systematic errors in Eq. (2) where P_W^* is the observation value of P_W. In this paper, the systematic errors are represented with a notation 'Δ'.

Table. 2. The definition and description of the variables embedded in the sensor equation.

Error variables	Definition
Δr	r bias
ΔR_{0L}	R_{0L} bias
ΔR_{LI}	R_{LI} bias
ΔR_{IG}	R_{IG} bias
Δt_{GL_I}	t_{GL_I} bias
Δt_{WG_W}	t_{WG_W} bias

$$\begin{aligned} P_W^* = {} & R_{GW} \Delta R_{NG} R_{NG} (\Delta R_{LN} R_{LN} \Delta R_{0L} R_{0L} u_z (r + \Delta r) \ldots \\ & + t_{NL_N} + \Delta t_{NL_N} + t_{GN_N} + \Delta t_{GN_N}) + t_{WG_W} + \Delta t_{WG_W} + \Delta t_{TB}, \end{aligned} \tag{2}$$

4. DATA SIMULATION

The data simulation requires the system parameters of a LIDAR system such as flight parameters, laser emitter parameters, target/background models, etc. With these input data, the computation of each position of the 3D points where laser pulses are reflected can be performed by following procedures.

1. Determine the origin of a transmitted laser pulse by computing the position of the platform at the firing time of the laser pulse using its flight path and velocity.

2. Calculate the direction of the laser pulse using the instantaneous scan angle with the assumption on the attitude of the platform.

3. Generate a virtual straight ray that the laser pulse travels.

4. Search for the facet of the object on the ground where the virtual ray intersects using a 'ray-tracing' algorithm.

5. Compute the 3D coordinate of the point at which the ray intersects with the facet.

6. Compute the true value of the range from the laser scanner to the surface of the target.

7. Using Eq. (2), compute the 3D position of the erroneous intersection point with the true range from Step 6 and the given systematic errors.

In these procedures, we used the 'ray-tracing' algorithm to find the facet which intersects with a laser beam. It is not efficient to inspect the intersection one by one since there are many target/background objects in which each object has a number of facets, for examples, about 3 millions in an area of 500m by 500m. To avoid this problem, we establish a grid structure in which all the facet are linked to their corresponding cells of the grid. We can then obtain the candidate facets by applying the 'ray-tracing' algorithm to search for the grid cell that can be intersected with the ray. Fig. 5 illustrates the 'ray-tracing' algorithm. From the figure, the cell intersected with the ray can be detected by reducing the vertical and horizontal range from one to another.

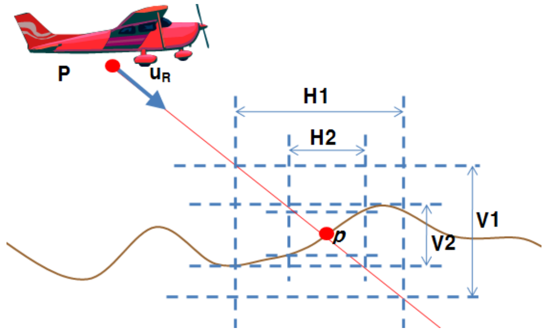

Fig. 5. The basic concept of the ray-tracing algorithm (Kim, 2008).

5. TARGET AND BACKGROUND MODELING

In order to perform the simulation, it is necessary to design the target and background data model. Thus, we attempt to design the model structure for storing the terrain and building objects. There are many existing data models to represent, record and visualize real world, for example, Constructive Solid Geometry and Boundary Representation. In this work, we select the B-rep structure based on polyhedron for building modeling since it is suitable for the 'ray-tracing' algorithm which works by simply searching for a polygon that intersects with a laser beam. In a similar way to the building model, terrain is defined as TIN structure based on triangular shapes. Moreover, we employed a spatial database, PostgreSQL in our work. PostgreSQL is a spatial database system that can store a huge set of 3D spatial data with efficient spatial indexing scheme and can provide rapid spatial search and access. Generally, there are four models for use with DBMS (Data Base Management System):

Type 1. Solid-Face-Edge-Node

Type 2. Solid-Face-Edge

Type 3. Solid-Face-Node

Type 4. Solid-Face

The simulation method does not need topological relationship among primitives (e.g. solid, face, edge and node). However, a set of positions of polygons and spatial query are required to perform rapidly. As a result, we applied the Type 4 model in our work. Fig. 6 presents the UML diagram of Type 4, and Fig. 7 shows the building model defined by the suggested method.

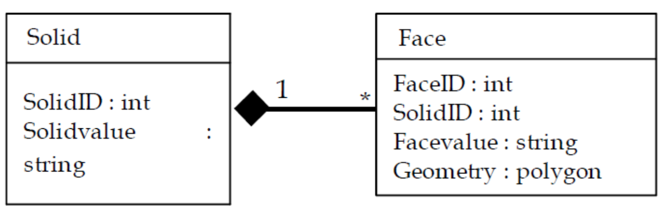

Fig. 6. UML class diagrams describing storage of a polyhedron of Type 4 (Kim, 2009).

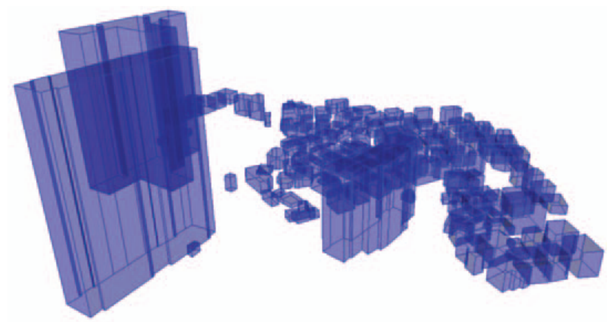

Fig. 7. Building models as formatted in B-rep (Kim, 2007).

6. EXPERIMENT

The proposed method has been tested with a real 3D city model. The city model is a part of Yeongdeungpo-gu in Seoul, South Korea. It is from the digital maps (1:1000) and the draft maps (1:1000). The area consists of the ground and building models. The ground model is defined as a TIN structure and the building models are polyhedral structures. The number of polygons in the building models is about 37 thousands and ground model is about 20 thousands. Fig. 8 shows the 3D city model as the input data.

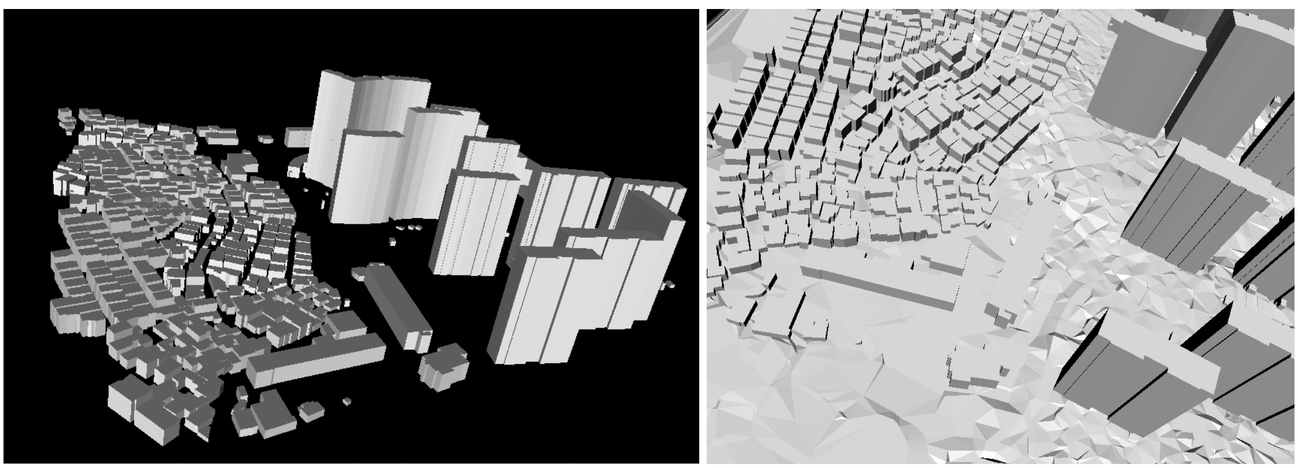

Fig. 8. 3D city model of the test area.

To simulate LIDAR data, the system and flight parameters are need to be specified. The LIDAR system consists of three sensors which are GPS, IMU and laser scanner mounted with a platform. Each sensor has its own system parameters. For examples, the pulse frequency, scan rate and scan angle of the laser scanner which are provided in Table. 3. The pulse rate is the number of laser pulses transmitted per second and the scan rate is the number of scans per second. The scan angle indicates the scan range. In this case, the 20 degree of scan angle means that the system scans -10 ~ 10˚ from the nadir direction. The systematic errors for each sensor should be also specified. As the flight parameters, the flight path, attitude and velocity are given. The flight trajectory of the platform is assumed to be a straight line starting from (100, 240, 1000) to (400, 250, 1000) m and its flight velocity as 100 m/s.

Table 4 shows the systematic errors as additional input parameters. We determined the GPS errors to be 1 m and the others value to 0 in order to compare the difference between the results with errors and without errors. The other parameters not specified in Table 3, above than those error parameters incorporated into Eq. (2), are negligible comparing to the specified ones and thus assumed to be zero.

Table. 3. The systematic parameters of the laser scanner

Parameter	Unit	Value
Pulse Rate	kHz	40
Scan Rate	Hz	200
Scan Angle	deg	20

Table. 4. The systematic parameters of the laser scanner

Bias	Symbol	Unit	Value
GPS bias, x	Δt_{WG_W}	m	1
GPS bias, y		m	1
GPS bias, z		m	1
INS bias, omega	ΔR_{NG}	deg	0
INS bias, phi		deg	0
INS bias, kappa		deg	0
Range bias	Δr	m	0

We implemented the suggested method using the C++ language and PostgreSQL. The simulation output is a set of a huge number of 3D points, as shown in Fig. 9 and Fig. 10. The simulator generates about 120,200 points for the running time of 3 seconds. The point density of the simulated LIDAR data is 1.08 points/m^2; its point spacing is 1.72 m; the range of its x-coordinate value is 94 ~ 406 m and the range of its y-coordinate value is 68 ~ 423 m.

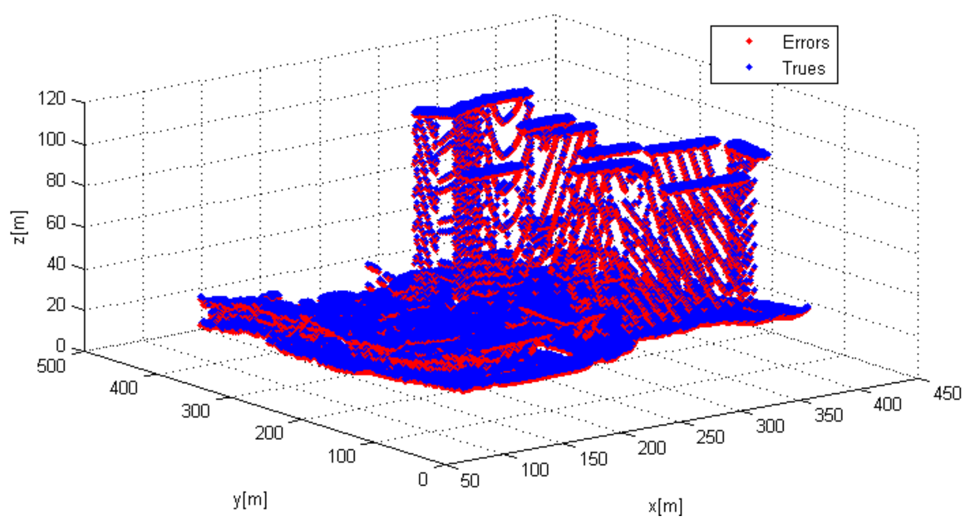

Fig. 9. Visualization the generated simulation data (red: with errors, blue: with no errors).

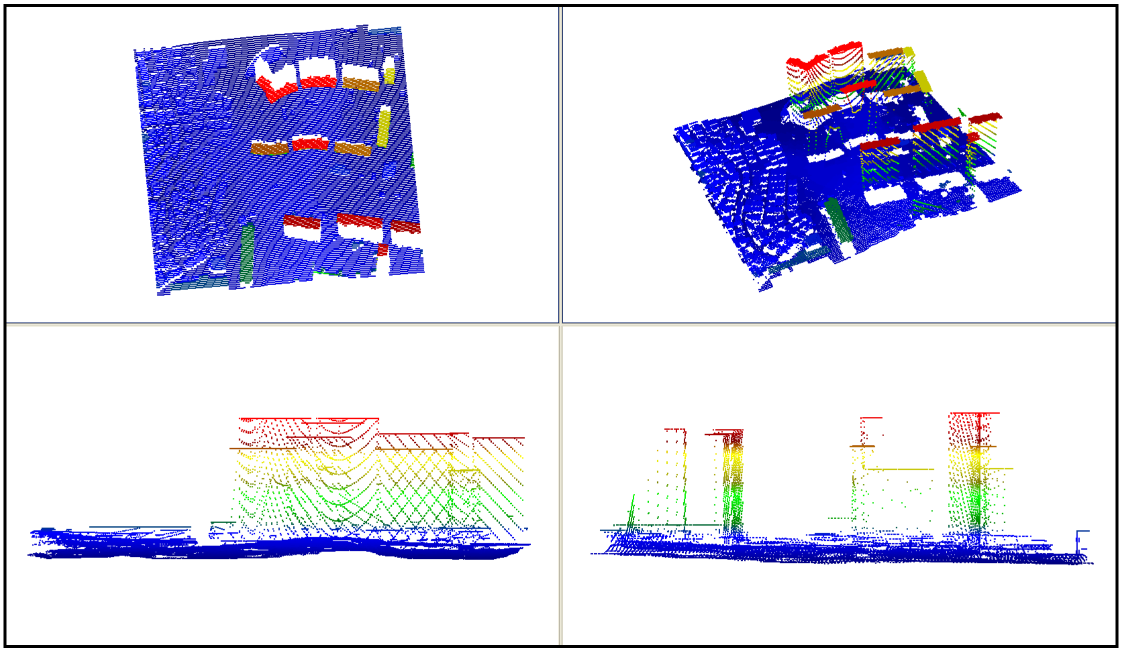

Fig. 10. The simulated LIDAR data without systematic errors

To verify the proposed method of the simulation, we checked the coordinate differences between the true laser points and the simulated laser point when the only GPS errors with 1 m in each axis are assumed. The distributions of the differences in each axis are shown in Fig. 11. As expected, all differences are concentrated on 1 m. A few difference rather than 1 m are caused from the computational errors on the intersection points between the ray of a beam and a facet.

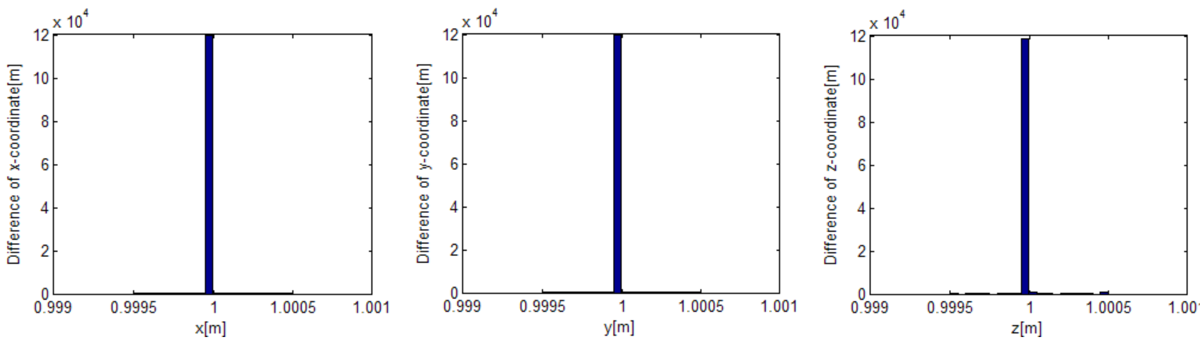

Fig. 11. Histograms of the differences between the true and simulated laser points.

7. CONCLUSIONS

We attempted to perform the geometric modeling of the LIDAR system and derived the sensor equations with systematic sensor errors. We proposed a simulation method based on our derived sensor equations and also proposed an optimal data (target/background) modeling for efficient 3D spatial query. The suggested method has been tested with real urban models and could successfully generated simulated LIDAR data along with the system parameters. The results of this study will be useful for the system design of LIDAR sensors and the development of relevant algorithms as well as for the assessment of LIDAR applications. Furthermore, since the simulation will be able to predict the data quality of the sensors to be acquired before the real flight, it will be helpful to find the optimized flight parameters suitable to user requirements.

The future work will focus in the sensor model in radiometric and electronic aspects. Through these modeling, our LIDAR simulator will be able to deal with full-waveform of a beam and its power. At the present status, we are in progress to model full-waveforms by summing each returned pulses of a number of sub-beams.

8. ACKNOWLEDGEMENT

This research was supported by the Defense Acquisition Program Administration and Agency for Defense Development, Korea, through the Image Information Research Center at Korea Advanced Institute of Science & Technology under the contract UD070007AD.

REFERENCES

[1] Filin, S. and Csatho, B., "An Efficient algorithm for the Synthesis of Laser Altimetry Waveforms," BPRC Technical Report No. 2000-02, The Ohio State University, Columbus, OH, p.27 (2000).

[2] Haala, N., Brenner, C., "Extraction of buildings and trees in urban environments," ISPRS Journal of Photogrammetry and Remote Sensing, Vol. 54, pp. 130-137 (1999).

[3] Kim, H.,Kim, G.,Jun, C.,Y, H., "Comparing DBMS-based approaches for representing 3D building objects," The 5th GRSS/ISPRS Workshop on Data Fusion And Remote Sensing over Urban Areas (2009).

[4] Kim, S., Min, S., Lee, D., Park, J., Lee, I., "Automatic Generation of 3D Building Models using a Draft Map," Journal of the Korean society for geospatial information system, vol.15 no.2 : 3-14 (2007).

[5] Kim, S., Min, S., Lee, I., Choi, K., "Geometric Modeling and Data Simulation of an Airborne LIDAR System," Journal of the Korean society for geospatial information system, vol.26 no.3 : 311-320 (2008).

[6] Lee, I. , "Perceptual Organization of Surfaces," Ph.d. Dissertation, Dept. of Civil and Environmental Engineering, OH, USA (2002).

[7] Lee, I., "Verification and Calibration of LIDAR Data based on Simulation with High-resolution DEMs," KSCE Annual Conference, pp. 4436~4440 (2003).

[8] Min, S., Kim, S., Lee, I., "Data Simulation of LIDAR Sensor: Focusing on Geometric Modeling," International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, ISPRS, Beijing, China, Vol. 37-B1, pp. 439-443 (2008).

[9] Pack, R. T., Saunders, D., Fullmer, R., Budge, S., "The simulation of automatic ladar sensor control during flight operations using USU LadarSIM Software", Proceedings of SPIE Vol. 6214, 62140L (2006).

[10] Persson, Å., Holmgren, J. & U. Söderman, "Detecting and measuring individual trees using an airborne laser scanner," Photogrammetric Engineering and Remote Sensing, 68, 925-932 (2002).

[11] Rottensteiner, F., "Automatic generation of high-quality building models from Lidar data," IEEE CG&A 23(6), pp. 42-51 (2003).

[12] Schenk, T., "Modeling and analyzing systematic errors of airborne laser scanners," Technical Notes in Photogrammetry No. 19, The Ohio State University, Columbus, OH, USA (2001).

[13] Vosselman, G., "Fusion of Laser Scanning Data, Maps and Aerial Photographs for Building Reconstruction," IEEE International Geoscience and Remote Sensing Symposium and the 24th Canadian Symposium on Remote Sensing, IGARSS'02, Toronto, Canada, June 24-28 (2002).

[14] Wagner, W., Ullrich, A., Ducic, V., Melzer, T., and Studnicka, N., "Gaussian decomposition and calibration of a novel small-footprint full-waveform digitising airborne laser scanner," ISPRS Journal of Photogrammetry and Remote Sensing, 60(2), pp. 100-112 (2006).