

항공라이다시스템의 기하모델링 및 데이터 시뮬레이션 Geometric Modeling and Data Simulation of an Airborne LIDAR System

김성준¹⁾ · 민성홍²⁾ · 이임평³⁾ · 최경아⁴⁾

Kim, Seong Joon · Min, Seong Hong · Lee, Im Pyeong · Choi, Kyung Ah

Abstract

A LIDAR can rapidly generate 3D points by densely sampling the surfaces of targets using laser pulses, which has been efficiently utilized to reconstruct 3D models of the targets automatically. Due to this advantage, LIDARs are increasingly applied to the fields of Defense and Security, for examples, being employed to intelligently guided missiles and manned/unmanned reconnaissance planes. For the prior verification of the LIDAR applicability, this study aims at generating simulated LIDAR data. Here, we derived the sensor equation by modelling the geometric relationships between the LIDAR sub-modules, such as GPS, IMU, LS and the systematic errors associated with them. Based on this equation, we developed a program to generate simulated data with the system parameters, the systematic errors, the flight trajectories and attitudes, and the reference terrain model given. This program had been applied to generating simulated LIDAR data for urban areas. By analyzing these simulated data, we verified the accuracy and usefulness of the simulation. The simulator developed in this study will provide economically various test data required for the development of application algorithms and contribute to the optimal establishment of the flight and system parameters.

Keywords : LIDAR, LADAR, Simulation, Sensor Model, Geometric Modeling, Ray-tracing

초 록

라이다는 레이저 펄스를 이용해서 대상표면에서 추출된 조밀한 3차원 좌표를 신속하게 취득할 수 있어서 대상물의 3차원 모델을 자동으로 생성하는 것에 효과적으로 이용된다. 이러한 장점에 힘입어 국방 및 안보 분야에서 특히 지능형 유도 미사일과 유무인 정찰기에 라이다의 적용이 빠르게 증가하고 있다. 이에 본 연구는 라이다의 활용가능성을 사전에 검증하기 위해 라이다데이터를 시뮬레이션을 통해 생성하는 것을 목표로 한다. 이를 위해 라이다를 구성하는 GPS, IMU, LS의 기하모델링을 통해 레이저 펄스의 3차원 좌표를 계산하는 센서 방정식을 유도하였고, 이를 기반으로 주어진 시스템변수, 비행경로 및 자세, 시스템오차, 기준 DEM 등을 입력으로 모의 라이다데이터를 생성하는 프로그램을 개발하였다. 도심지역의 모의 라이다데이터 생성에 적용하였고, 생성된 데이터를 분석하여 시뮬레이션의 정확도 및 활용성을 확인하였다. 본 연구에서 개발된 시뮬레이터는 활용 알고리즘 개발을 위해 필요한 다양한 실험 데이터를 경제적으로 제공하고, 나아가 최적의 비행 및 시스템 변수의 설정에 기여할 수 있다.

핵심어 : 라이다, 레이다, 시뮬레이션, 센서모델, 기하모델링, 레이트레이싱

1. 서 론

라이다시스템은 지표면이나 대상물의 3차원 좌표를 조

밀하고 신속하게 취득하는 센서이다. 라이다시스템은 크게 GPS(Global Positioning System), IMU(Inertial Moment Unit) 및 LS(Laser Scanner)로 구성되며, 구성된 개별센서

1) 정회원 · 서울시립대학교 대학원 공간정보공학과 박사과정(E-mail:sinus7953@uos.ac.kr)
2) 정회원 · 서울시립대학교 대학원 공간정보공학과 석사과정(E-mail:minssy0306@uos.ac.kr)
3) 연결저자 · 정회원 · 서울시립대학교 도시과학대학 공간정보공학과 조교수(E-mail:iplee@uos.ac.kr)
4) 정회원 · 서울시립대학교 대학원 공간정보공학과 박사과정(E-mail:shale@uos.ac.kr)

로 취득한 위치, 자세, 거리를 조합하여 레이저펄스가 반사된 지점의 3차원 좌표를 생성한다. 라이다시스템은 대상물의 표면에서 조밀하게 취득된 3차원 좌표를 제공하기 때문에 대상물의 정밀한 3차원 모델링에 효율적으로 이용될 수 있다. 예를 들어, 도시모델의 생성, 변화의 탐지(Change Detection), 삼림지역의 DTM(Digital Terrain Model)의 생성, 나무의 높이 결정, 해안 지형의 관리 등 다양한 분야에서 활용이 되고 있다. 또한, 최근에는 국방 분야에서 무인정찰기나 지능형유도미사일에 탑재되어 활용되고 있다(Lockheed Martin, 2006; Allen, 2005). 예를 들어, 지능형유도미사일인 LOCAAS(Low Cost Autonomous Attack System)에 탑재되어 표적에 대한 정밀 타격을 위한 미사일 유도 목적으로 사용되고 있다.

이와 같이 여러 분야에서 라이다데이터의 활용이 증가하면서, 활용에 적합한 다양한 데이터 처리 알고리즘의 개발이 요구된다. 개발된 알고리즘을 시험 및 검증하기 위해서는 다양한 형태의 라이다데이터가 필요하다. 라이다데이터는 촬영할 당시의 비행고도(flight height), 이동경로(trajjectory), 펄스레이트(pulse rate), 스캔앵글(scan angle) 등의 시스템 변수에 따라 많은 차이가 나타나기 때문이다. 그러나 현실적으로 알고리즘의 성능을 평가하기 위해서 동일지역에서 다양한 시스템 변수로 촬영된 라이다데이터를 획득하는 것은 시간/경제적으로 비효율적이다. 예를 들어 국방 분야에서 라이다 센서를 장착한 유도미사일을 개발하기 위해 자동표적인식 알고리즘 시험을 위해 실제 미사일을 발사하는 것은 위험하고 많은 비용이 소요된다.

이에 본 연구는 라이다데이터 처리 알고리즘의 성능평가를 위해 다양한 환경에서 취득될 수 있는 실험데이터를 주어진 DEM으로부터 시뮬레이션을 통해 생성하는 방법

을 제시한다. 먼저 라이다시스템의 구성요소에 대한 기하학적 모델링을 통해 레이저펄스의 3차원 좌표를 계산하기 위한 센서방정식을 수립한다. 센서방정식을 이용하여 주어진 DEM 영역위에 플랫폼이 이동하는 경로에 따라 취득되는 라이다데이터를 모의 생성한다. 이러한 시뮬레이션 데이터를 이용하면 데이터 처리 알고리즘에 대해 다양한 종류의 데이터를 적용하여 검증 및 평가하는 것이 가능해진다(Schenk, 2001 ; 이임평, 2003). 또한, 효율적으로 고품질의 데이터를 획득할 수 있도록 라이다의 시스템 변수를 최적화할 수 있게 된다.

2. 센서의 기하모델링

라이다데이터를 모의 생성하기 위해서 센서의 기하모델링을 통해 센서 방정식을 수립해야 한다. 센서 방정식은 라이다를 구성하는 개별센서인 GPS, IMU 및 LS의 정보를 조합하여 대상물의 표면에서 레이저 펄스가 반사된 지점의 3차원 좌표를 계산하기 위한 공식이다. 먼저 시스템 오차가 고려되지 않은 센서방정식을 수립하고, 여기에 시스템 오차 요소를 포함하여 시스템 오차 요소가 포함된 최종적인 센서방정식을 유도한다.

2.1 좌표계의 설정

기하모델링을 통해 센서방정식을 수립하기 위해 먼저 그림 1에 보는 바와 같이 개별센서 및 지상에서 정의된 다양한 좌표계를 설정해야 한다. 먼저 항공기에 탑재된 라이다시스템을 구성하는 세 개의 개별센서의 원점에 위치한 GPS좌표계, IMU좌표계와 LS좌표계를 정의하였다. LS좌표계의 원점은 레이저펄스의 송신원점에 위치하고, X-Y 평면은 레이저 펄스의 스캔면과 Z축은 순간(IMU

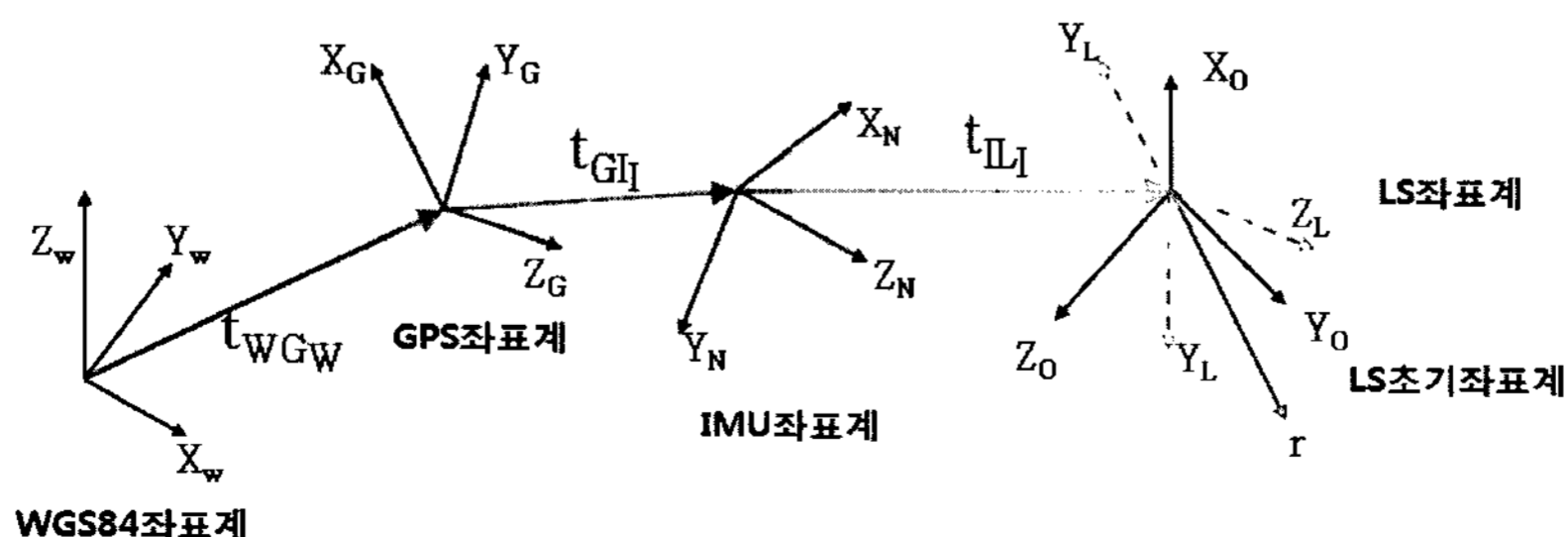


그림 1. 개별센서에 정의된 좌표계

tantaneous) 스캔각이 0도 일 때 레이저펄스의 송신방향에 일치하도록 정의하였다. LS초기좌표계는 LS좌표계와 동일한 원점을 가지며 X-Y평면은 LS좌표계와 일치하고 +Z축은 순간 스캔각에 따라 달라지는 특정한 레이저 펄스가 발사되는 방향으로 설정하였다. 지상의 기준좌표계로 WGS84좌표계를 설정하였다. LS초기좌표계, LS좌표계, IMU좌표계, GPS좌표계, WGS84좌표계를 나타내는 첨자로 각각 0, L, N, G, W를 설정하였다(이임평 등, 2005).

라이다를 구성하는 개별센서의 데이터를 조합하기 위해서는 하나의 좌표계로 표현된 데이터를 또 다른 좌표계로 변환할 수 있어야 한다. 이러한 좌표변환을 위해서 개별 좌표계 사이의 기하학적 관계가 먼저 수립되어야 한다. 정의된 모든 개별 좌표계는 동일한 축척(scale)을 갖는 3차원 좌표계이기 때문에 두 개의 좌표계 사이의 관계는 좌표계 원점의 차이를 나타내는 이동(translation) 벡터와 좌표축의 방향의 차이를 회전(rotation) 행렬로 정의된다. 예를 들어, 그림 1에서 t_{IG} 는 GPS좌표계와 IMU좌표계의 원점의 차이를 나타내는 이동벡터이다. 여기서 t_{IG} 는 IMU좌표계(I)의 원점에서 출발하여 GPS좌표계(G)의 원점에 연결되는 이동벡터(t)를 IMU좌표계(I)로 표현한다는 것을 의미한다. 또한, R_{IG} 는 IMU좌표계(I)를 GPS좌표계(G)로 변환하기 위한 회전행렬을 의미한다. 이와 비슷한 방식으로 개별 좌표계 사이의 관계를 나타내는 이동벡터와 회전행렬에 대한 기호를 표 1에서와 같이 정의한다. 표 1에서 WGS84좌표계의 관계를 나타내는 이동벡터

나 회전행렬은 라이다가 항공기에 탑재되어 데이터를 취득할 때 항공기의 위치나 자세에 따라 항상 달라진다. 이는 탑재된 GPS 및 IMU로 취득된 정보를 통해 결정된다. 나머지 이동벡터나 회전행렬은 항공기에 탑재된 개별센서 좌표계 사이의 관계를 나타내기 때문에 항공기의 위치나 방향에 무관한 일정한 값을 갖는다. 이러한 값들은 항공기에 개별센서를 탑재한 후에 지상에서 정밀측량을 통해 결정한다.

2.2 센서방정식의 수립

LS로 취득된 레이저 펄스의 송신원점에서 반사지점까지의 거리(r)를 이용하여 반사지점의 좌표를 LS초기좌표계로 표현하면 식(1)과 같다. 여기서 P_0 는 LS초기좌표계(0)로 표현된 좌표값 벡터이고, u_z 는 +Z축 방향의 단위벡터를 의미한다.

$$P_0 = [001] \cdot r = u_z r \quad (1)$$

P_0 를 IMU좌표계(I)로 변환하면 식(2)와 같다. 여기서 P_I 는 IMU좌표계(I)로 표현된 반사지점의 좌표값 벡터이다. 또한, R_{0L} 은 LS초기좌표계에서 LS좌표계로 변환하는 회전행렬을 의미하는 데, 이는 레이저 펄스를 다양한 방향으로 송신하기 위해 적용되는 선형, 타원형 등의 스캔 방식에 따라 결정된다.

$$P_I = R_{LI}R_{0L}P_0 + t_{LI} = R_{LI}R_{0L}u_z r + t_{LI} \quad (2)$$

P_I 를 GPS좌표계(G)로 변환하면 식(3)과 같다. 여기서, P_G 는 GPS좌표계(I)로 표현된 반사지점의 좌표값 벡터이고, t_{GI} 은 GPS좌표계와 LS좌표계 사이의 이동벡터를 의미하며 $(t_{LI} + t_{GI})$ 로 정의된다.

$$P_G = R_{IG}(P_I + t_{GI}) = R_{IG}(R_{LI}R_{0L}u_z r + t_{LI} + t_{GI}) = R_{IG}(R_{LI}R_{0L}u_z r + t_{GI}) \quad (3)$$

P_G 를 WGS84좌표계(W)로 변환하면 식(4)와 같다. 식(4)는 개별센서의 정보를 조합해서 반사지점의 3차원 좌표를 계산하기 위한 최종적인 센서방정식이다. 여기서, P_W 는 WGS84좌표계(W)로 표현된 반사지점의 좌표값 벡터이고, R_{IW} 은 IMU좌표계에서 WGS84좌표계로 변환하는 회전행렬을 의미하며 $R_{GW}R_{IG}$ 로 정의된다.

표 1. 개별 좌표계 사이의 관계를 나타내는 이동벡터 및 회전행렬

기호	정의
t_{LI}	IMU좌표계로 표현된 IMU좌표계와 LS좌표계 사이의 이동벡터
t_{GI}	IMU좌표계로 표현된 GPS좌표계와 IMU좌표계 사이의 이동벡터
t_{WG}	WGS84좌표계로 표현된 WGS84좌표계와 GPS좌표계 사이의 이동벡터
R_{0L}	LS초기좌표계에서 LS좌표계로 변환하는 회전행렬
R_{LI}	LS좌표계에서 IMU좌표계로 변환하는 회전행렬
R_{IG}	IMU좌표계에서 GPS좌표계로 변환하는 회전행렬
R_{GW}	GPS좌표계에서 WGS84좌표계로 변환하는 회전행렬

$$P_W = R_{GW}P_G + t_{WG_w} = R_{GW}R_{IG}(R_{LI}R_{0L}u_z r + t_{GL_I}) + t_{WG_w} = R_{IW}(R_{LI}R_{0L}u_z r + t_{GL_I}) + t_{WG_w} \quad (4)$$

식(4)의 센서방정식에서 t_{WG_w} 는 GPS에서 취득한 WGS84좌표계로 표현된 GPS수신기의 위치 즉, GPS 좌표계의 원점의 좌표이고, R_{IW} 는 IMU에서 취득한 IMU좌표계를 WGS84좌표계로 변환하기 위한 회전행렬이고, r 는 LS에서 취득한 반사지점까지의 거리이다. t_{WG_w} , R_{IW} , r 은 개별 레이저 펄스를 송신하는 시점에서의 비행기의 위치나 자세에 따라 항상 달라진다. 나머지 R_{0L} , R_{LI} , t_{GL_I} 는 비행기의 위치와 자세와 관계없이 항상 일정한 값이다.

식(4)의 센서방정식은 개별센서 데이터에 포함된 오차와 개별센서를 통합에 관련된 오차가 고려되지 않았다. 개별센서오차로 GPS 데이터에 대기오차, 시계오차, 다중경로오차 등이, IMU 데이터에 속도와 회전량, 스케일, 중력효과에 대한 오차 등이 포함될 수 있고, LS 데이터는 거리와 방향에 포함된 편이가 존재할 수 있다. 또한, 센서 통합오차로는 플랫폼의 진동과 같은 외력으로 인해 센서 간의 상대적인 위치와 방향이 변화되기 때문에 발생하는 오차와 서로 다른 발생 주기를 갖는 데이터를 조합함에 따라 발생하는 내삽오차 등이 있다. 이러한 개별센서오차 및 센서통합오차에 대한 원인, 종류 및 수학적 모델에 대한 자세한 내용은 이임평 등(2005) 또는 Schenk(2001)을 참고한다.

개별센서와 센서통합에 관련된 오차 중에 주된 요소들을 수학적으로 모델링하여 식(4)의 센서방정식에 추가하면 식(5)와 같이 시스템 오차가 포함된 센서방정식이 유도된다(이임평 등, 2005). 여기에 포함된 오차변수는 표 2

표 2. 오차변수의 정의 및 설명

오차변수	설명
Δr	r 의 편이
ΔR_{0L}	R_{0L} 의 편이
ΔR_{LI}	R_{LI} 의 편이
ΔR_{IG}	R_{IG} 의 편이
Δt_{GL_I}	t_{GL_I} 의 편이
Δt_{WG_w}	t_{WG_w} 의 편이

에 기술하였다.

$$P_W^* = R_{GW}\Delta R_{IG}R_{IG}(\Delta R_{LI}R_{LI}\Delta R_{0L}R_{0L}u_z(r + \Delta r) + t_{GL_I} + \Delta t_{GL_I}) + t_{WG_w} + \Delta t_{WG_w} \quad (5)$$

3. 데이터 시뮬레이션 방법론

3.1 시뮬레이션의 개요

라이다데이터를 시뮬레이션을 통해 생성하려면 입력으로 1)라이다 시스템을 탑재한 비행기의 경로 및 자세에 대한 정보, 2)라이다 시스템으로 측량을 하고자하는 지역의 DEM, 3)라이다의 시스템 변수가 주어져야 한다. 라이다의 시스템 변수는 개별센서의 운용 특성을 나타내는 변수와 라이다를 구성하는 개별센서 사이의 기하학적 관계를 설정하는 표 1에 명시된 이동벡터와 회전행렬, 표 2에 명시된 다양한 종류의 오차변수 등을 의미한다. 위와 같은 3가지 종류의 입력 데이터를 바탕으로 실제 항공 라이다 측량을 통해 취득될 수 있는 데이터를 시뮬레이션을 통해 생성하고자 한다.

레이저 펄스가 반사된 지점의 3차원 좌표에 대한 모의 생성은 식(4)와 식(5)에 기술된 센서방정식에 기반한다. 시스템 오차를 고려하지 않은 센서방정식 식(4)를 식(6)처럼 WGS84좌표계로 나타낸 레이저 펄스의 송신 방향에 대한 단위벡터(u_L)와 송신원점(P_L)이 명시적으로 포함되도록 정리한다. 식(6)을 살펴보면 u_L 과 P_L 은 각각 식(7)과 식(8)처럼 표현된다.

$$P_W = R_{IW}R_{LI}R_{0L}u_z r + R_{IW}t_{GL_I} + t_{WG_w} \equiv u_L r + P_L \quad (6)$$

$$u_L \equiv R_{IW}R_{LI}R_{0L}u_z \quad (7)$$

$$P_L \equiv R_{IW}t_{GL_I} + t_{WG_w} \quad (8)$$

결국 식(6)을 기반으로 하나의 레이저 펄스의 반사지점에 대한 3차원 좌표(P_W)를 계산하기 위해서 식(6)에 포함된 P_L , u_L 및 r 을 결정해야 한다. 먼저 P_L 과 u_L 은 라이다를 탑재한 항공기의 경로 및 자세에 대한 정보를 이용하여 결정한다. 결정된 P_L 을 시점으로 u_L 의 방향을 갖는 직선을 정의하고, 정의된 직선과 DEM이 만나는 교점을 “Ray-Tracing” 알고리즘을 이용하여 탐색한다. 그리고 결정된 교점과 레이저펄스의 송신원점까지의 거리(r)을 계산한다. 위에서 결정된 P_L , u_L 및 r 과 시스템 변수로

주어진 표 2의 다양한 시스템 오차 값을 시스템오차가 고려된 센서방정식인 식(5)에 대입하여 반사지점의 오차가 포함된 3차원 좌표값을 계산한다.

3.2 송신 원점 및 방향의 결정

특정 시각 t 에서 발사되는 레이저 펄스의 송신방향 u_L 은 식(7)에서 표현된 것처럼 R_{TW} , R_{LI} , R_{0L} 및 u_z 의 조합으로 결정된다. 여기서 R_{TW} 와 R_{0L} 만 시간에 따라 달라지는 값이다. R_{TW} 는 각 축에 대한 회전각(ω , ϕ , κ)을 시간에 따른 함수로 명시한 항공기의 자세에 대한 정보로부터 결정된다. 또한, R_{0L} 은 시간에 따라 달라지는 개별 레이저 펄스의 순간 스캔각의 함수로 표현된다. 나머지 R_{LI} 와 u_z 는 고정된 값이다. R_{LI} 는 항공기에 IMU와 LS를 탑재한 후에 정밀하게 측정되는 값으로 시뮬레이션에서는 3차원 단위행렬로 설정해도 된다. u_z 는 +Z축 방향의 단위 벡터를 의미한다.

특정 시각 t 에서 발사되는 레이저 펄스의 송신원점 P_L 을 계산하기 위해서 먼저 레이저 펄스를 송신한 시각에서 라이다를 탑재한 비행기의 위치를 결정해야 한다. 비행기의 경로 및 속도는 시뮬레이션의 입력 데이터로 주어진다. 여기서 비행경로는 등속비행을 하는 직선구간의 합으로 명시한다. 하나의 구간은 시점(P_s), 종점(P_e), 속력(v)로 명시된다. 이 경우 비행기의 속도(V)는 식(9)처럼 표현된다.

$$V = \frac{P_e - P_s}{\|P_e - P_s\|} v \quad (9)$$

만약 비행기과 시점에서 출발한 시각이 t_s 라면, 하나의 레이저 펄스를 송신한 시각 t 에서 비행기의 위치(비행기에 탑재된 GPS에 고정된 GPS좌표계의 원점의 위치)는 식(10)처럼 표현된다.

$$\begin{aligned} t_{WGW} &= P_s + V(t - t_s) \\ &= P_s + \frac{P_e - P_s}{\|P_e - P_s\|} v(t - t_s) \end{aligned} \quad (10)$$

식(8)에서 표현된 것처럼, 특정 시각에서의 비행기의 위치 t_{WGW} 에 GPS좌표계의 원점과 LS좌표계의 원점사이의 이격벡터를 나타내는 $R_{TW}^{t_{GL}}$ 를 더하여 레이저 펄스의 송

신원점 P_L 을 계산한다. t_{GL} 은 GPS좌표계와 LS좌표계 사이의 이동벡터를 지상에서 측정하여 IMU좌표계로 표현한 것이다. 여기에 특정시각에서 IMU좌표계를 WGS84좌표계로 변환하는 회전행렬 R_{TW} 를 곱하여 $R_{TW}^{t_{GL}}$ 는 결국 WGS84좌표계로 표현된 이동벡터로 변환한 것이다. R_{TW} 는 시뮬레이션의 입력데이터로 주어지는 비행기의 자세정보에 따라 달라지는 송신방향을 결정할 때 사용한 특정시각에서 IMU좌표계를 WGS84좌표계로 변환하는 회전행렬이다.

3.3 반사지점의 결정

식(6)에서 표현한 반사지점의 위치는 송신원점에서 반사지점까지의 거리(r)를 변화함에 따라 레이저 펄스의 송신원점(P_L)에서 송신방향(u_L)을 따라 설정되는 직선상에 위치한다. 결국 식(6)의 반사지점의 위치를 나타내는 WGS84좌표계로 표현된 직선의 방정식이다. 레이저 펄스의 송신 원점 및 방향으로 정의된 직선과 시뮬레이션의 입력데이터로 지형지물에 대한 정보로 주어진 DEM과의 교차점을 “Ray-Tracing” 알고리즘을 사용하여 탐색하고 그 위치를 결정한다.

Ray-Tracing 알고리즘은 3차원 공간상에서 정의된 직선을 따라 움직이는 광선(ray)이 가장 첫 번째로 만나는 표면상의 교차점을 결정하는 알고리즘으로 특히 컴퓨터 그래픽스 분야에서 많이 개발되어 활용되고 있다(Hearn and Baker, 2004, pp.597-614). 본 연구에서 사용된 Ray-Tracing 알고리즘은 기존의 알고리즘을 약간 변형하여 사용하였다. 사용된 알고리즘을 이용하여 레이저 펄스와 DEM이 교차하는 지점을 찾아내는 원리를 그림 2에서 보여준다. 먼저 전체 DEM의 최대 및 최소 고도값을 이용하여 교차점의 수직방향의 범위(V1)를 설정한다. 최대 및 최소 고도값을 갖는 가상의 수평면을 가정하고, 각각의 평면이 레이저 펄스가 비행하는 직선과 만나는 위치를 결정한다. 결정된 위치의 수평범위는 결국 교차점의 수평범위(H1)가 된다. 수평범위에 포함된 DEM의 고도값의 범위를 이용해서 교차점의 수직방향의 범위(V2)를 재설정한다. 재설정된 수직방향의 범위로부터 가정된 두 개의 가상 수평면과 직선이 만나는 위치로 수평범위(H2)를 재설정한다. 이와 같이 교차점에 대한 수직방향의 범위 및 수평방향의 범위를 반복적으로 재설정하여 축소시킨다. 축소되는 변화량이 주어진 임계값 이하로 수렴되면, 이를 중단하고 최종 설정된 수평범위 내에서 주어진 직선과 가

장 첫 번째 만나는 교차점을 계산한다. 이 점은 결국 레이저 펄스가 지표면에 반사되는 지점 P_w 를 의미한다. 또한, 이 때 레이저 펄스의 송신원점(P_L)과 반사지점(P_w)과의 거리를 계산해서 식(6)에 포함된 r 을 결정한다.

여기까지 계산된 반사지점은 시스템의 오차를 전혀 포함하지 않은 참값을 의미한다. 여기에 시뮬레이션의 입력 데이터로 주어지는 표 2에서 명시된 다양한 시스템 오차 변수를 고려하여 식(5)를 이용하여 반사지점의 좌표 P_w^* 를 다시 계산한다.

4. 실험 및 결과 분석

본 연구에서 제안하는 라이다데이터를 모의 생성하는 방법을 적용하여, 주어진 기준(Reference) DEM을 이용하여 모의 라이다데이터를 생성하고자 한다. 또한, 동일 지역에 대해 각각 다른 비행경로로 설정되어 생성된 모의 라이다데이터를 비교해봄으로써, 라이다시스템 변수 중에 비행기의 이동경로(trajectory)가 실제 라이다데이터에 어떤 영향을 미치는지에 대해 알아보하고자 한다.

4.1 입력데이터의 설정

모의 라이다데이터를 생성하기 위해서 LS의 시스템 변수, 비행경로 및 자세, 시스템오차 등을 설정해야 한다. 본 연구에서는 생성될 모의 데이터의 점 간격을 기준으로 시스템 변수를 설정하였다. 또한, 라이다는 선형, 타원형 등 다양한 스캔 방식으로 구현되어 있으나, 본 실험에서는 가장 일반적으로 많이 활용되는 선형방식을 가정하였다. 비행방향을 기준으로 종(along)방향과 횡(across)방향에 대한 점 간격이 대략 1m가 되도록 설계하였다. 종방향의 점 간격(d_{along})은 식(11)과 이 플랫폼의 속도(v) 및 초당 스캔 회수(스캔레이트, r_{scan})를 조합하면 계산할 수 있고, 횡방향의 점 간격(d_{across})은 식(12)와 같이 초당 펄스 송신 회수(펄스레이트, r_{pulse}), 초당 스캔 회수, 비행고도(H), 스캔앵글(θ_{scan})을 이용하여 계산이 가능하다.

$$d_{along} = \frac{v}{r_{scan}} \quad (11)$$

$$d_{across} = \frac{\theta_{scan}}{r_{pulse}/r_{scan}} \times H \quad (12)$$

실험에서 사용된 LS의 시스템의 변수는 표 3과 같다.

하나의 스캔 라인(scan line)당 약 286개의 펄스(pulse)가 발사된다. 스캔앵글(scan angle)은 스캔(scan)의 범위를 의미하여 이 경우에는 30° 이므로 연직선을 기준으로 $-15^\circ \sim 15^\circ$ 의 범위를 스캔하게 된다.

비행기의 비행경로는 시점과 종점을 설정함으로써 결정한다. 본 연구에서는 그림 3과 같이 총 네 번의 비행촬영을 가정하여, 네 개의 스트립(strip)에 대해 비행속도는 65.66 m/s, 비행고도는 500m이며 시점과 종점의 좌표는 표 4와 같이 설정하였다. 각 스트립간의 중첩은 약 20% 정도로 설정하였다. 시뮬레이션에서 고려된 개별센서의 시스템 오차는 표 5와 같다. 시스템 오차는 표 2에서 명시된 것 중에서 오차의 크기가 상대적으로 큰 것을 선택하였으며, 선택된 변수에 대해서 실제 라이다 측량에서

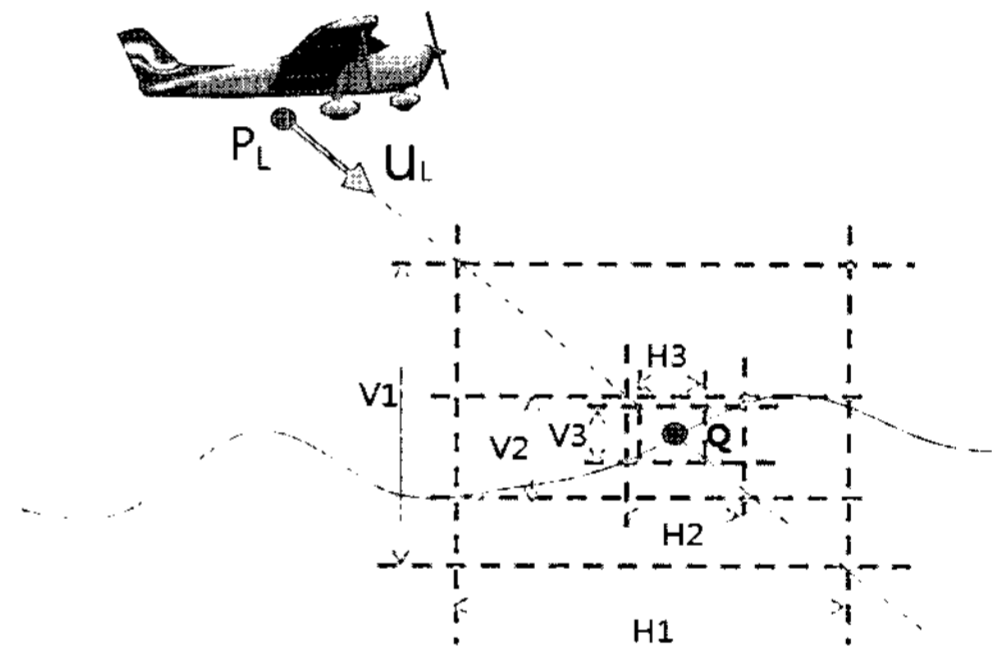


그림 2. Ray-tracing 알고리즘을 이용한 탐색방법

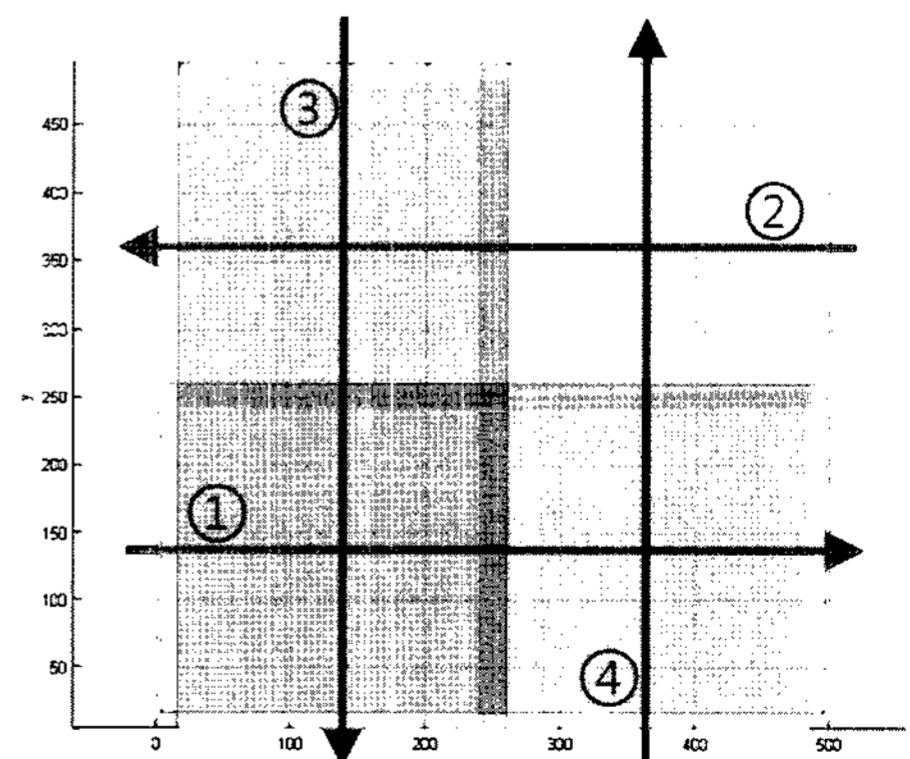


그림 3. 비행경로 1, 2, 3, 4의 모습

표 3. LS 시스템 변수의 설정

변수명	값
Pulse rate[kHz]	20
Scan rate[Hz]	70
Scan angle[deg]	30

발생할 수 있는 정도의 크기로 합리적으로 설정하였다. GPS의 Bias는 1-2m 정도로, IMU Bias는 0.1-0.2° 정도로 가정하였다. 이상과 같이 시스템 변수를 설정한 결과, 종방향과 횡방향의 점 간격이 이론적으로 약 0.9m로 계산되었다.

실험대상지역은 경기도 고양시 일산구의 일부 지역으로 선정하였다. 그림 4는 일산지역의 DEM을 나타낸다. 본 지역은 도심지역으로 다양한 형태 및 높이의 건물들이 포함되어 있고, 비행기의 이동경로에 따라 폐색지역이 나타날 가능성이 높은 지역이다. 본 연구에서는 제안한 시뮬레이션 방법이 다양한 지형지물에 대해서 적용이 가능한지 알아보려고 한다. DEM 격자간격은 1m, 면적은 0.5km x 0.5km 영역이다. 이는 위에서 설정한 비행경로를 충분히 포함한다.

4.2 시뮬레이션 적용결과

시뮬레이션을 수행한 결과로 약 7.4812초 동안 149,243(경로1), 149,246(경로2), 149,152(경로3), 149,325(경로4) 개의 점이 획득되었다. 그림 5는 생성된 가상데이터의 지상커버리지(ground coverage) 및 비행경로를 시뮬레이션

표 4. 비행경로(Trajectory) 설정(단위: m)

	시점의 좌표	종점의 좌표
경로1	(4, 138, 500)	(496, 138, 500)
경로2	(496, 362, 500)	(4, 362, 500)
경로3	(138, 496, 500)	(138, 4, 500)
경로4	(362, 4, 500)	(362, 496, 500)

표 5. 시스템 오차의 설정

오차	분류	변수	값
GPS Bias, x [m]	Δt_{WG_w}	ΔX	2
GPS Bias, y [m]		ΔY	1
GPS Bias, z [m]		ΔZ	0
IMU Bias, Omega [deg]	ΔR_{IG}	$\Delta \omega$	0.1
IMU Bias, Phi [deg]		$\Delta \phi$	0.2
IMU Bias, Kappa [deg]		$\Delta \kappa$	0
Range Bias [m]	Δr	Δr	0.0

을 위해 주어진 기준 DEM과 함께 나타낸다. 중앙에 있는 네 개의 굵은 실선은 비행경로를 의미하고, 이를 둘러싼 가는 실선은 지상커버리지의 경계를 나타낸다. 모의 생성된 점 데이터의 밀도는 1.21 점/m², 평균점간격은 1.05m로 나타났다.

그림 6, 그림 7, 그림 8, 그림 9는 각 비행경로에 따라 생성된 모의 점 데이터(녹색)의 모습이며 우측의 그래프는 생성된 모의 점 데이터의 수평위치 좌표를 기준DEM과의 내삽을 통하여, 그 높이 차이를 히스토그램으로 나타낸 것이다. 각각의 그림에서 왼쪽 히스토그램은 시스템 오차를 고려하지 않은 경우이고, 오른쪽 히스토그램은 시스템 오차를 고려한 경우이다. 시스템 오차를 전혀 고려하지 않은 경우도 DEM의 내삽에 따르는 오차가 존재한다.

최종적으로 생성된 라이다 모의 데이터의 모습은 그림 10에서 보여준다. 왼쪽은 비행경로 1과 2를 통해 획득된 Set 1이고, 오른쪽은 비행경로 3과 4를 통해 획득된 Set

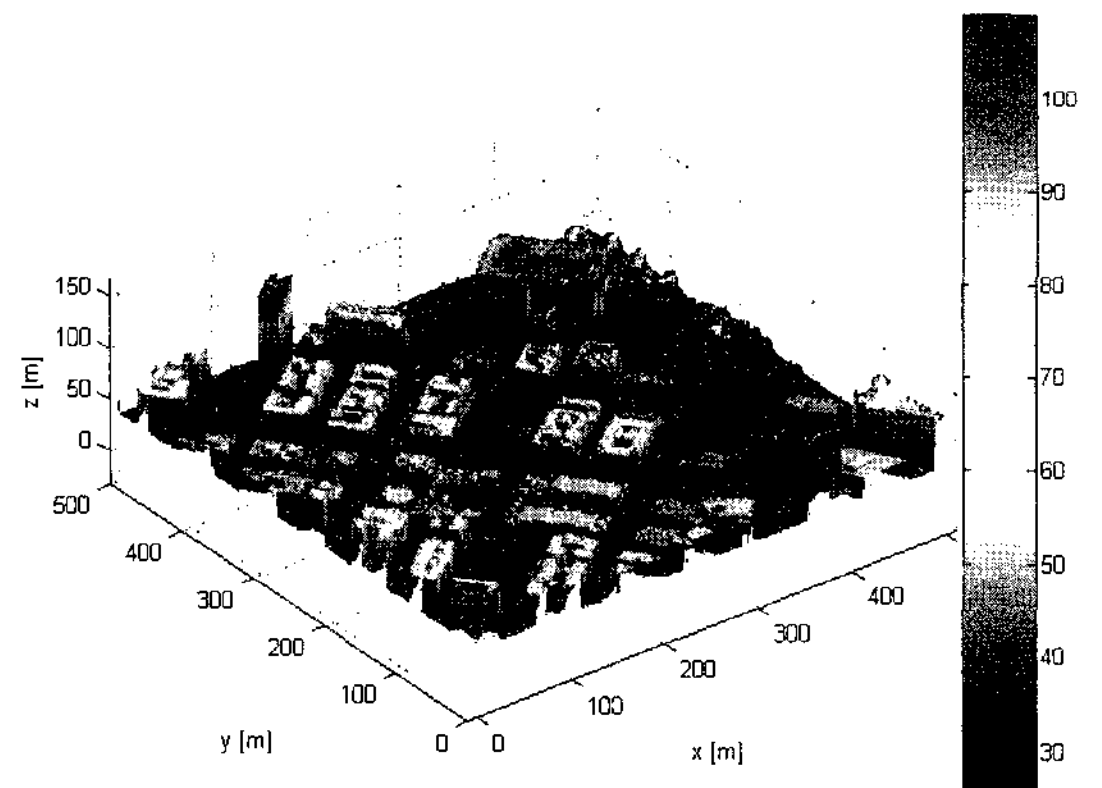


그림 4. 시뮬레이션에 사용한 기준 DEM

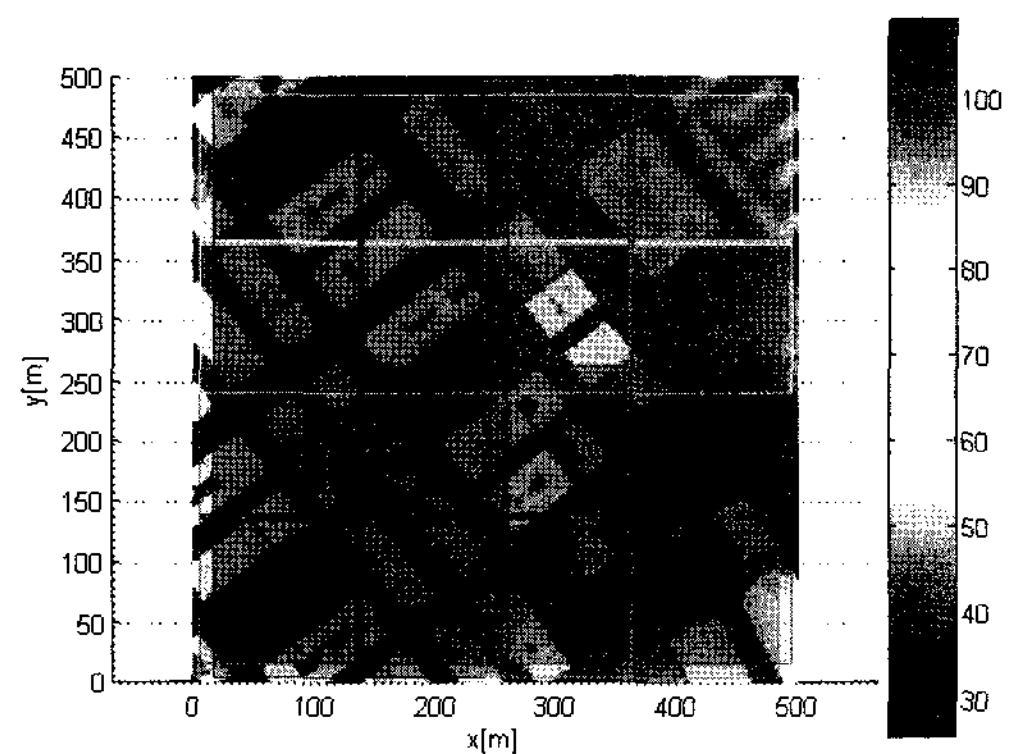


그림 5. 기준 DEM 상에 표현된 가상데이터의 지상커버리지 및 비행경로

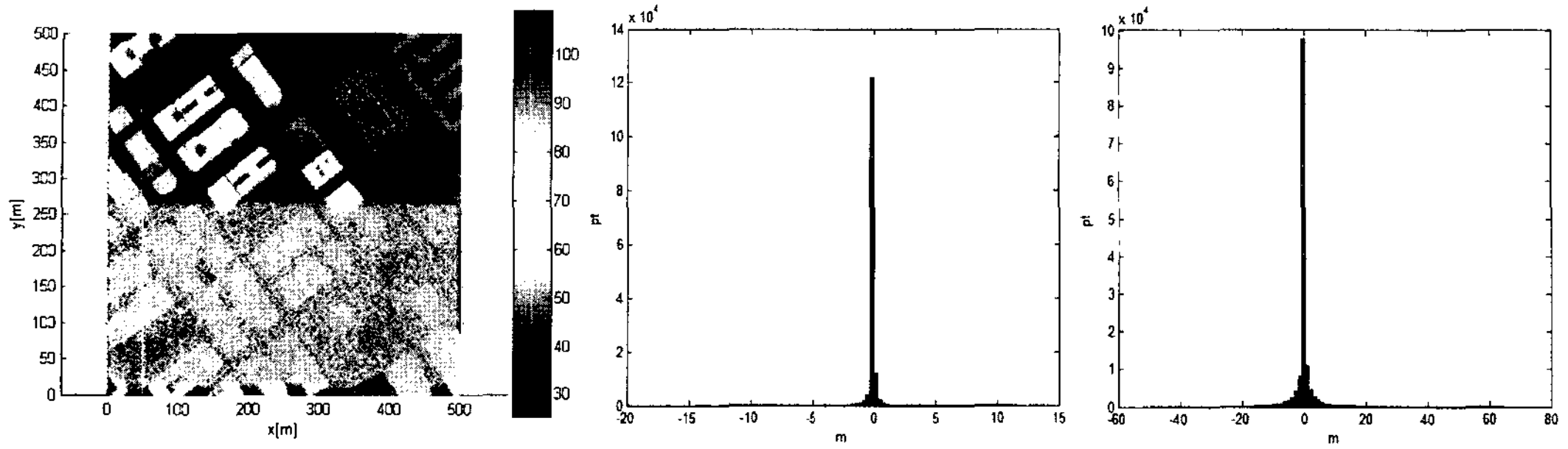


그림 6. 시뮬레이션 결과(경로1)와 고도 차이 히스토그램(좌:오차X, 우:오차O)

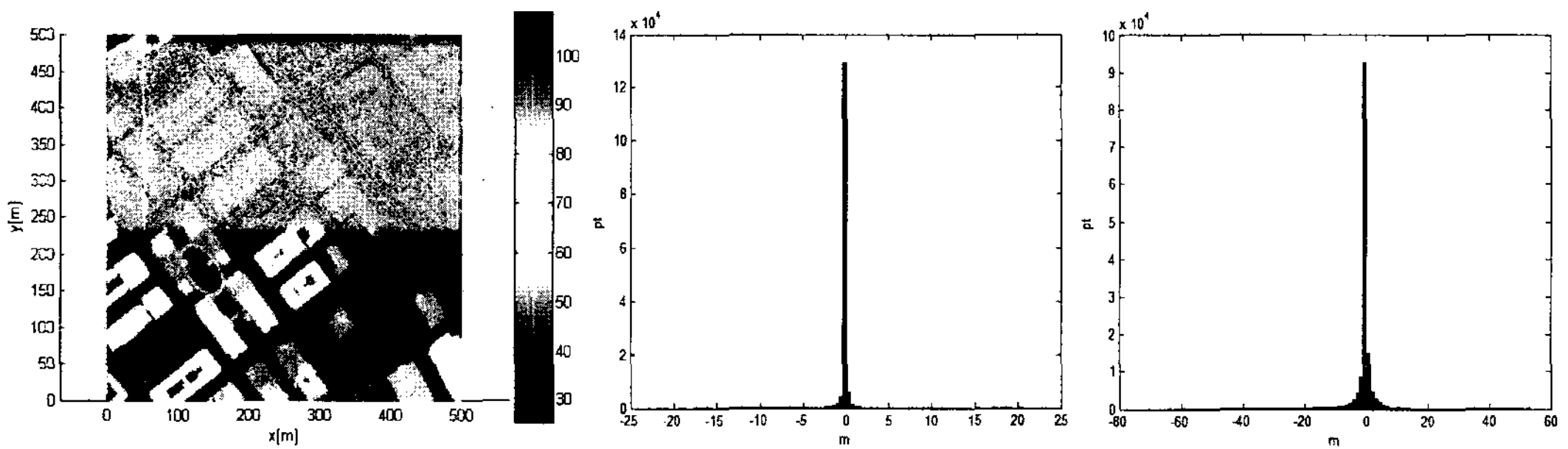


그림 7. 시뮬레이션 결과(경로2)와 고도 차이 히스토그램(좌:오차X, 우:오차O)

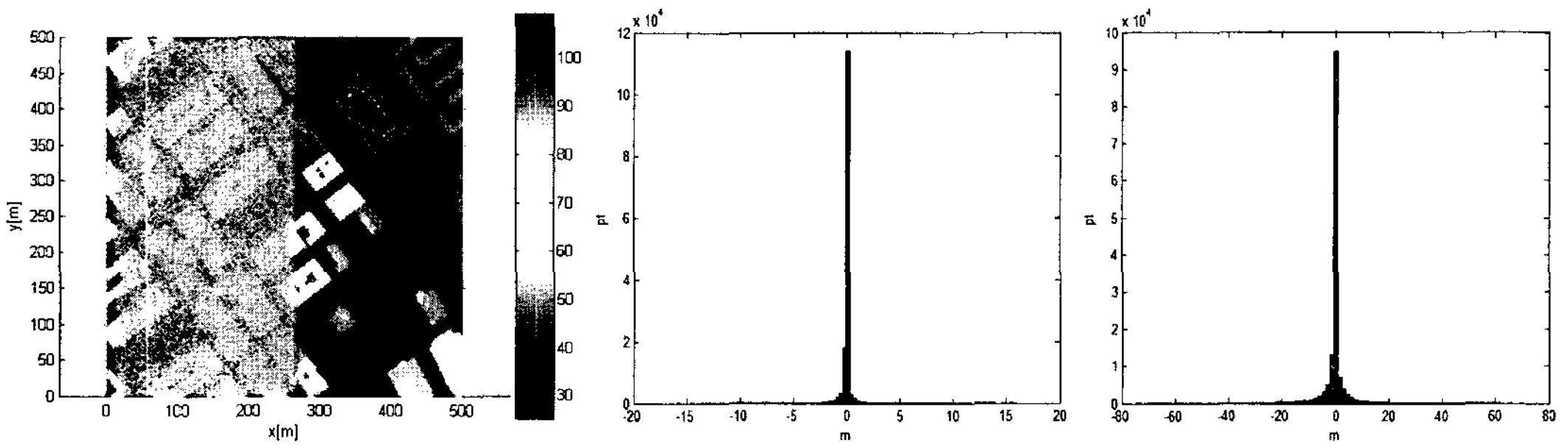


그림 8. 시뮬레이션 결과(경로3)와 고도 차이 히스토그램(좌:오차X, 우:오차O)

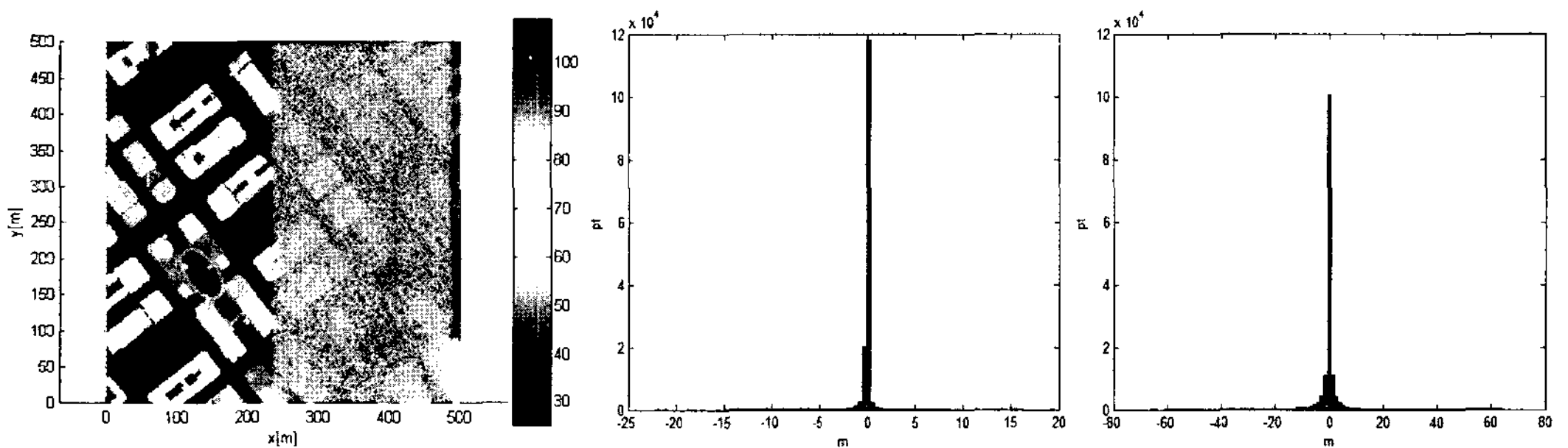


그림 9. 시뮬레이션 결과(경로4)와 고도 차이 히스토그램(좌:오차X, 우:오차O)

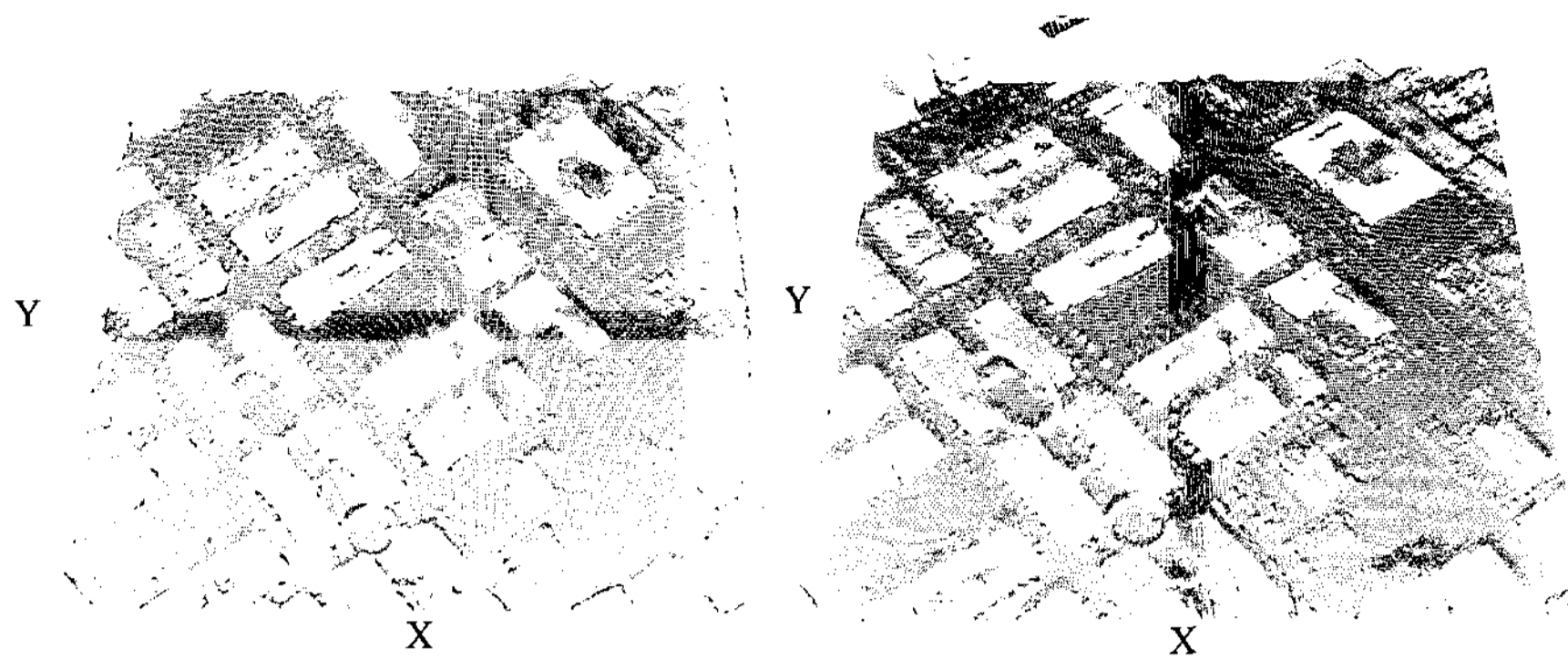


그림 10. 생성된 모의 데이터 - 좌: Set 1(경로1~2), 우: Set 2(경로3~4)

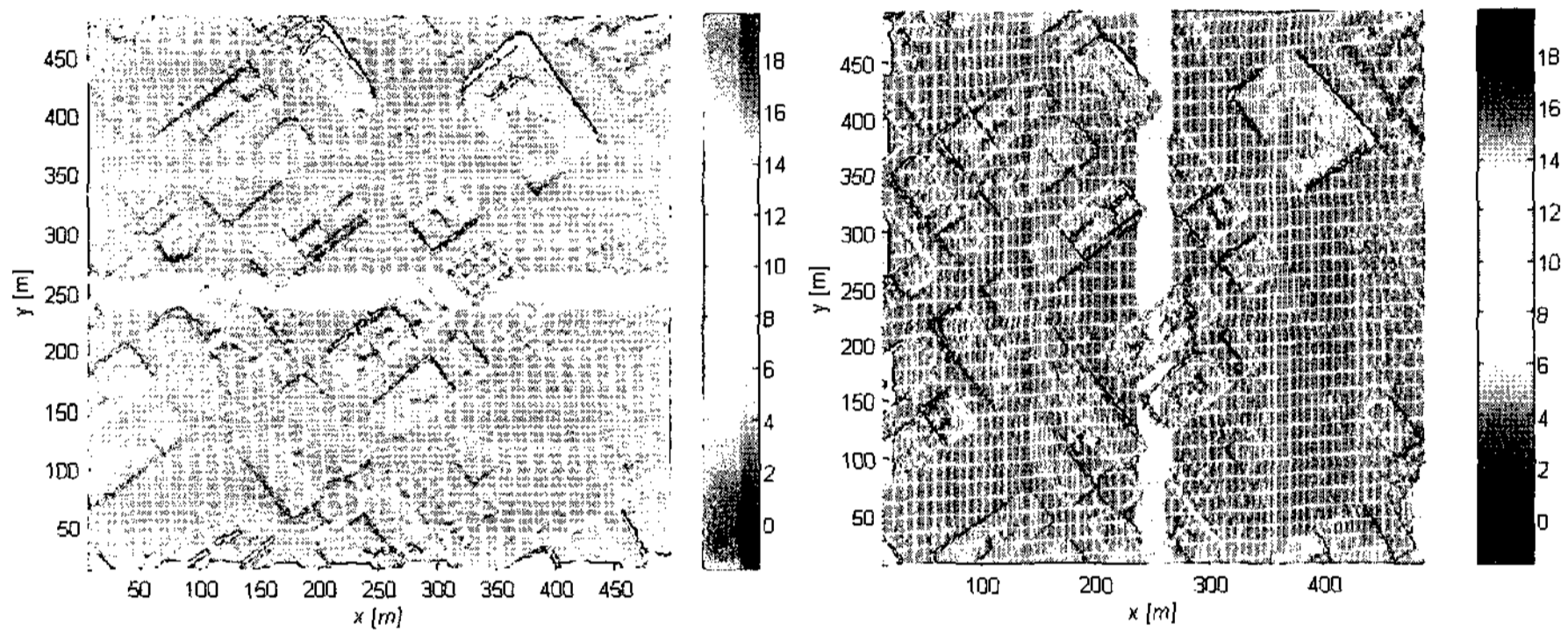


그림 11. Set 1과 Set 2의 Local 점밀도 분포도(좌:Set 1, 우:Set 2)

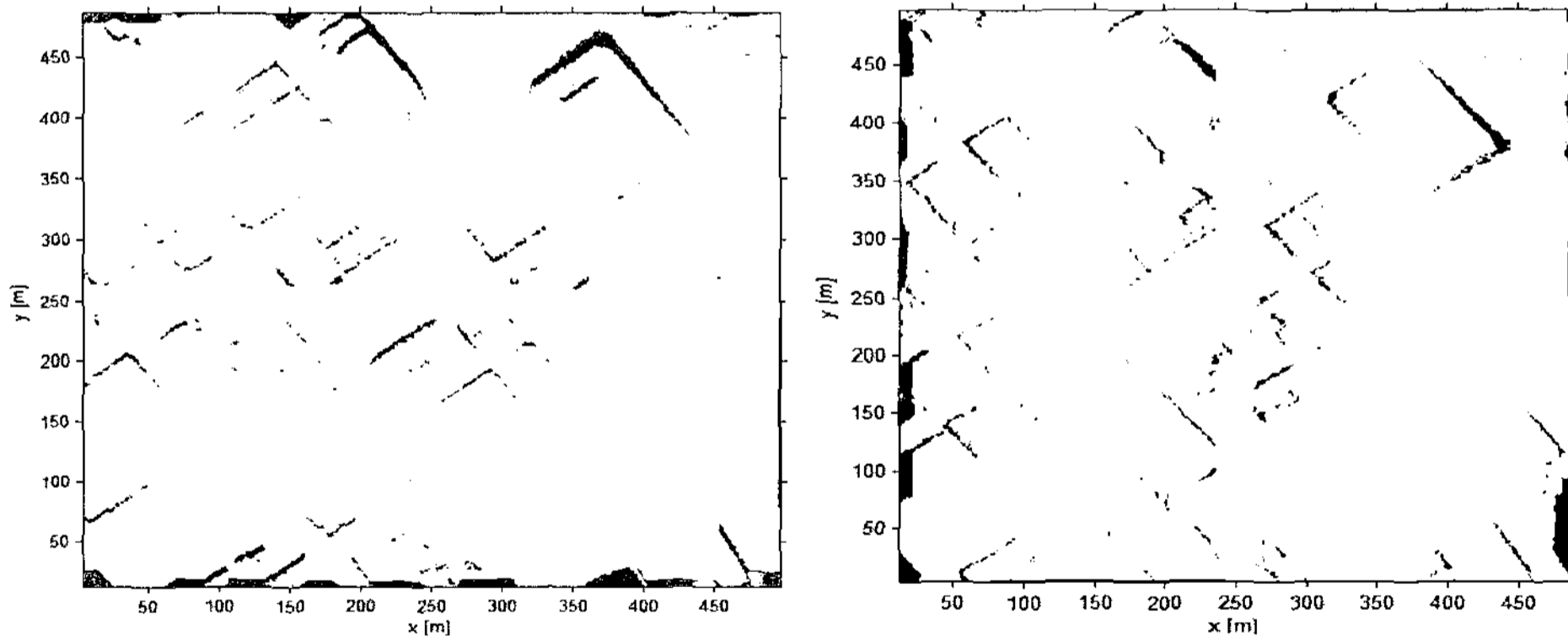


그림 12. Set 1과 Set 2의 점밀도가 0인 빈공간(gap)의 분포도(좌:Set 1, 우:Set 2)

2이다.

그림 11은 1m 간격의 격자(grid)위에 각 셀(cell) 내부에 위치하는 점의 개수로 표현되는 각 셀에 대한 국부(Local) 점밀도를 나타낸다. 각 Set에서 국부 점밀도가 0인 영역만을 그림 12에서 보여준다. 점밀도가 0인 셀들을 세어 본 결그림 7. 시뮬레이션 결과(경로2)와 고도 차이

히스토그램(좌:오차X, 우:오차O) 과, Set 1에서는 6,849개, Set 2에서는 6,356개로 나타났다. Set 1과 Set 2의 전체 영역(Set 1: 493x475, Set 2: 475x493)에서 각각 2.9%와 2.7% 영역에서 데이터 공백(gap)이 발생하였음을 알 수 있다.

5. 결 론

본 연구에서는 다양한 활용 알고리즘의 성능평가를 위해 모의 라이다데이터를 시뮬레이션을 통해 생성하였다. 이를 위해 센서의 기하모델링을 통해 레이저 펄스의 3차원 좌표를 계산하는 센서 방정식을 유도하였다. 센서방정식은 먼저 시스템오차가 전혀 고려되지 않고 유도한 방정식에 개별센서에 수반된 오차와 센서들의 통합에 따라 발생하는 오차를 추가하여 수립하였다. 수립된 센서방정식을 기반으로 주어진 LS 시스템변수, 비행경로 및 자세, 시스템오차, 기준 DEM 등을 입력으로 시뮬레이션을 통해 모의 라이다데이터를 생성하는 방법을 제시하였다. 제시된 방법을 일산지역의 기준 DEM에 적용하여 모의 라이다데이터를 생성하였다. 생성된 데이터를 기준DEM과 비교하여 정확도를 확인하였고, 또한, 국부 점밀도 계산하여 데이터의 Gap 분포를 분석하였다.

추후 연구에서는 시뮬레이션을 통해 생성된 모의데이터를 이용해 분할(Segmentation), 빌딩모델링, 분류(Classification) 등의 다양한 알고리즘을 구체적으로 적용하여 그 성능을 평가하고자 한다. 본 연구는 라이다센서에 관한 알고리즘 연구를 위해 다양한 표적 및 지표에서 취득되는 데이터를 모의 생성하여 제공해 줄 수 있으며, 나아가 최적의 비행 및 시스템 변수를 설정하는 연구에 도움이 될 것이다.

감사의 글

본 연구는 한국과학기술원 영상정보특화연구센터의 II-21 “레이저 영상 신호특성모델링 기법 연구”과제를 통한 방위사업청과 국방과학연구소의 연구비 지원으로 수행되었습니다.(계약번호 UD070007AD)

참고문헌

- 이임평, (2003). “고해상도 DEM을 이용한 시뮬레이션에 기반하는 라이다데이터의 검증 및 보정에 관한 연구”, 대한토목학회 2003년도 정기 학술대회 논문집, pp. 4436~4440.
- 이임평 외, (2005). “다차원공간정보구축에 관한 연구”, 국토지리정보원.
- Allen, J. (2005). Lockheed Martin Successfully Flight Tests Low-Cost Autonomous Attack System, Lockheed Martin, Orlando, <http://www.lockheedmartin.com/products/LOCAAS/index.html>
- Hearn, D. and Baker M. P. (2004). Computer Graphics with OpenGL, 3rd Ed., Pearson Prentice Hall, Upper Saddle River, NJ, USA.
- Lockheed Martin (2006). Low Cost Autonomous Attack System, International Online Defense Magazine, the United Kingdom, <http://www.defense-update.com/products/l/locaas.htm>
- Schenk, T. (2001). “Modeling and analyzing systematic errors of airborne laser scanners”, Technical Notes in Photogrammetry No. 19, The Ohio State University, Columbus, OH, USA.

(접수일 2008. 6. 9, 심사일 2008. 6. 23, 심사완료일 2008. 6. 23)