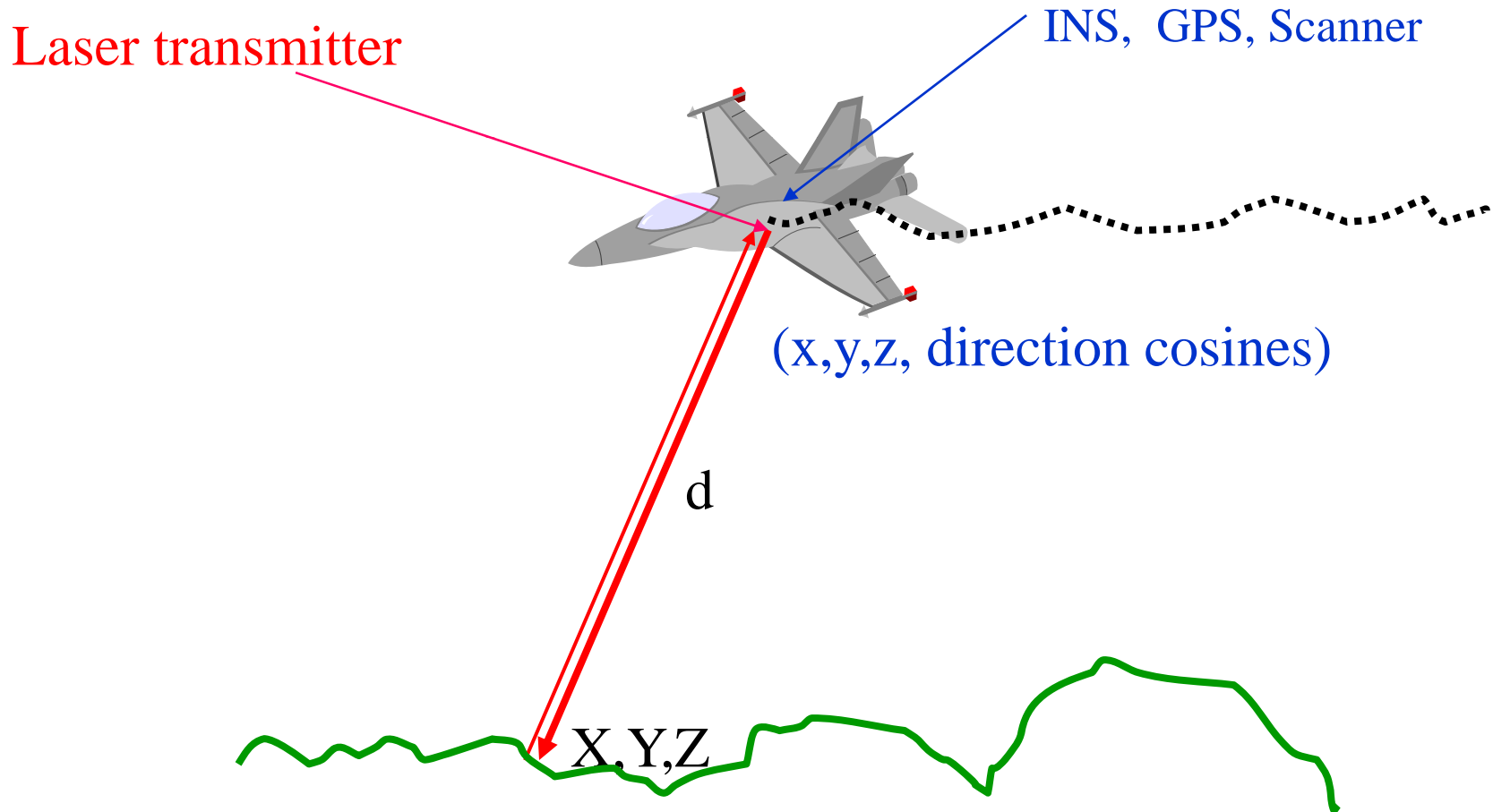


# Object-oriented software engineering for designing an aerial survey LiDAR Simulator

Rakesh Kumar Mishra, Dr. Bharat Lohani  
Geoinformatics division  
Indian Institute of Technology Kanpur  
Kanpur, INDIA

# Principal of LiDAR



---

# LiDAR Technology

- ❑ Provides accurate topographic data at high speed
- ❑ Data collection with higher density, accuracy & less time
- ❑ Weather and light independent

## **Applications:**

- DEM generation
- Flood hazard zonation
- Cellular networks etc.

# Object-oriented software engineering for software development

---

## Initial investigation (Why simulator ?)

- ❑ LiDAR data is not available in most of the countries
- ❑ LiDAR data is not available for teaching as required
- ❑ LiDAR data is not available for research as required
- ❑ Software for flight planning

# Requirements

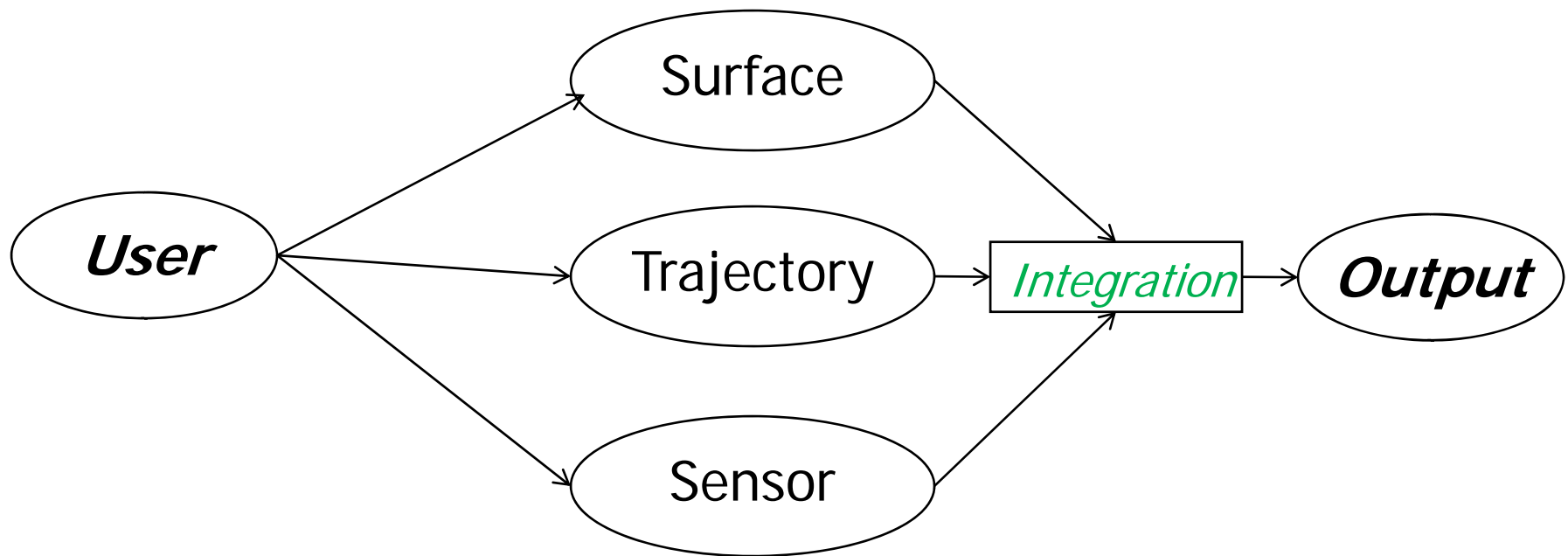
- User friendly GUI
- Simulation of generic as well as commercial sensors
- Simulation of earth like surfaces
- Flight trajectory as in case of actual flight
- Possibilities of error introduction
- Output data in common format
- Help and tutorial

---

## Feasibility study

- ❑ Sufficient background is available for the system development
- ❑ The system can be engineered using current technology
- ❑ Development can be done within the budget & time
- ❑ Developed system will be useful for the user group

# Object-oriented analysis

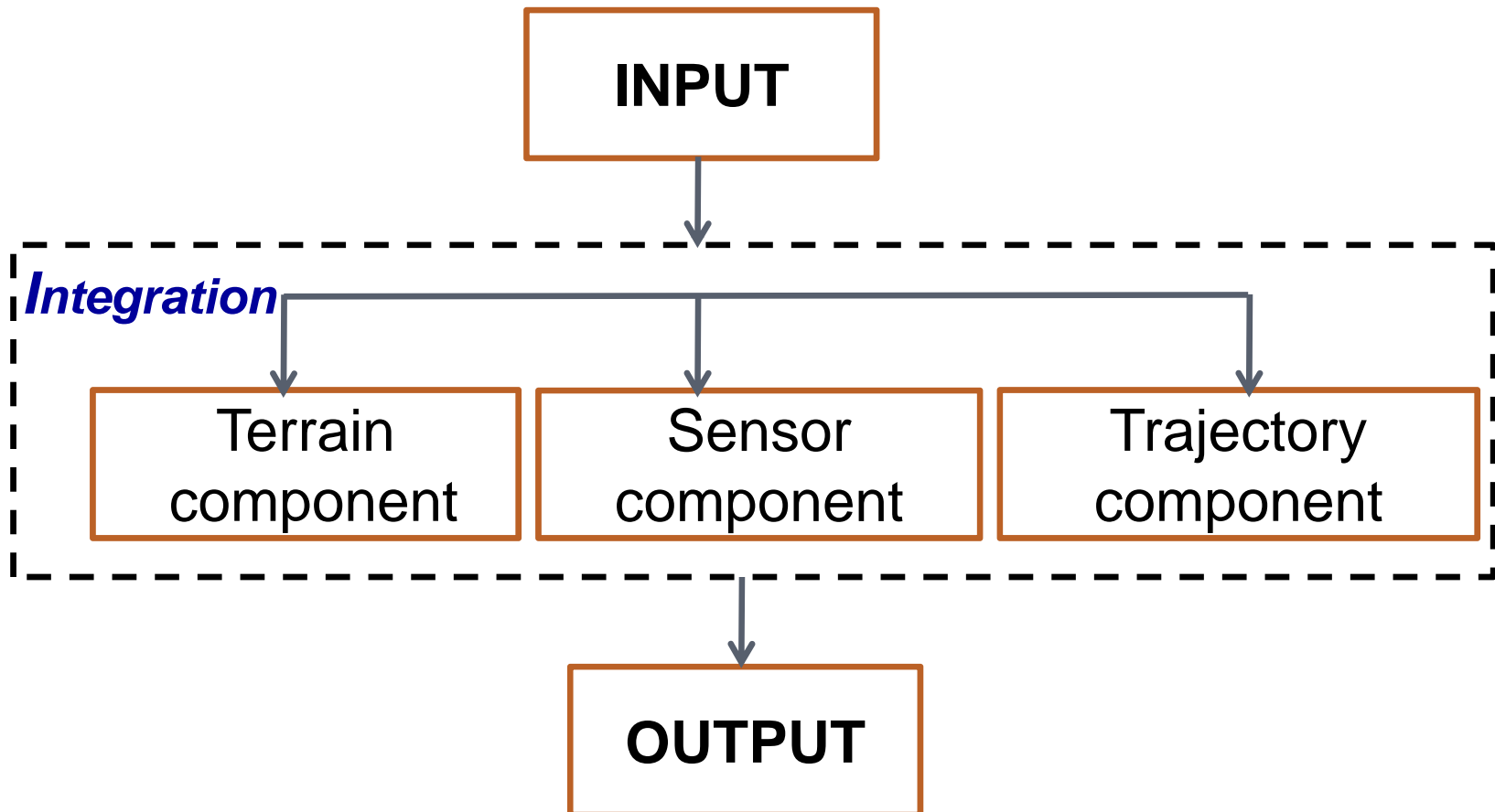


Use case diagram

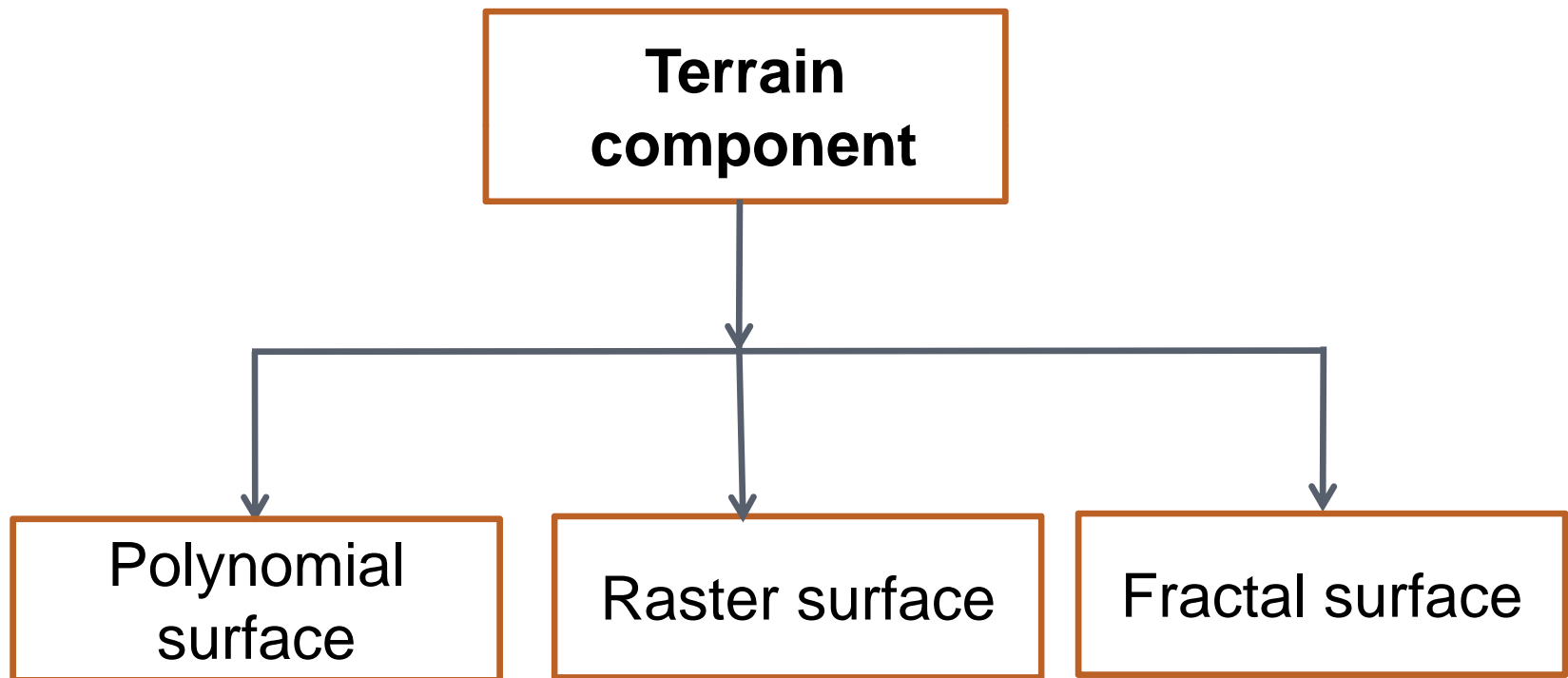


- 
- ❑ Objects in the problem domain is identified
  - ❑ Object relationships are made
  - ❑ Object state table is developed
  - ❑ Inheritance diagram for objects is made

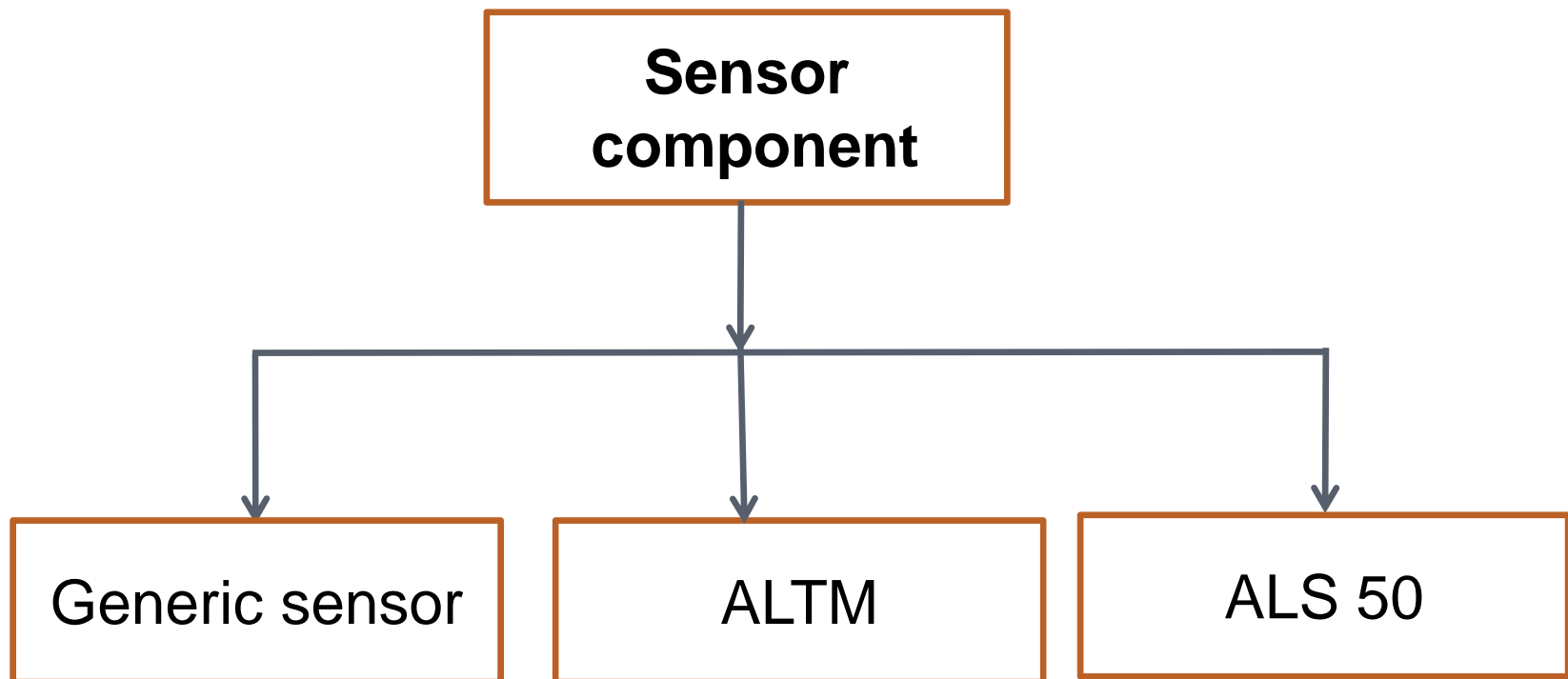
# Object-oriented design



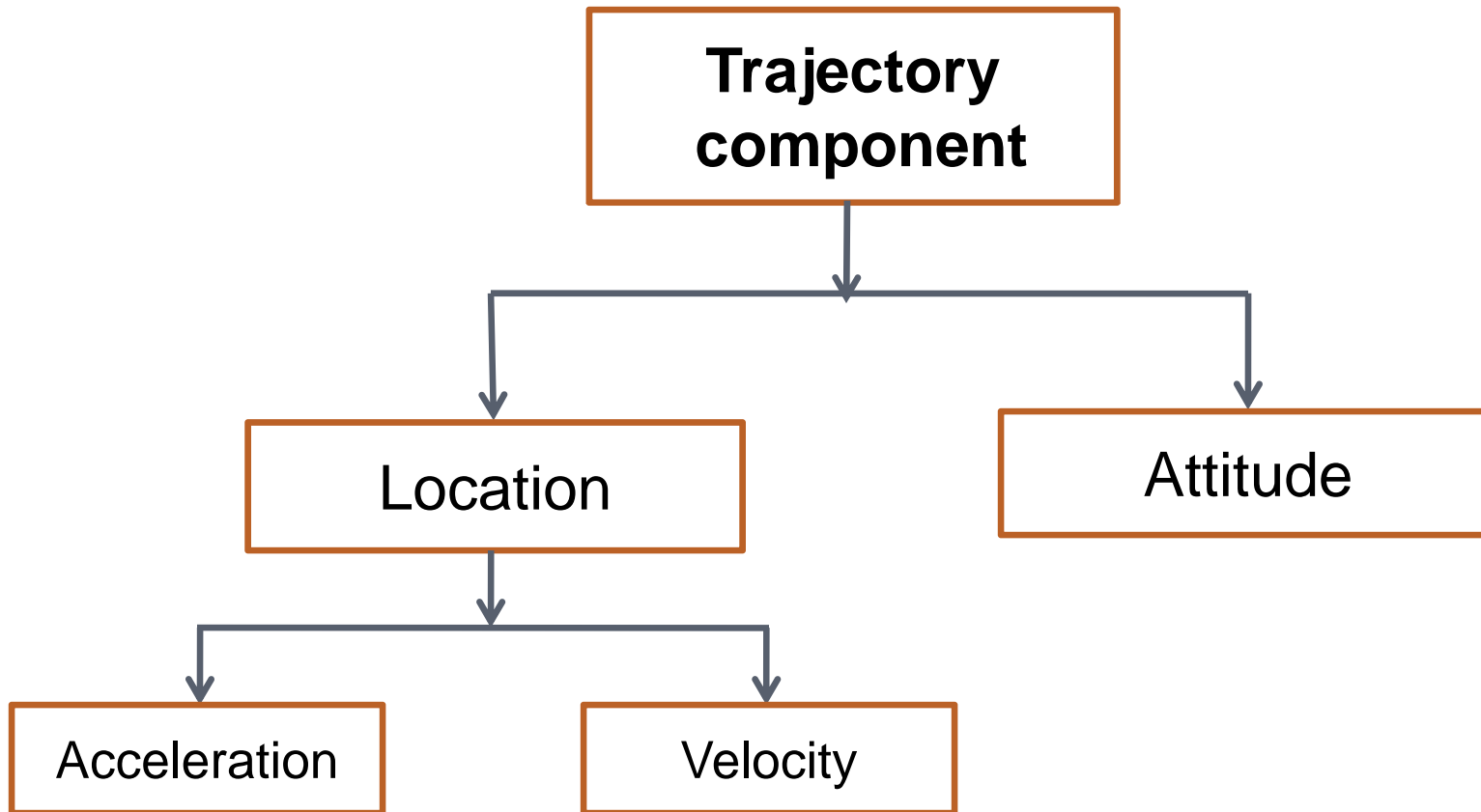
# Terrain components



# Sensor components



# Trajectory components



---

# Class design

- ❑ Identify classes from the components
- ❑ Identify subclass within each class
- ❑ Identify abstraction in each class
- ❑ Identify the common behaviour of classes

---

# System implementation

- ❑ Java is used to realize the design
- ❑ Each classes are implemented with its relationship
- ❑ Standard way of coding is used
- ❑ Methods are designed for each basic task

---

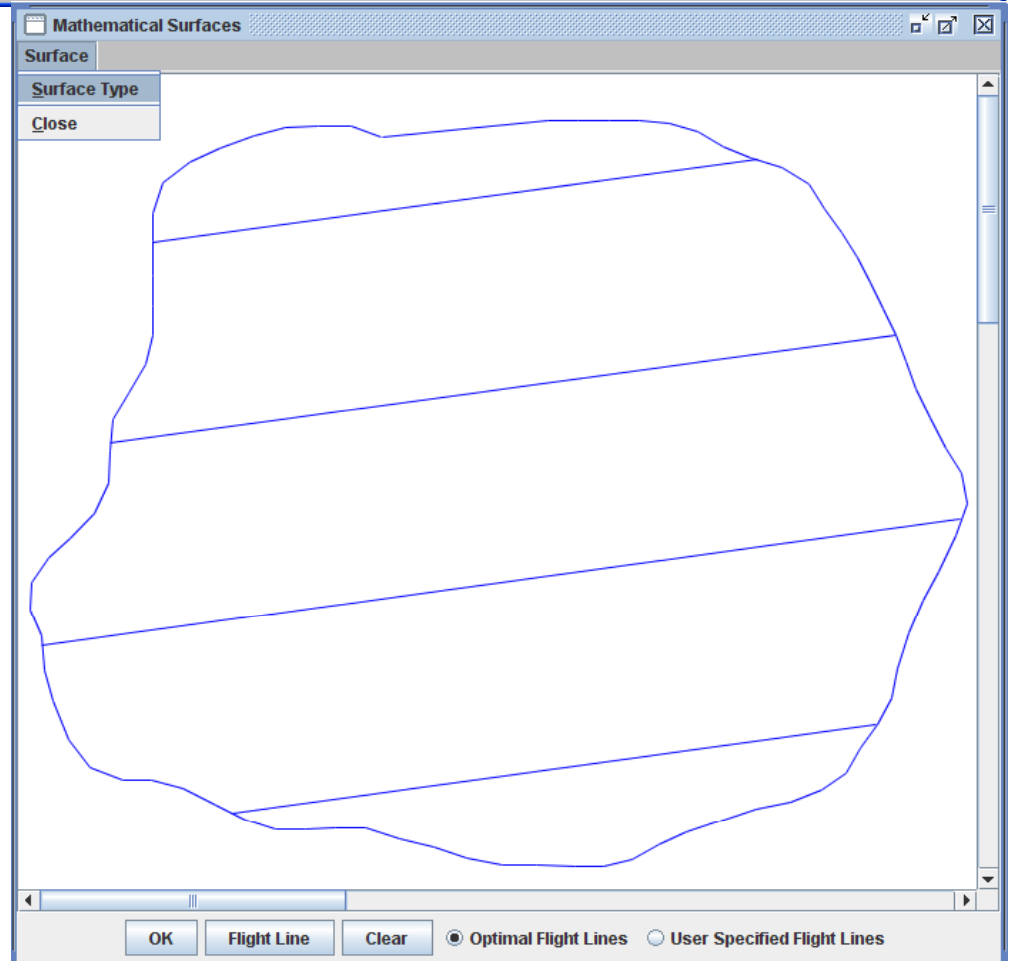
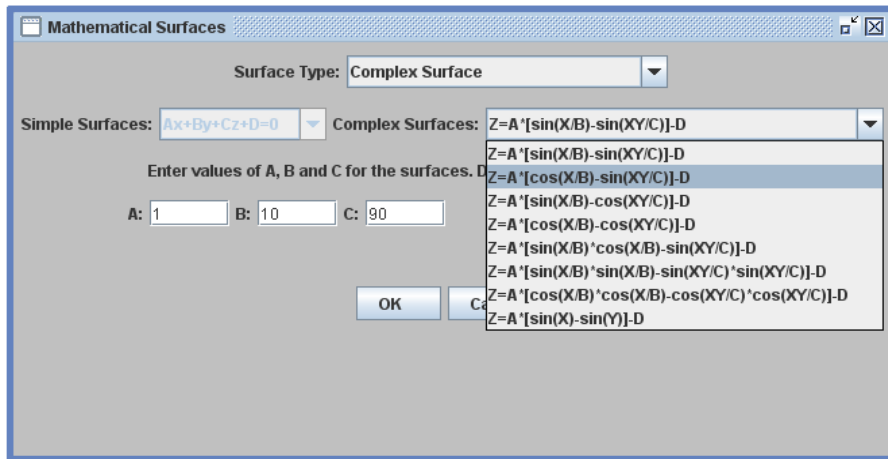
# Complexities handled

- ❑ Efficient algorithms are designed
- ❑ Threads are used to optimize software execution
- ❑ Special data structures are designed to handle memory problems
- ❑ New file formats are designed to improve I/O.



# GUI Screenshots

# Polynomial surface



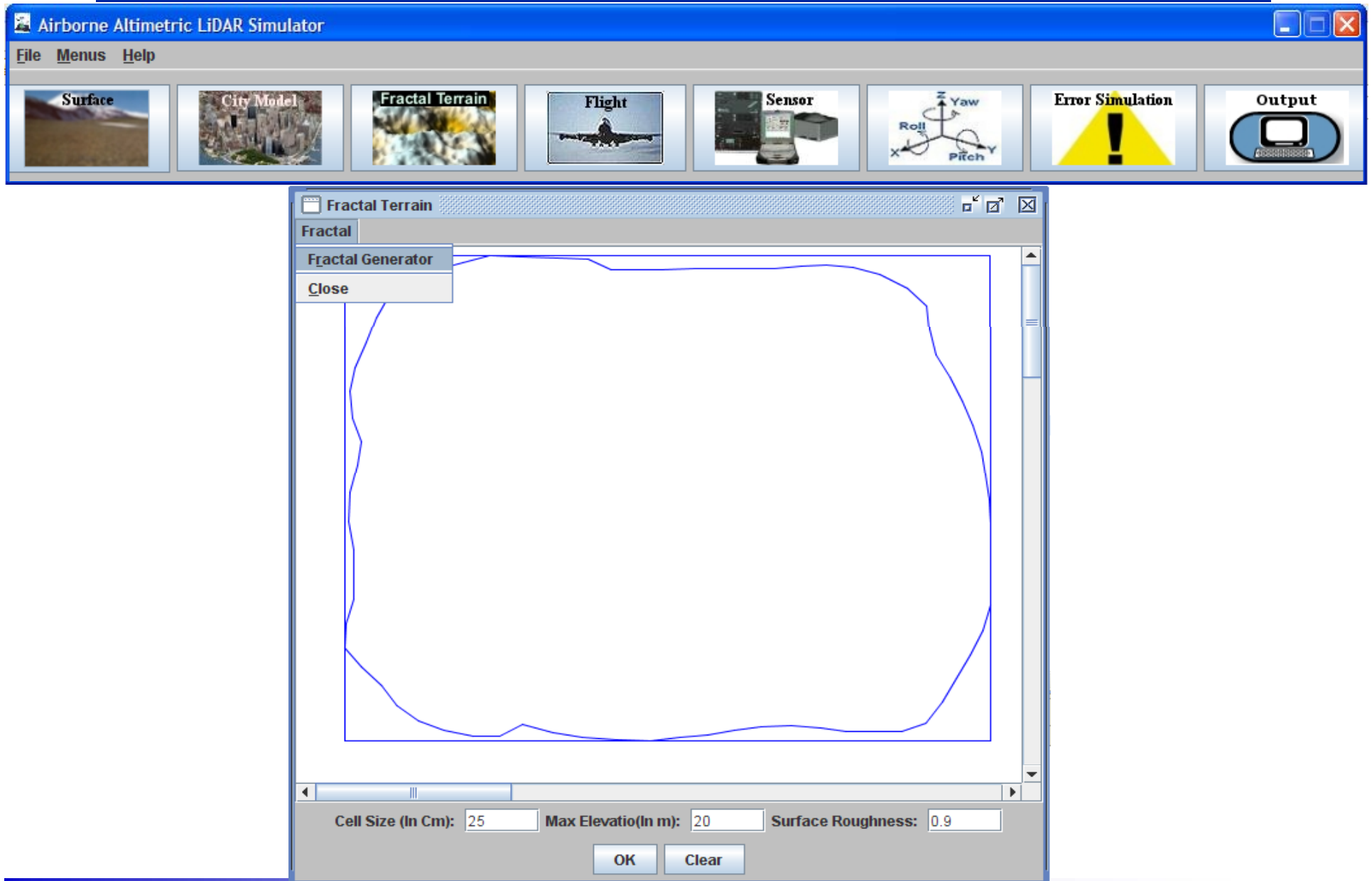
# Raster surface(City model)

The screenshot displays the 'Airborne Altimetric LiDAR Simulator' software interface. At the top, there is a menu bar with 'File', 'Menus', and 'Help'. Below the menu bar is a toolbar with icons for 'Surface', 'City Model', 'Fractal Terrain', 'Flight', 'Sensor', a 3D coordinate system (Roll, Pitch, Yaw), 'Error Simulation', and 'Output'. The main workspace shows a 2D floor plan of a city with various building footprints. A blue outline indicates the current drawing area. A toolbar above the workspace contains icons for different building shapes: a rectangular prism, a cylinder, a gabled roof, a conical roof, and a dome. A status bar at the bottom shows 'Cell Size (In Cm): 25' and buttons for 'OK', 'Draw Area', 'Rotate', 'Delete', and 'Clear'. There are also radio buttons for 'Optimal Flight Lines' and 'User Specified Flight Lines'.

Several dialog boxes are open over the workspace:

- Mathematical Surfaces For Raster:** This dialog allows selecting a 'Surface Type' (Complex Surface) and defining 'Simple Surfaces' (e.g.,  $Ax+By+Cz+D=0$ ) or 'Complex Surfaces' (e.g.,  $Z=A*\sin(Y/C)$ ,  $Z=A*\cos(Y/C)$ ,  $Z=A/2*\sin(X/b)+\cos(Y/C)$ ,  $Z=A/2*\sin(X/b)*\cos(XY/BC)+\cos(Y/C)$ ). It includes an 'Enter value of A:' field with the value '5' and 'OK'/'Cancel' buttons.
- Flight Direction:** A small dialog with a 'Clear' button.
- Writing Raster Data:** A progress dialog showing a 32% completion bar.
- Input:** A dialog with a question mark icon and the text 'Enter height of the building (meter)', with an input field and 'OK'/'Cancel' buttons.

# Fractal surface



# Acceleration

The screenshot displays the 'Airborne Altimetric LiDAR Simulator' application window. The main menu includes 'File', 'Menus', and 'Help'. The toolbar contains icons for 'Surface', 'City Model', 'Fractal Terrain', 'Flight', 'Sensor', a coordinate system diagram (Roll, Pitch, Yaw), 'Error Simulation', and 'Output'. The 'Platform Component' pane is set to 'Flight'. Two dialog boxes are open:

- Acceleration Choice:** A dialog box titled 'Acceleration Choice' with the instruction 'Select Acceleration Type'. It features two radio buttons: 'Simulated Acceleration' (which is selected) and 'Without Acceleration'. 'OK' and 'Cancel' buttons are at the bottom.
- Acceleration Parameters:** A dialog box titled 'Acceleration Parameters' with the instruction 'Enter values of acceleration parameters'. It contains a table for 'A, B, C & D values for accelerations:' and a field for 'm'.

	A	B	C	D
ax:	1.05	2.38	0.51	2.77
	0.25	4.45	0.27	3.88
	0.3	80	0.2	100
ay:	0.05	3.38	0.51	3.77
	1.25	2.45	1.07	2.88
	0.3	85	0.2	95
az:	0.85	1.38	1.51	4.77
	0.25	4.45	1.07	0.88
	0.2	80	0.3	100
m:	0.0			

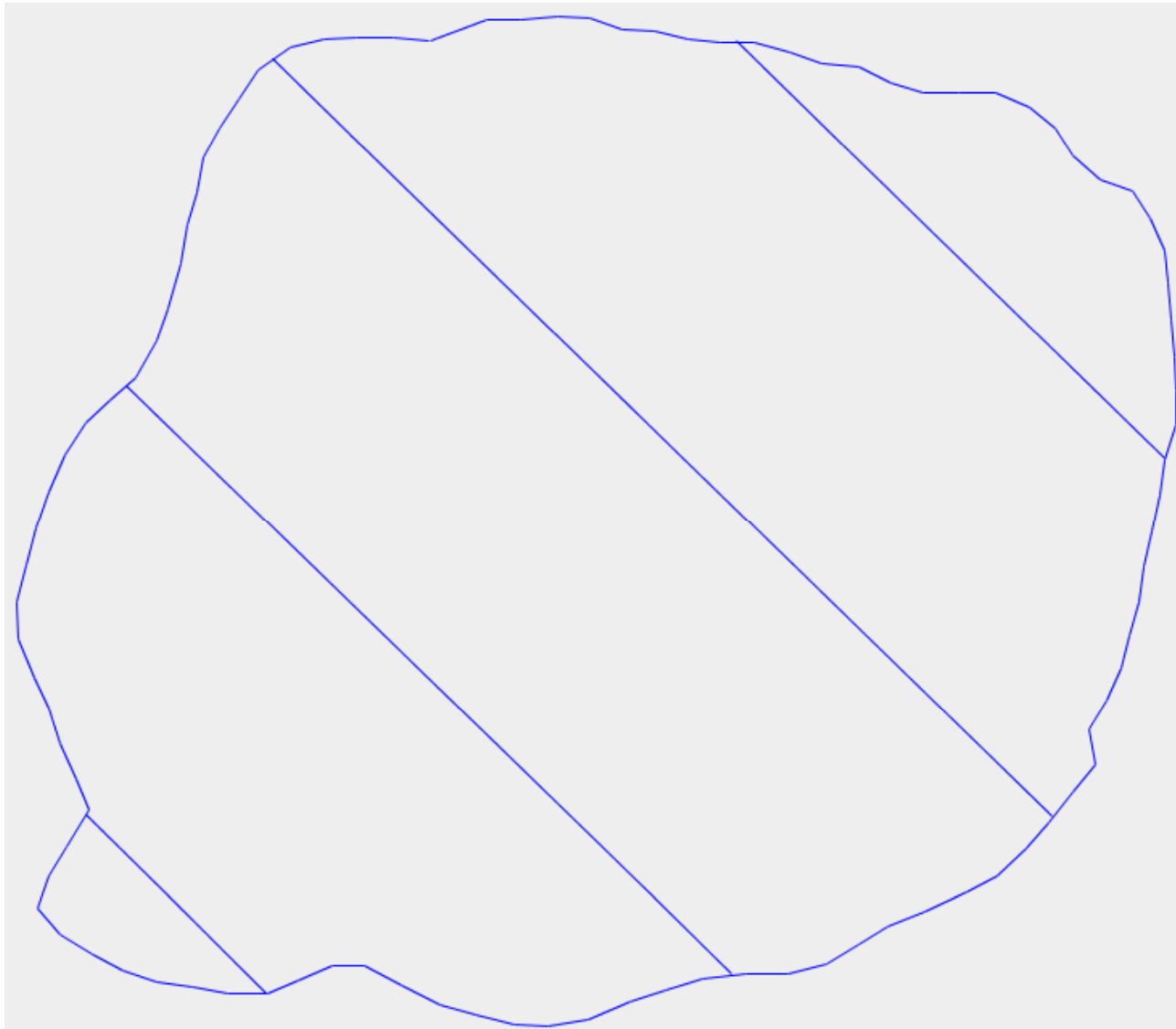
# Sensor component

The screenshot displays the 'Airborne Altimetric LiDAR Simulator' software interface. At the top, there is a menu bar with 'File', 'Menus', and 'Help'. Below the menu bar is a toolbar with icons for 'Surface', 'City Model', 'Fractal Terrain', 'Flight', 'Sensor', a 3D coordinate system (Roll, Pitch, Yaw), 'Error Simulation', and 'Output'. The main window shows a 'Sensor Types' list on the left with 'ALS 50', 'ALTM', and 'Close' options. Three dialog boxes are open, each titled 'Parameters for [Sensor Type] Sensor'. Each dialog box has a 'Flight Plan' section with 'Altitude (m AGL)', 'Overlap (%)', and 'Velocity (m/s)' fields, and a 'LiDAR Settings' section with 'Scan Pattern', 'Firing Frequency (kHz)', 'Scan Frequency (Hz)', and 'Scan Angle (deg)' fields. The 'ALS50 Sensor' dialog includes 'Firing Frequency (kHz)', 'Scan Frequency (Hz)', and 'Field of View (deg)' fields. The 'ALTM Sensor' dialog includes 'Firing Frequency (kHz)', 'Scan Frequency (Hz)', and 'Scan Angle (deg)' fields. Each dialog box has 'OK' and 'Cancel' buttons at the bottom.

## System defined optimal flight lines



## User defined optimal flight lines





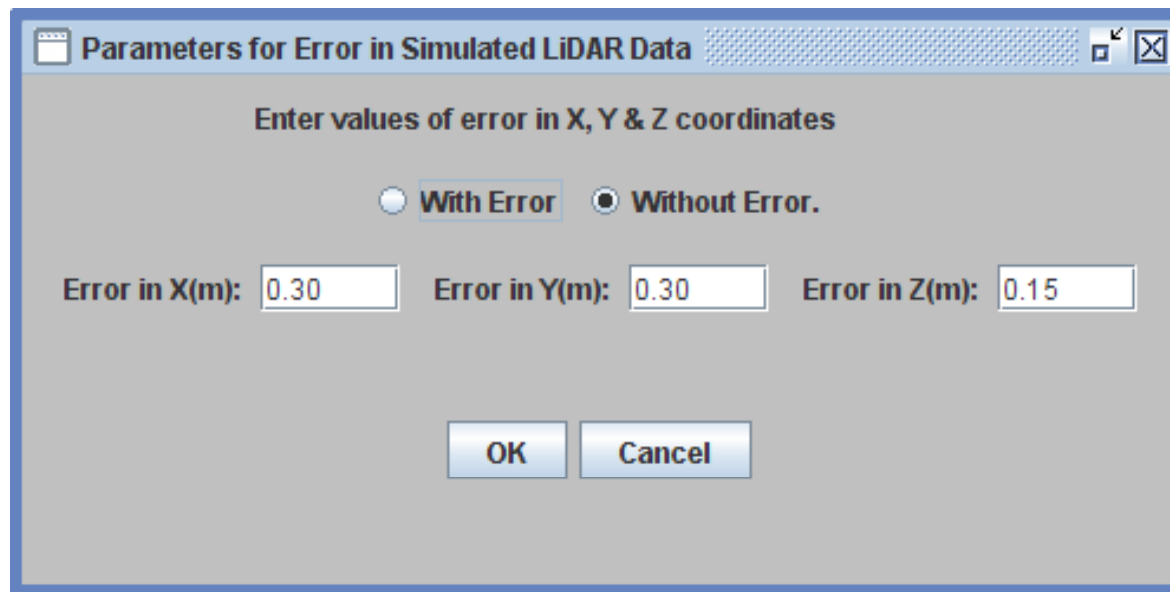
# Attitude

The screenshot displays the 'Airborne Altimetric LiDAR Simulator' application. The main window features a menu bar (File, Menus, Help) and a toolbar with icons for Surface, City Model, Fractal Terrain, Flight, Sensor, a 3D coordinate system (Roll, Pitch, Yaw), Error Simulation, and Output. Below the toolbar, there are two active windows:

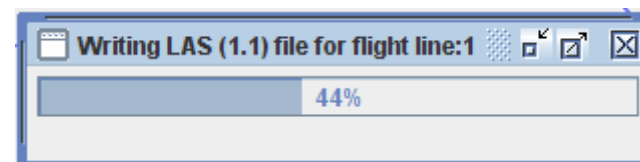
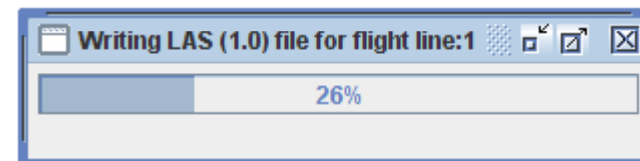
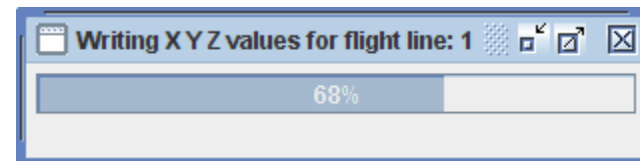
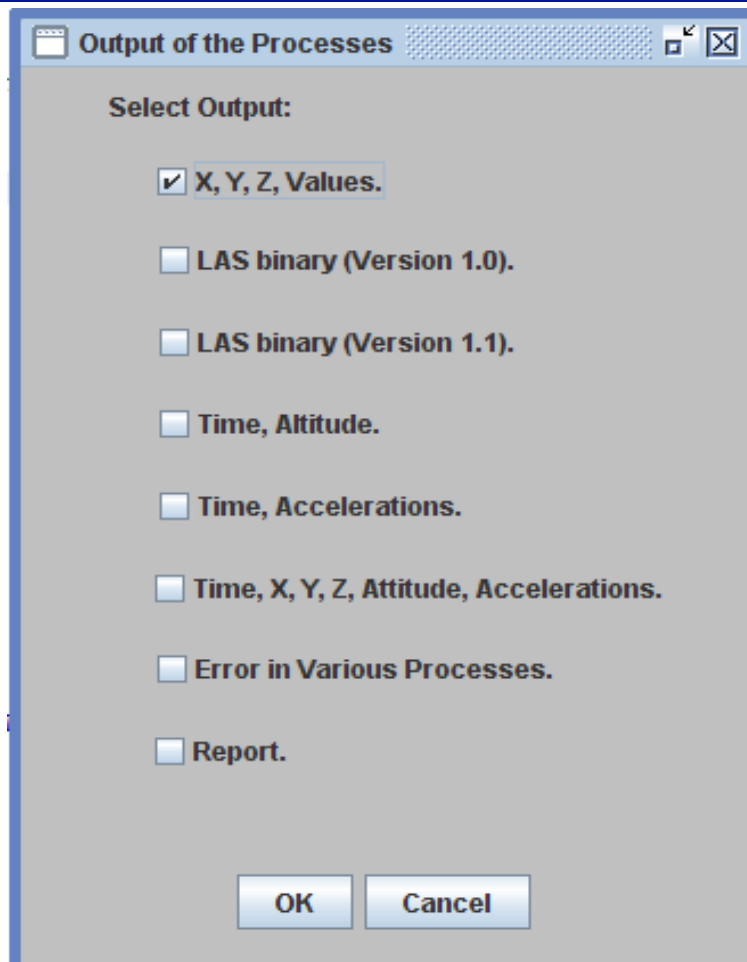
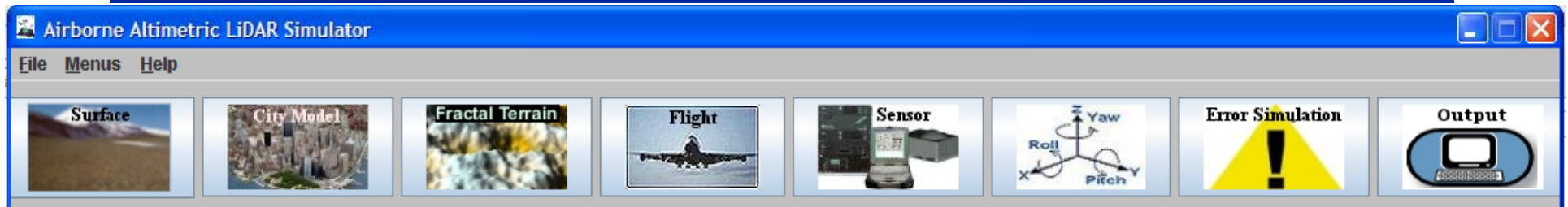
- Roll, Pitch and Yaw Component**: A panel with a tree view containing 'RPY Parameters', 'RPY Simulated', 'RPY from File', and 'Close'.
- Parameters for Roll, Pitch and Yaw (Simulated)**: A dialog box with the title 'Select Roll, Pitch, and Yaw'. It contains two radio buttons: 'Simulated RPY' (selected) and 'Without RPY'. At the bottom are 'OK' and 'Cancel' buttons.
- Roll, Pitch and Yaw Parameters**: A dialog box titled 'Enter values of RPY parameters'. It contains a table for 'A, B, C & D values for RPY:' and a field for 'm:'.

	A	B	C	D
Roll:	0.05	1.38	0.51	1.77
	0.25	0.8	0.27	0.88
Pitch:	0.7	0.5	0.51	1.4
	0.25	0.7	0.57	0.7
Yaw:	0.85	0.4	0.51	1.3
	0.8	0.8	0.7	0.7
m:	0.0			

## Error simulation



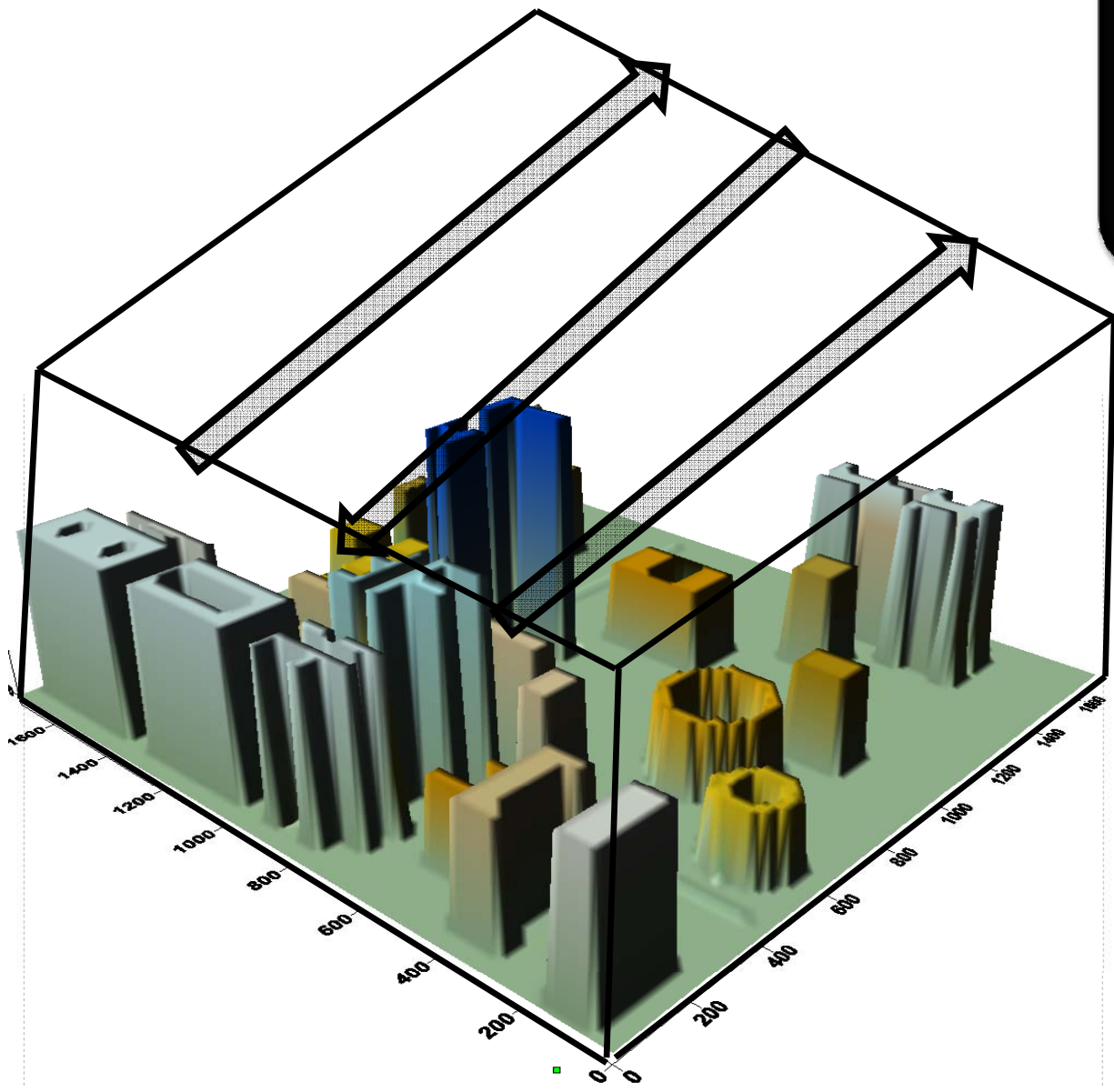
# Output generation



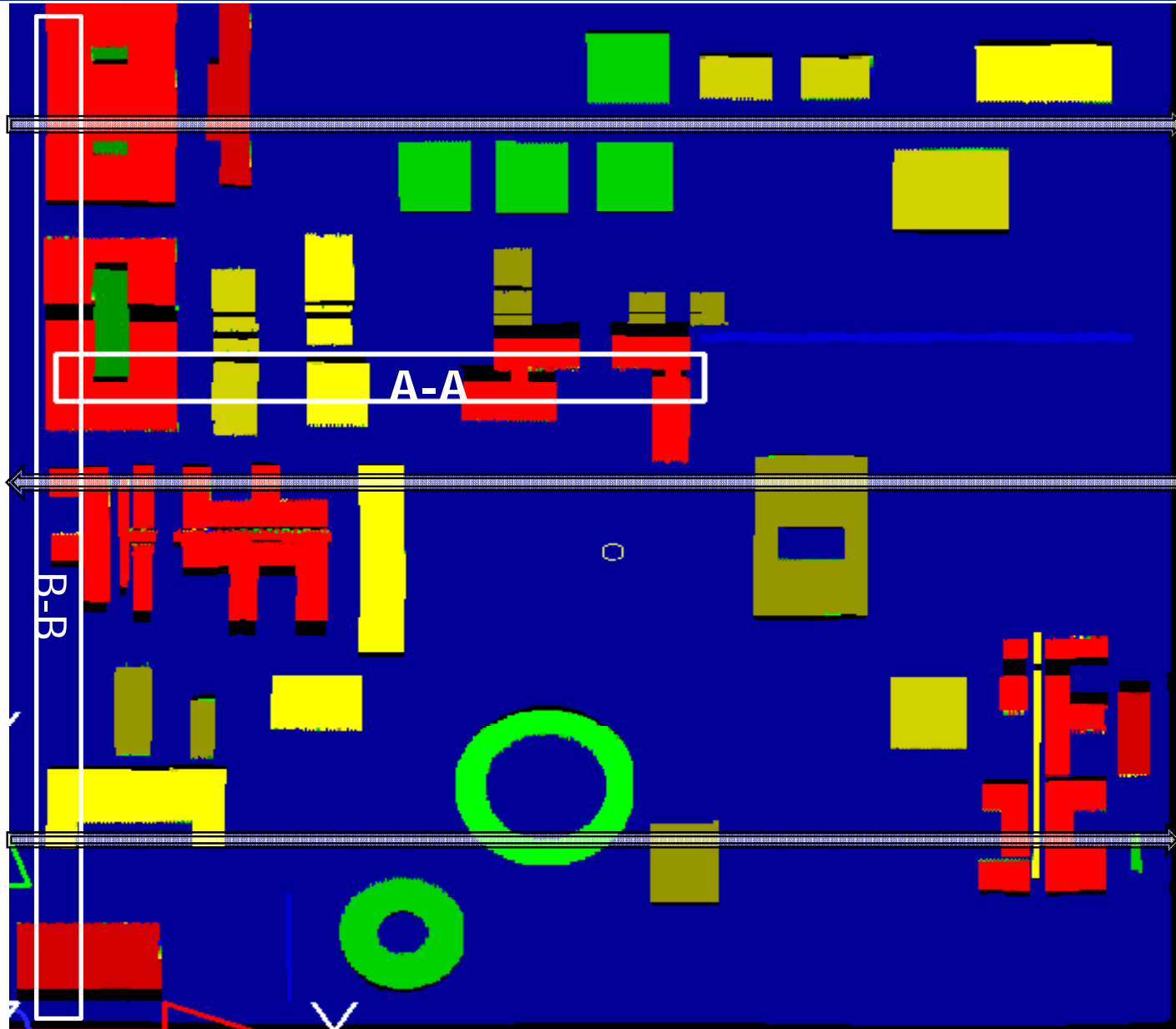
# Simulator Results

# 3D Raster terrain (Displayed in Surfer)

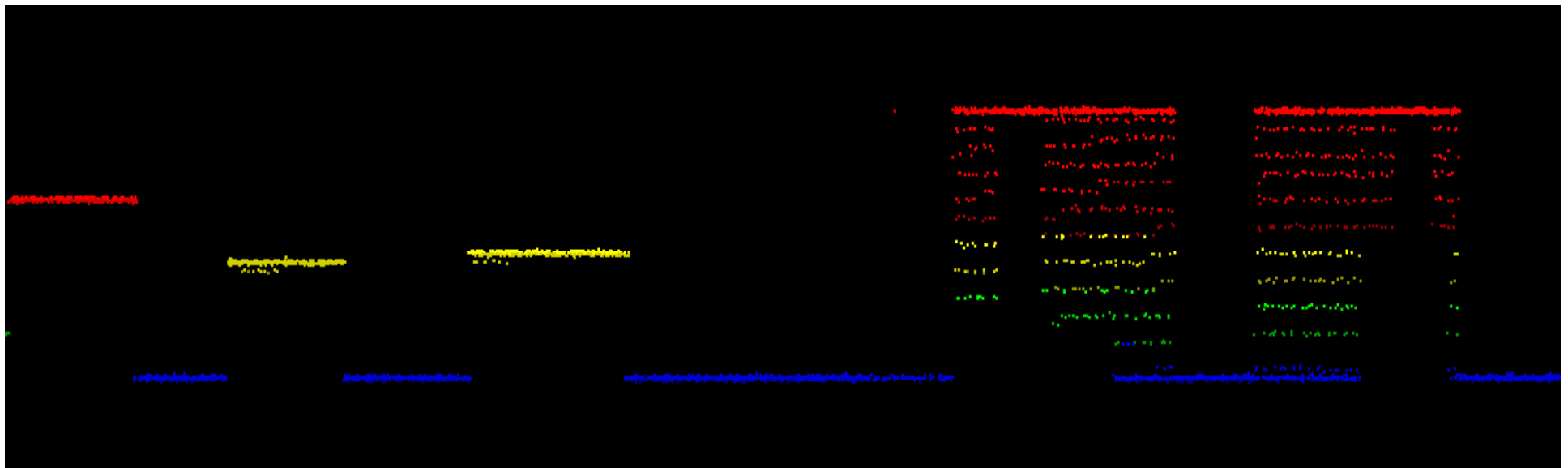
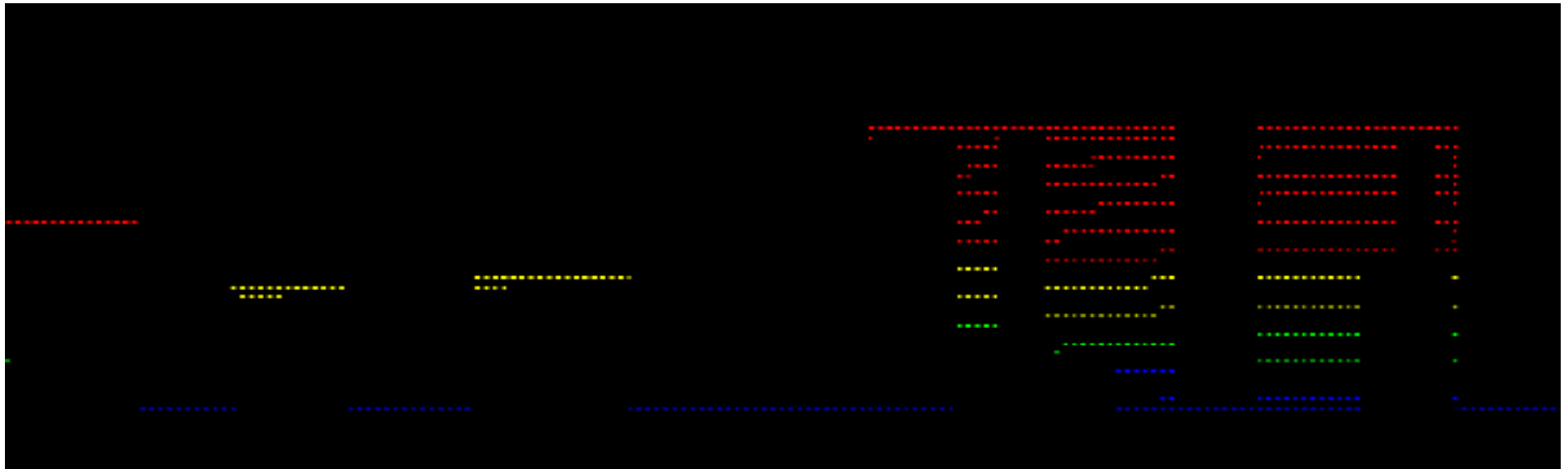
Altitude=210m  
Overlap=4%  
Velocity=60m/s  
Sensor-ALS-50  
Firing frequency=20KHz  
Scan frequency=48Hz  
Scan angle=40°  
Flight area=430m×430m



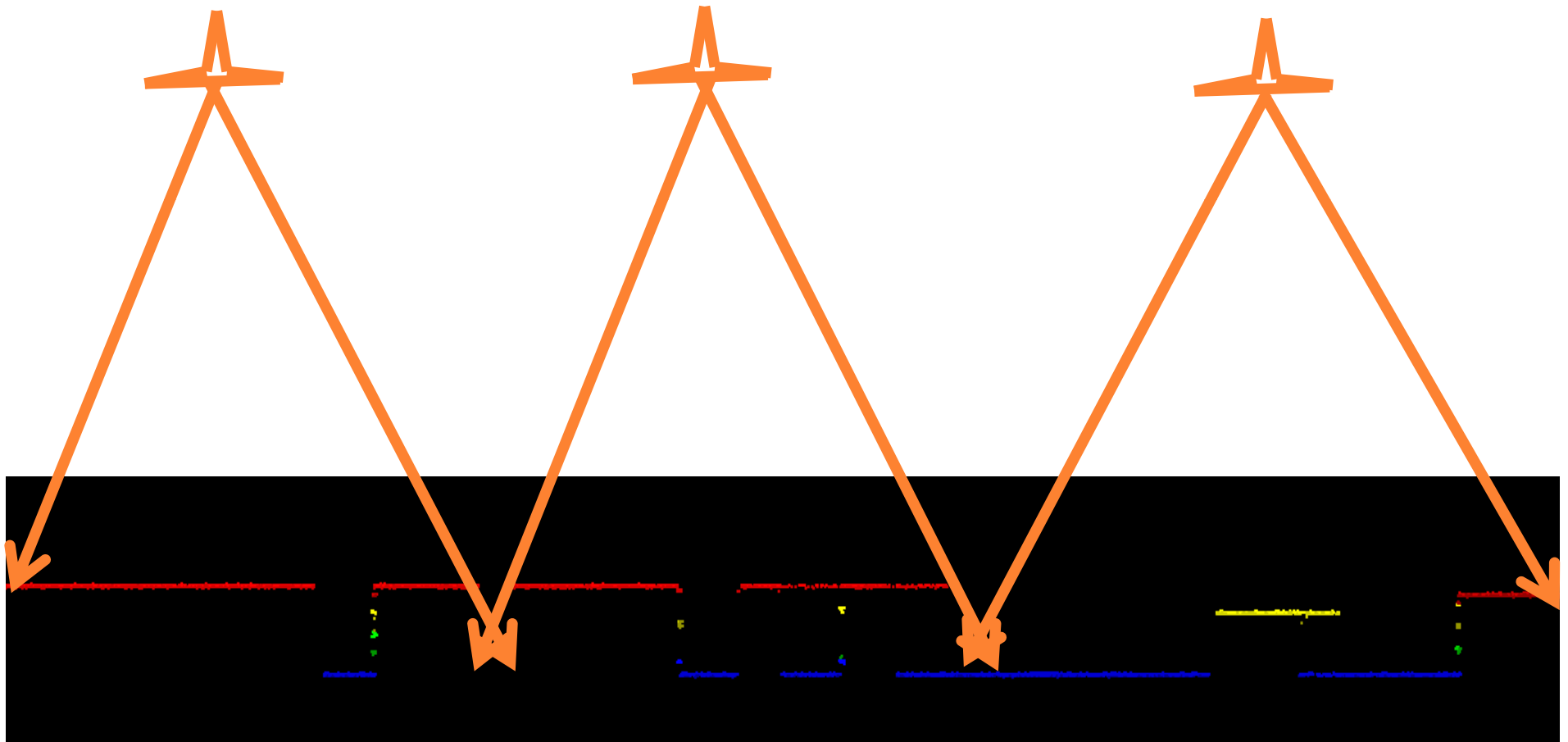
# Lidar data plot in plan



# Profile A-A with and without error



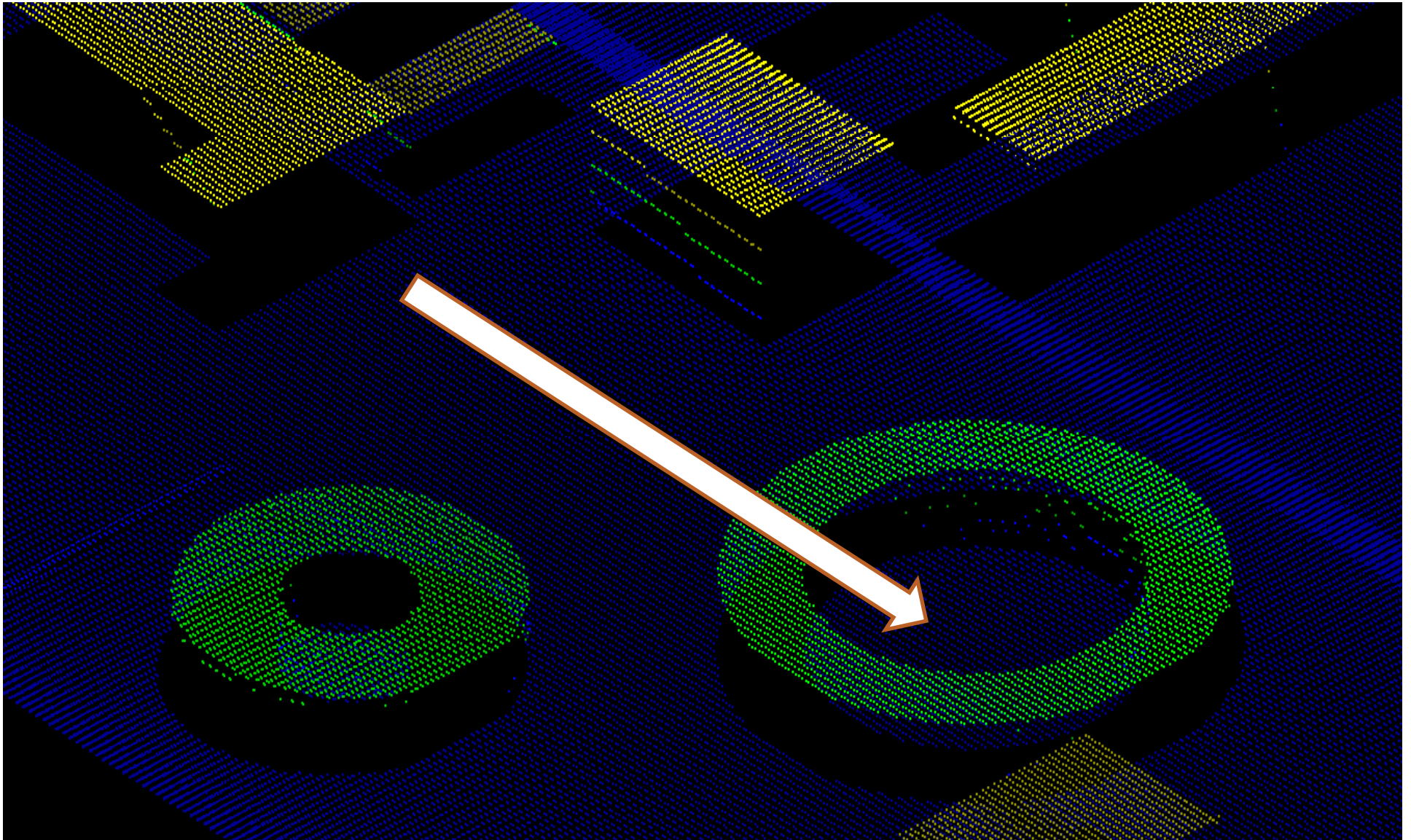
# Profile B-B with respect to flight lines





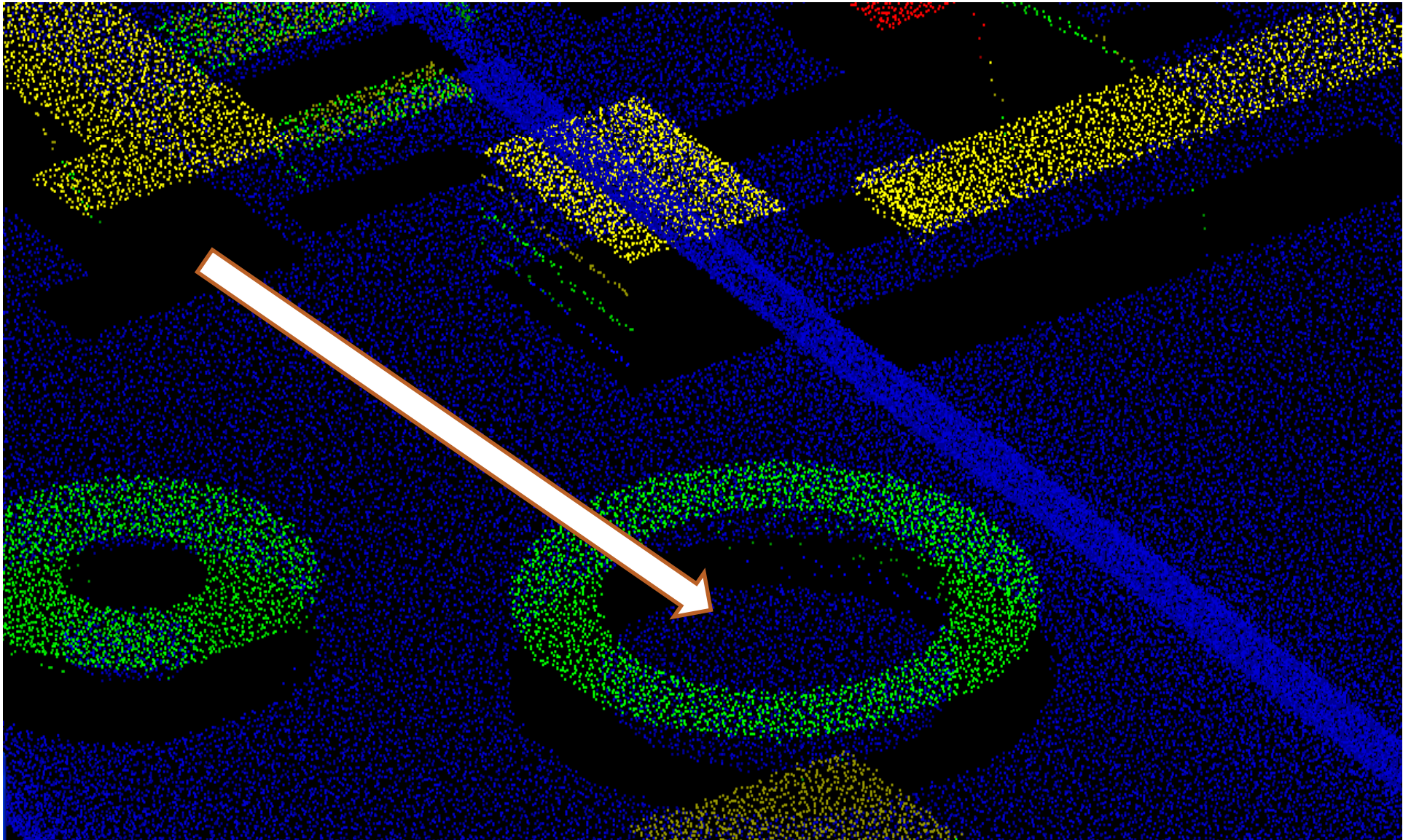
# LiDAR data without error

---



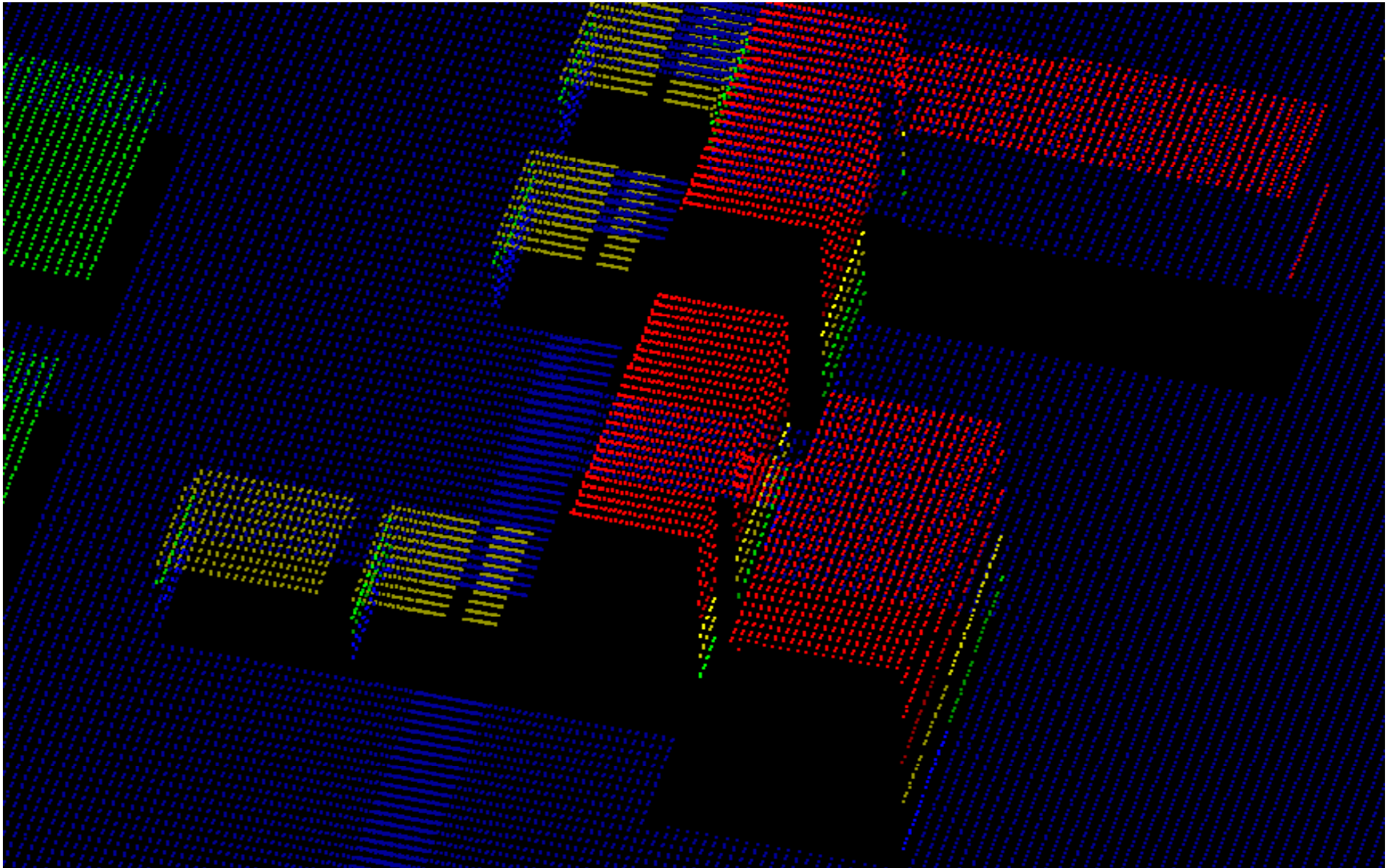
# LiDAR data with error

---



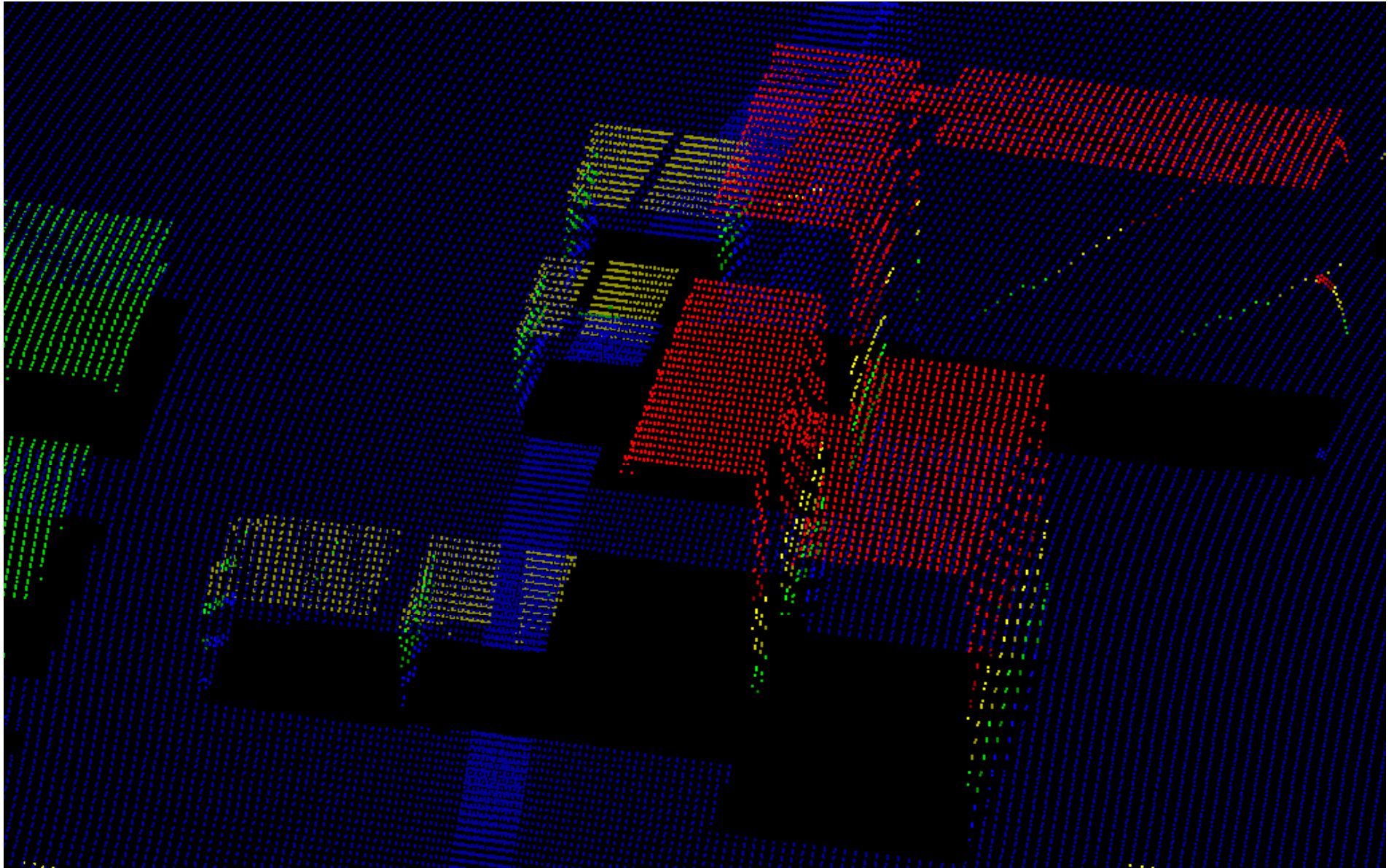
# Data without attitude variation

---

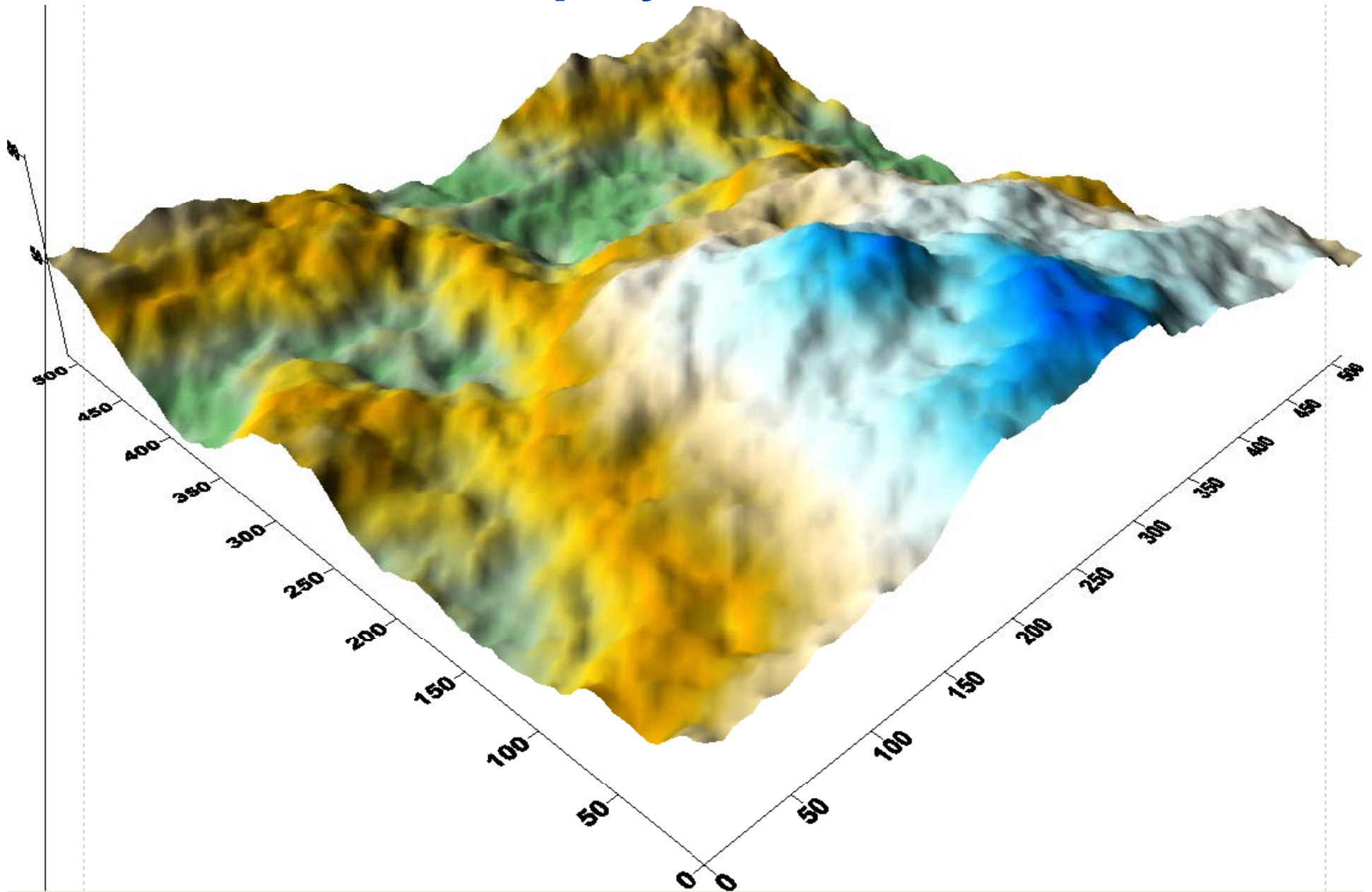


# Data with attitude variation

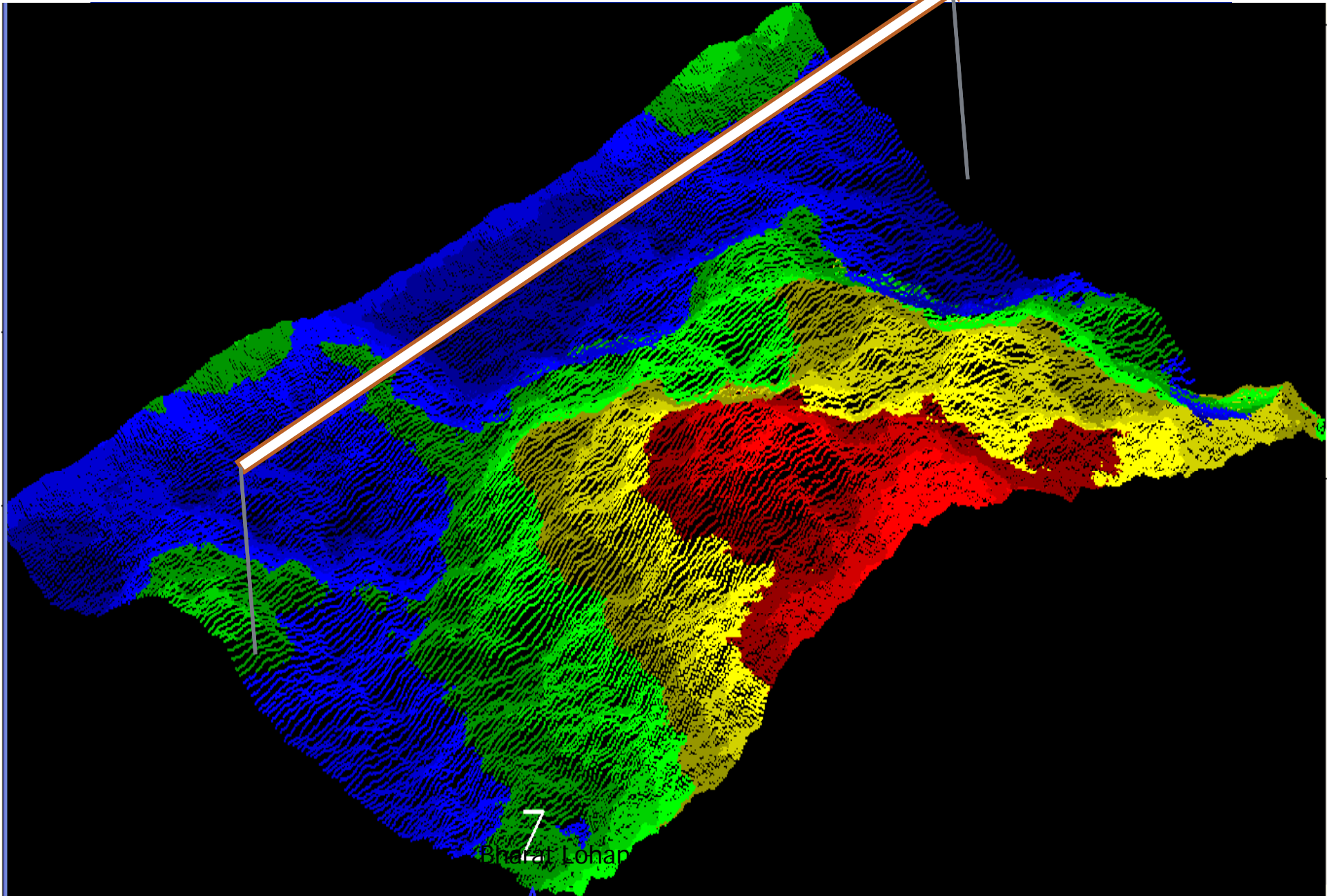
---



# Fractal surface displayed in Surfer

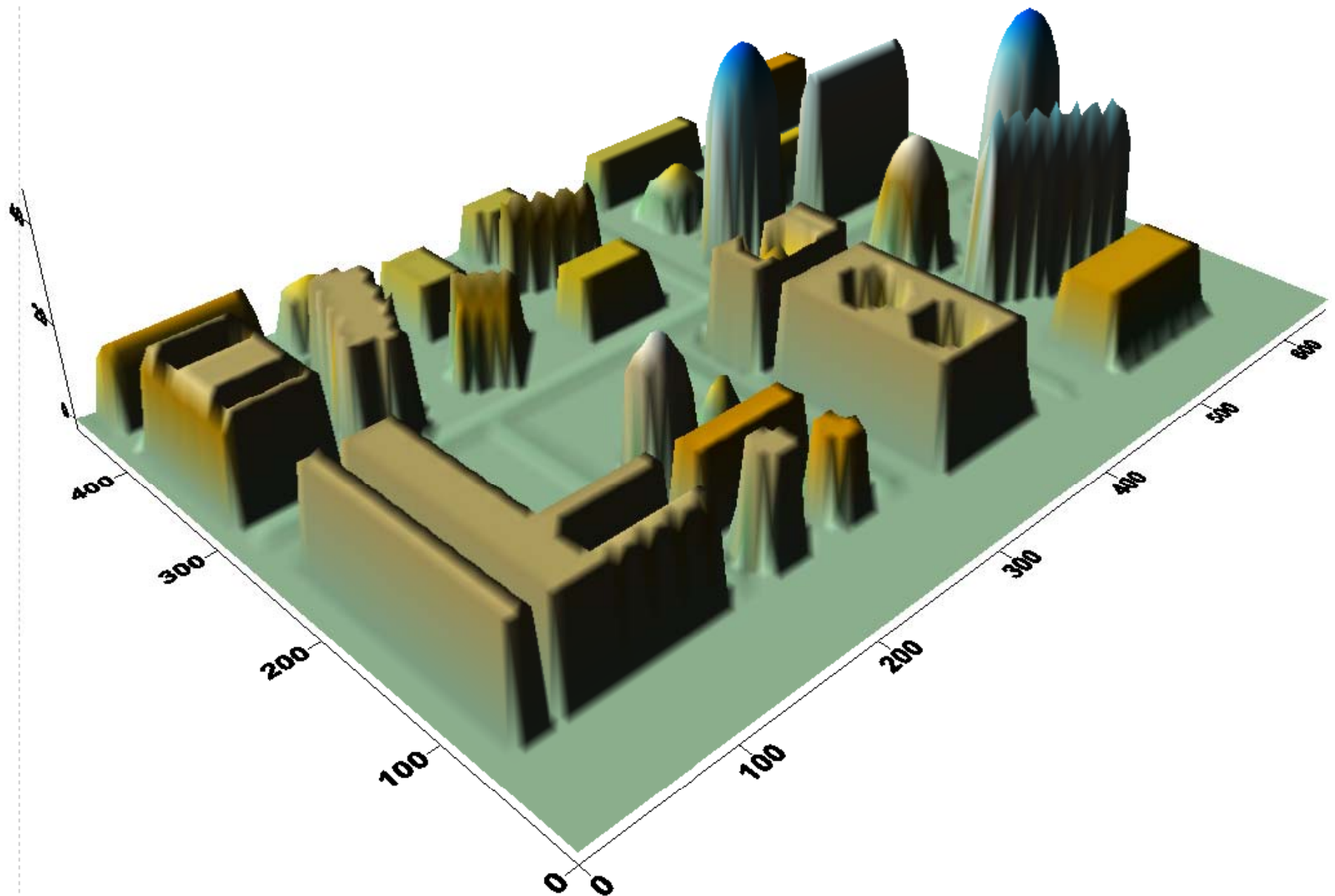


# LiDAR data of fractal surface



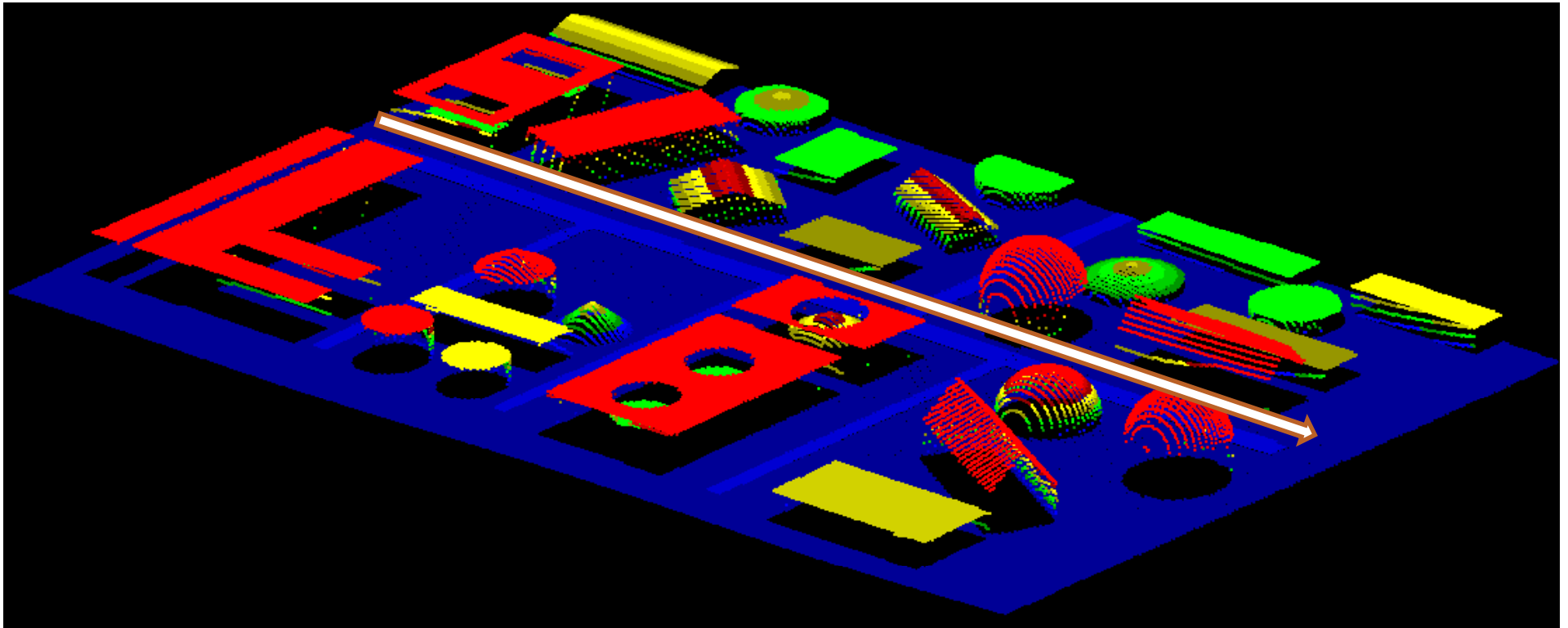
7  
Bharat Lohan

# Terrain with objects

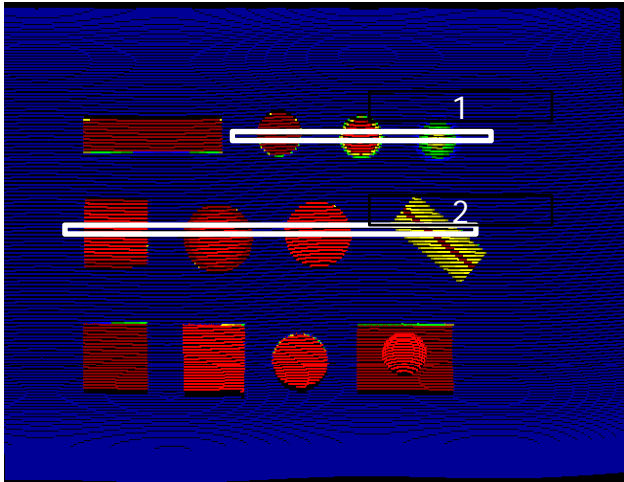


# LiDAR data of terrain with objects

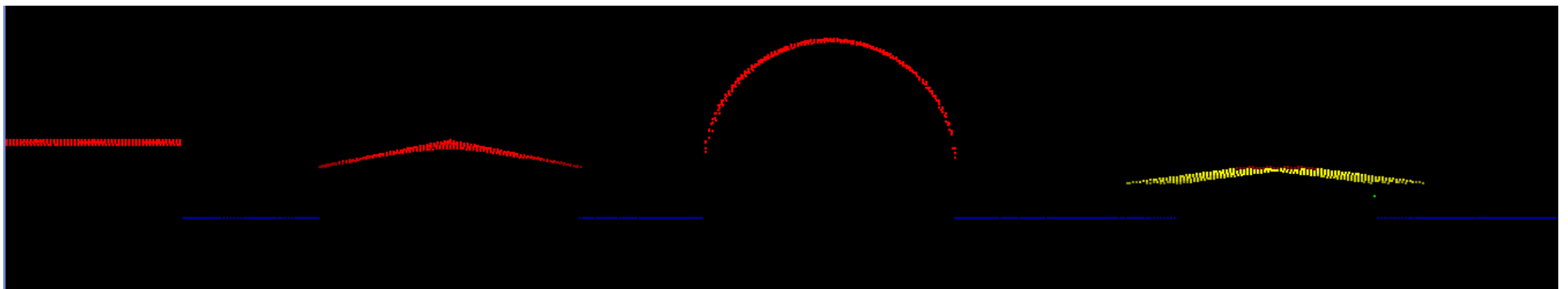
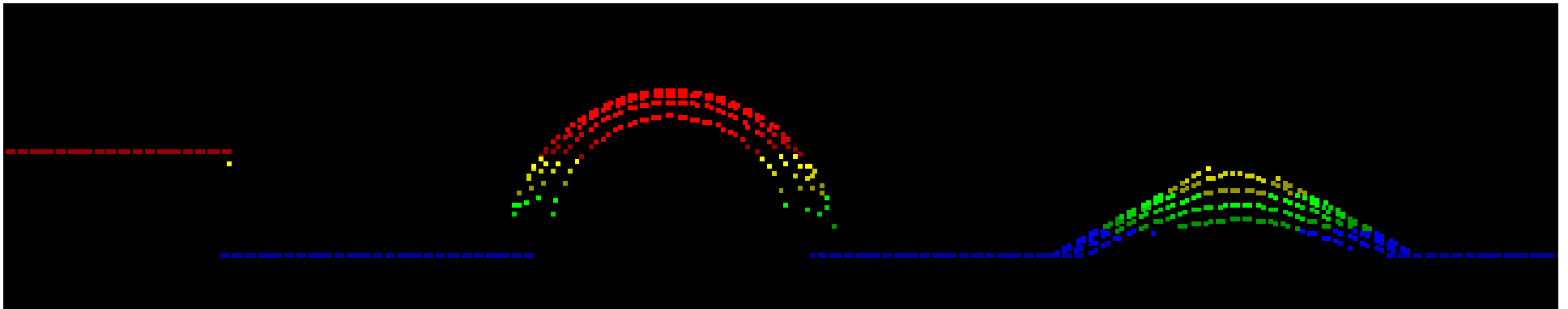
Altitude=490m  
Overlap=2%  
Velocity=60m/s  
Sensor-ALS-50  
Firing frequency=20KHz  
Scan frequency=48Hz  
Scan angle=50°  
Flight area=640m×460m



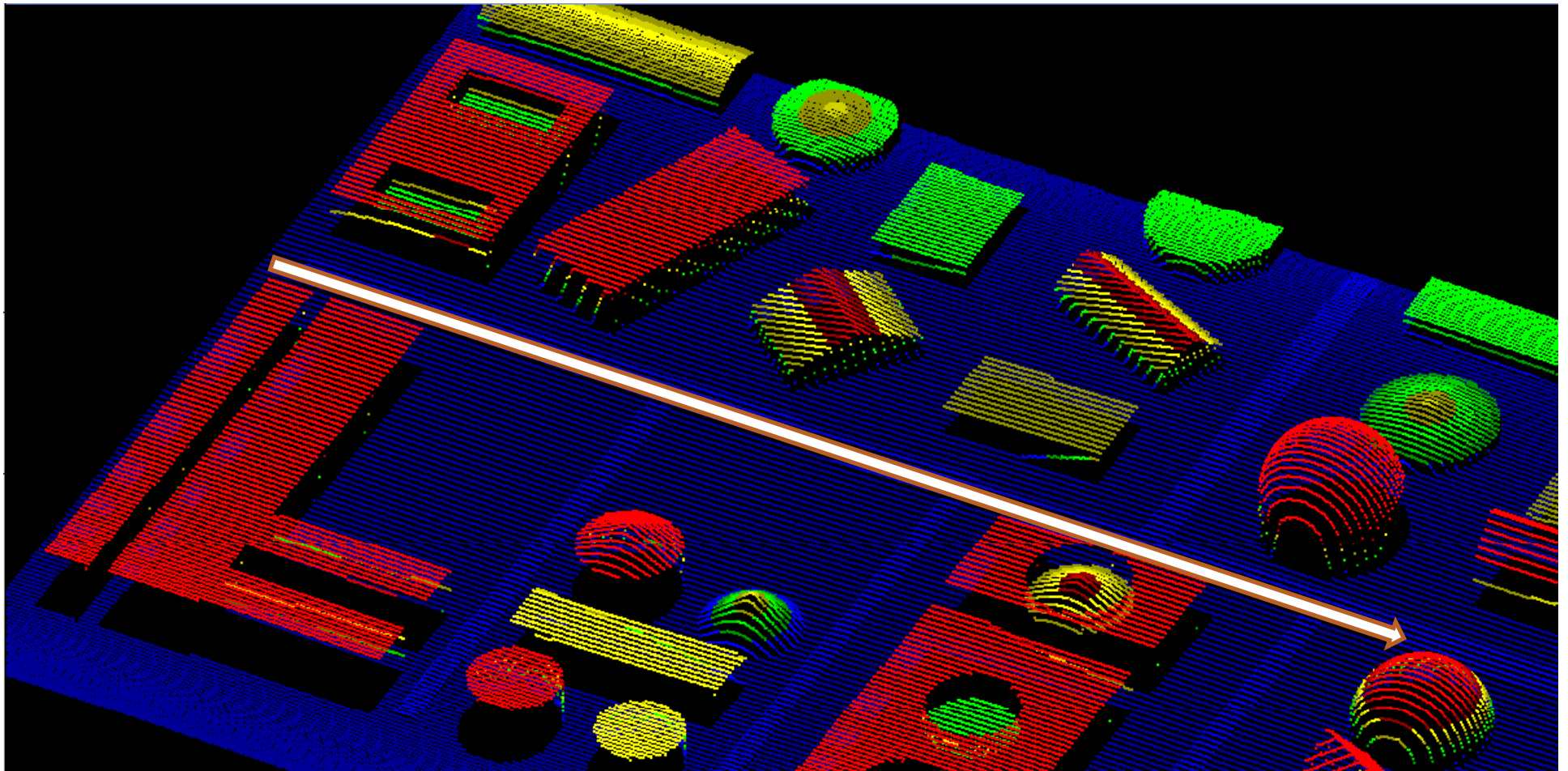


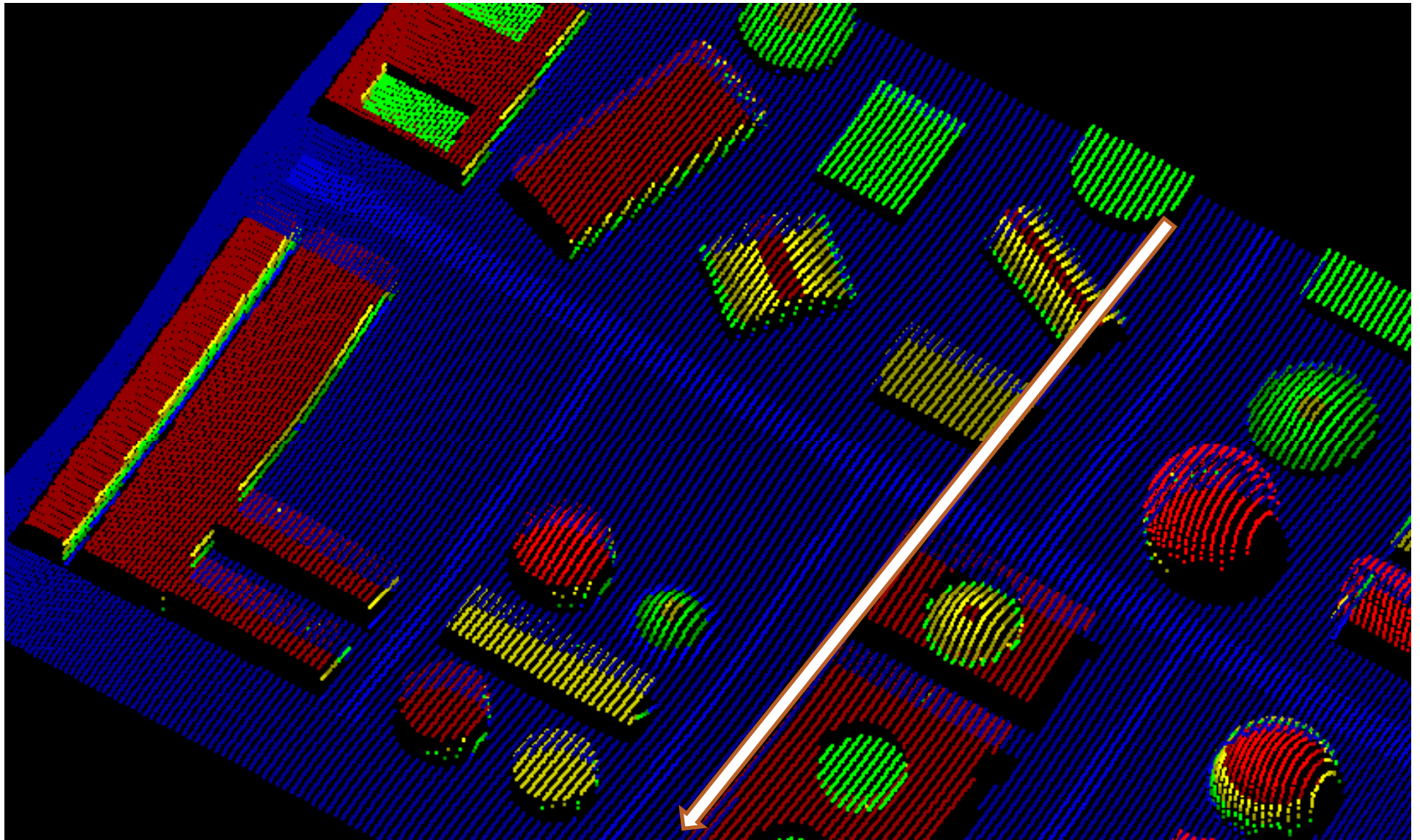


# Profile view of buildings



# Effect of different flight direction





# Use of Simulator

## Education

- Process of LiDAR data generation
- Effect of change in various parameters
- Effect of error in data
- Effect of different sensors on LiDAR data
- Generating data of known ground truth
- Conducting various lab exercises

## Research

- ❑ Evaluation of information extraction algorithms
- ❑ Assessing effect of error on performance of algorithms
- ❑ Study the effect of parameters on data
- ❑ Generate data of different specifications with no cost
- ❑ Finding optimal data specification for an application

## Flight planning

- Determine the optimal flight line
- Effect of sensor parameters on data
- Effect of data density
- Determine the optimal sensor parameters

---

## Conclusion

- ❑ Offers a user friendly GUI based interface
- ❑ Simulate the process of LiDAR data collection
- ❑ Freedom to set the sensor parameters
- ❑ Many data sets can be generated for the same terrain
- ❑ Ideal software tool for LiDAR research and education
- ❑ OOSE makes it easily maintainable and scalable software



**Thanks !!**