MULTI-DIMENSIONAL DATA DISCOVERY

Haregu Ferede, National Geospatial-Intelligence Agency,
PhD candidate at George Washington University
Shahram Sarkani
Thomas A. Mazzuchi
System Engineering Department
George Washington University
1776 G Street NW, Washington, DC 20052
feredeh@gwmail.gwu.edu
sarkani@gwu.edu
mazzu@gwu.edu

ABSTRACT

Laser scanner systems provide economical and efficient three dimensional geopositioning data. These systems have had an increasingly large impact on Topographic Information Systems (TIS) and Geographic Information Systems (GIS). LIght Detection And Ranging (LIDAR) is an optical remote sensing laser scanner that measures the properties of scattered light to determine the range to, and other information about distant targets. LIDAR returns are stored, in a multi-dimensional data set, called a point cloud, containing a location (X,Y,Z) and a number of other attributes (e.g. time, intensity, frequency, etc) for each point. The data volumes created by this technology are enormous and have led for a need for full dimensional data indexing and integrated data applications interpolation algorithms.

A single region may have many point clouds collected from multiple sources over different times and with different resolution/densities. While there is general consensus on file structure and metadata each sensor/platform data product is uniquely tailored to the vendors specific applications often with specific data fields unfilled. Discovery of these point clouds over an area of interest is manually intensive due the lack of multi-dimensional discovery software.

Key words: data management, discovery, LIDAR, multidimensional, 3D

INTRODUCTION

For the past few years Cary, Lidar Market: Status and Growth Trends (2009) and Stennett (2004) found a growing demand for LIDAR hardware/sensors, data, software applications, and products. In addition, LIDAR data collection hardware has become more available due to technology increases and as a result numerous government agencies and commercial companies are collecting and producing many kinds of spatial data. On a daily basis Terabytes of data and metadata are being created. The larger issue is that the data sets are not in a single standard, not in a common data storage environment and therefore not easily discoverable when researching an area of interest (Cary, Lidar Market: Status and Growth Trends, 2009; Stennett, 2004).

Cary in her presentation to the International LIDAR Mapping Forum, found Topography (DEM/DSM generation), Flood risk mapping and Watershed analysis to be the top three applications being identified by more than half of the survey respondents. In addition to these three applications, Tracy and Carbonell from an interview with the author in August of 2009, identified Feature Extraction and Line of Sight to be in the top five LIDAR applications for GEOINT (Cary, The Global Market for Airborne Lidar Systems and Services, 2009; Carbonell, 2009; Tracy, 2009). These five leading LIDAR application can be seen in Figure 1.

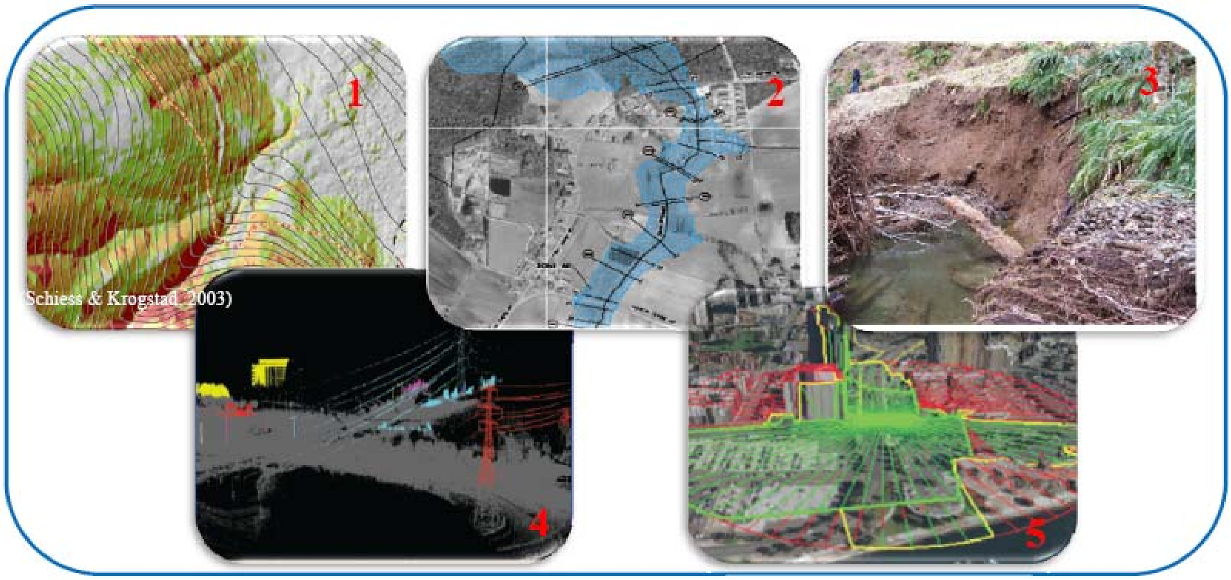


Figure 1. Five Leading LIDAR Applications.

1-- *Topography (DEM/DSM generation) (Peter Schiess and Finn Krogstad Proceedings of the 26th Annual Meeting of the Council on Forest Engineering, Bar Harbor, Maine, USA, September 7-10, 2003)*

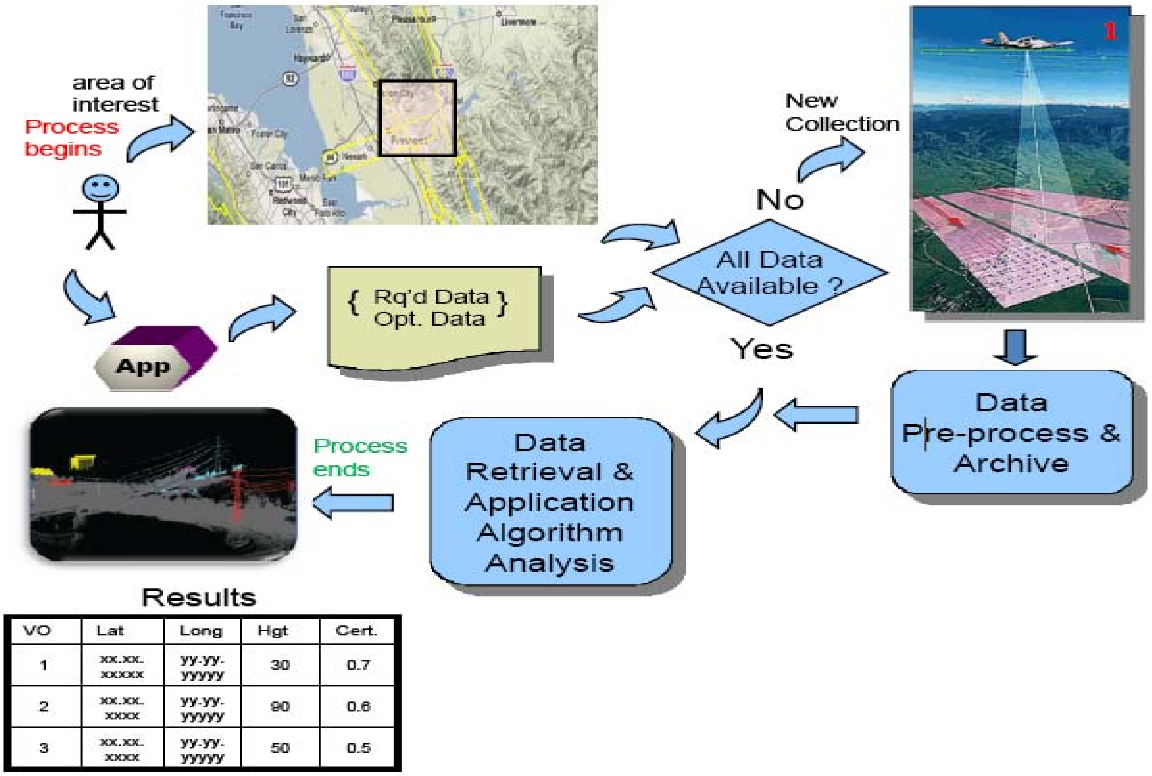
2-- *Flood risk mapping (United States General Accounting Office FLOOD MAP MODERNIZATION Program Strategy Shows Promise, but Challenges Remain)*

3-- *Watershed analysis (Wilson River Watershed Analysis FINAL – March 2008 -- Duck Creek Associates, Inc)*

4—*Feature Extraction (LIDAR and Compressive Sensing Myron Z. Brown, 26 February 2009; Compressive Sensing Workshop. Work performed under contract to NGA by M.Z. Brown; NGA Contract: HM1582-0-R-0004)*

5—*Line of Sight (NGA has created 3D virtual analytic environments of various U.S. cities, including Washington, D.C. Line-of-sight analysis for San Diego, California, conducted in preparation for security at the 2002 Super Bowl) BEAULIEU, B. R. (2004, JUL/AUG). Homeland Security through The Palanterra. GeoIntelligence . Wasington, D.C.: www.geointelmag.com*

LIDAR data has numerous applications but the workflow for an application generally follows the path as illustrated in Figure 2. In each application, the analyst must identify the geographic area of interest, identify specific data needs and availability in that region, retrieve the data elements that are applicable, present those data to the application algorithms and generate the application results in a form for the customer(s). This paper is primarily directed toward discovering and retrieving the data while the specifics of each application are left to future discussions. Using the LIDAR applications above and three examples “Figures 5-7”, these workflows are illustrated to describe where data sets were acquired (geographically), when the data sets were collected (date/time), how the data was sensed (instrument, flying height, orientation, etc), for what purpose the data was obtained (mission) and the level of processing that has occurred on that data. The discovery flow path taken for the applications is shown in Figure 2.



VO	Lat	Long	Hgt	Cert.
1	xx.xx. xxxxx	yy-yy. yyyyy	30	0.7
2	xx.xx. xxxx	yy-yy. yyyyy	90	0.6
3	xx.xx. xxxx	yy-yy. yyyyy	50	0.5

Application	Required Data Types	Optional Data Types
DEM Production	Point Cloud (X,Y,Z), Breakline data	Previous DEM data Shoreline data, spot elevations
Flood Risk Analysis	DEM Data Flood Plane delineation	Point Cloud Data, Spot elevations
Watershed Analysis	DEM Data, Streamline vectors, Rainfall area defn.	Snow depths
Feature Extraction	Point Cloud Data	
Line of Sight	Point Cloud Data, POV	DEM, TINs, etc.

Figure 2. Workflow chart for primary LIDAR applications.

1-- *LIDAR Collection Image obtained from the United States Department of Agriculture, Natural Resources Conservation Service at http://www.wy.nrcs.usda.gov/wygis/Lidar.gif*

Three search scenarios are used to discover data for the three out of five LIDAR applications shown in Figure 1. Once the application is determined, a query is done on the database to determine whether appropriate data exists. If the data exists, then it is retrieved and delivered. If not, the appropriate tools are found from the toolkits to create the necessary request for new data.

THREE DIMENTIONAL DATA DISCOVERY

Spatial data, depending on how it is collected can exist as geometry, topology, raster (a grid of squares) or point cloud data. In order to access and manage the data, the range of applications that are part of the LIDAR user's toolbox have to be reviewed and understood. The two dimensional (2D) and two-and-a-half dimensional (2.5D) models can be represented as a Digital Elevation Model (DEM) (as raster, a grid of squares) or Triangular Irregular Network (TIN). TINs, which are vector representations of DEMs, "cannot be update[d] or manipulated efficiently" (Schön, Bertolottoa, Laefer, & Seán, 2009). Alternatively, GIS has only recently addressed LIDAR, three-dimensional (3D) spatial data support. To date, GIS does not fully support meaningful analysis of these data sets.

The LIDAR instrument is a scanner that is typically mounted in the belly of an aircraft and flown over the area to collect data. A typical sized area might be, in the commercial world, the size of a county while the area that a LIDAR scanner can collect at one time is constrained to a path under the aircraft.

LIDAR returns are stored in a point cloud, a multi-dimensional data set containing, for each point, a location (X,Y,Z) and some number of attributes (e.g. time, intensity, frequency, etc). Point cloud data for the same region could be collected with different means, at a different time and for different missions and be stored in multiple forms and locations. The data volumes created by this technology are enormous. LIDAR acquisition over 1200 plus km^2 is approximately 8.3 billion LIDAR returns which creates over 3 times more data than produced by all the instruments combined on NASA's flagship Earth Observing System Satellite over the course of a full year (Crosby, 2006). These facts and the large user community with variable needs and levels of sophistication have led to higher demands for analyzing and interpolation algorithms as well as full multi-dimensional data storage need.

The most common transfer format for LIDAR data is ASCII or LASer (LAS). The file may or may not be stored in ".LAS" file format. Transfer format is used to move LIDAR data from one place to another. Most of the time, the "storage" format used in commercial systems splits the metadata and the point clouds rather than insisting upon keeping "pure" LAS format files that have to be searched sequentially. LAS is a binary format standard for the American Society for Photogrammetry and Remote Sensing (ASPRS). It is the most used with the LIDAR community and is the primary format which holds point cloud data type.

LAS 1.3, the latest standard which was released in July 2009, includes the ability to include pulse waveform data, storage of parameters necessary to geospatially traverse waveforms and additional global encoding flags for synthetically generated returns. The version consists of a public header block, variable length record (including waveform packet descriptors), point data record, and extended variable length record which are all in "Little Endian" format (a way of storing bytes in computer memory) (ASPRS, LAS Specification Version 1.3, 2009). The header block consists of generic, required and non-required public blocks. The non-required public header block, which is not used, must be zero filled. The header block includes File Signature (which is identified by "LASF"), File Source ID, and Global Encoding, X, Y, Z, as well as about two dozen other items. Six of the twenty-four are not required but must be filled with zero if no value is given. This and the previous version not only identify the hardware but they also identify whether or not the file is created from extraction, merging or by modifying existing data (ASPRS, LAS Specification Version 1.2, 2008). This point is essential for discovery of the data and where it originated.

Digital Elevation Model (DEM) data is required to create topography maps (contours). DEM data is also the base for flood risk map and Watershed analysis. Besides the DEM, creating a flood risk map will also require some knowledge of the flood plane regions (i.e. the legal definitions of the areas that have been determined by previous surveys to be in the insurance risk areas). Both Feature Extraction and Line of Sight will require the point cloud data or 1st return DEM if available. For Feature Extraction, the flight line path is also required. For Line of Sight, point of view data is essential to create the Feature Extraction output.

The selection criteria required for discovery, takes into account features which are required for these applications, (e.g. power lines or fences for Line of Sight application). Depending on the application, from the five LIDAR applications shown in Fig 1, requirements for the selection could involve supplementary information and available tools and the query will require different index parameters to create the applications from point cloud data.

Point cloud data contain information within the 3D sample space. At present it cannot easily be stored in a straightforward manner in traditional or object Relational DataBase Management Systems (RDBMS). Unlike the data from a terrain matrix or an image, point cloud data is not on a predefined information grid. This data management difficulty presents the technical challenge in the timely discovery of the data. Each application in Figure 1 would require different selection criteria depending on the necessary or required spatial data for describing the application. The "Data Retrieval & Application Algorithm Analysis" process step shown in Figure 2 will provide specific data required for the analysis.

Of course, the primary data type that must be held and manipulated is the first generation processed LIDAR data – that is, point information of the form: X, Y, Z; Time; and other Attributes. While this data is certainly voluminous, it is literally a list of records – typically millions (or billions) of records that come from a LIDAR first phase processing output. This data type can easily be represented as a list and, depending upon the application usage, would be ordered with a set of indices such as:

LIDAR Point Datatype:
< X >< Y >< Z > < T > < attribute 1 > < attribute 2 >…< attribute N >

where X, Y, Z are the three-dimensional data points followed by a number of attributes such as time, color and intensity depending on the collecting system.

In building a database of such records, the data types identified in the database tables or file need to be assessed for the operations that are critical for performance of the five applications. At a low level, these operations consist of, at the minimum:

1. Selecting records
2. Inserting/update records
3. Deleting records
4. Extracting/retrieving records
5. Combining/joining records

After reviewing the requirements and the necessary function the data will be indexed depending on the operation required and the database used. As initial criteria, indexing will be performed based on performance and memory usage and complexity of the query without taking advantage of larger direct access memories and higher processing rates. The evaluation of complexity will take into account whether or not there are secondary storage devices required and what that does to the overall system performance. Record selection will be the only one discussed in this paper. Figure 3 shows a simplified entity relation diagram showing the logical view for the record selected.

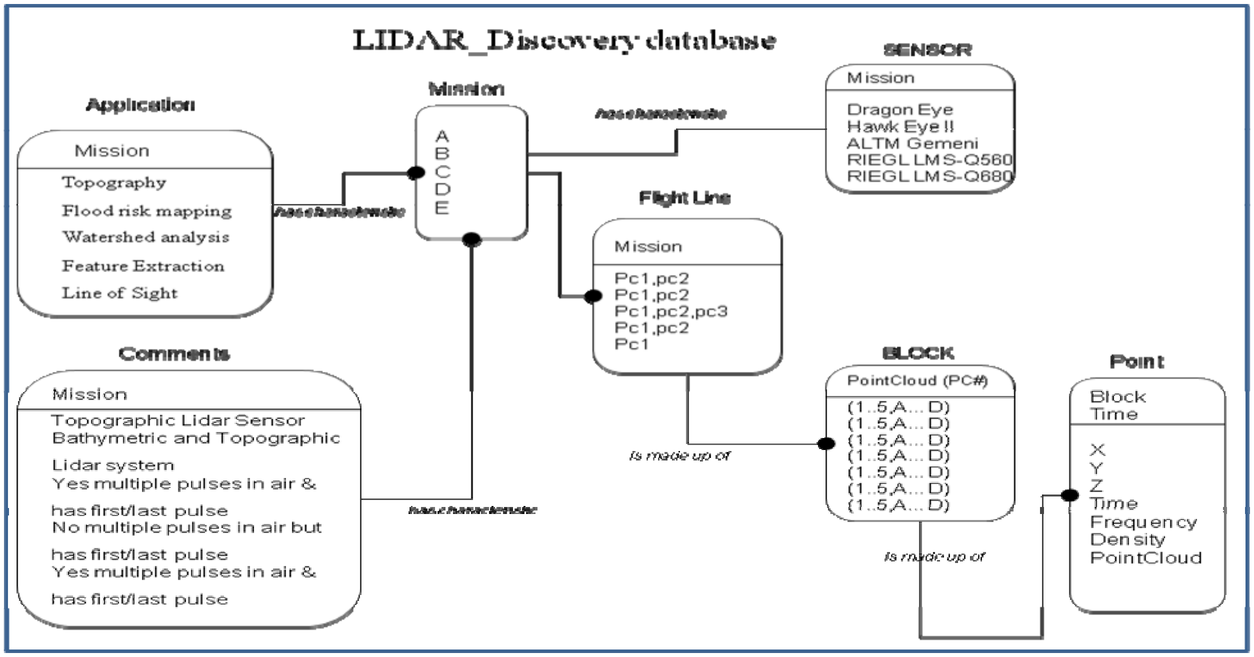


Figure 3. Entity relationship diagram for the logical view of data flow.

Figure 3 illustrates a mission consists of several flight lines. Flight lines are made up of several points which contain the metadata which is the data about the data. It also contains the point cloud data which could be either in the database, as it is in Oracle 11g, or in a flat file in most other databases. These points are organized in several blocks.

Using the illustration in Figure 4, it is easy to understand that the first part of the Selection step will be to identify the appropriate portion of the flight lines or swath that are within the region of interest for the application.

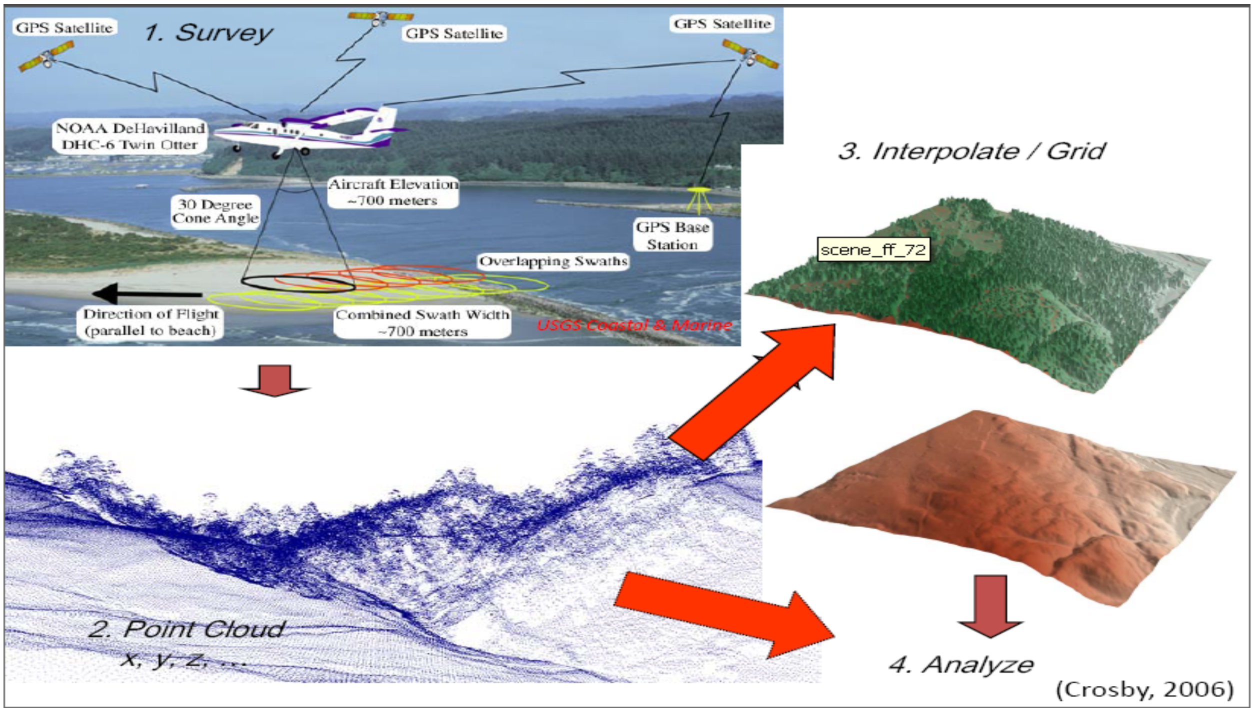


Figure 4. LIDAR flow chart (Crosby, 2006).

For example, the Flood risk mapping application would need to identify those points that are within some distance from a river/stream that the analyst was using as the source of the flood. From this quick analysis, it is clear that there needs to be some kind of an index to the set of data points that includes such elements as:

Initial Datapoint Index:
< Mission ID >< Flightline ID >< Footprint of Flightline >< Collection Date >< Point Record IDs >

Rather than listing all of the points that belong to a Mission / Flightline combination, the datapoints could be organized in some manner that would allow selection / retrieval of "all" the points within a particular Mission / Flightline by just using the index of a particular block. Taking this concept to a rational extension point, the data could further be "blocked" within a Mission / Flightline such that one is able to select and retrieve those subsets more easily. In as much as the entire point dataset may encompass literally billions of points, there is a clear motivation to be able to address various subsets of that dataset without resorting to individual point identifiers. This is the path taken in most GIS available on the market today where the database content is, in some manner, blocked for quick access to the data records according to location of those objects.

Realizing the need for some kind of a data blocking strategy, it is then clear that to facilitate access a datatype/index is needed at the data block level. This could be implemented with a set of records such as:

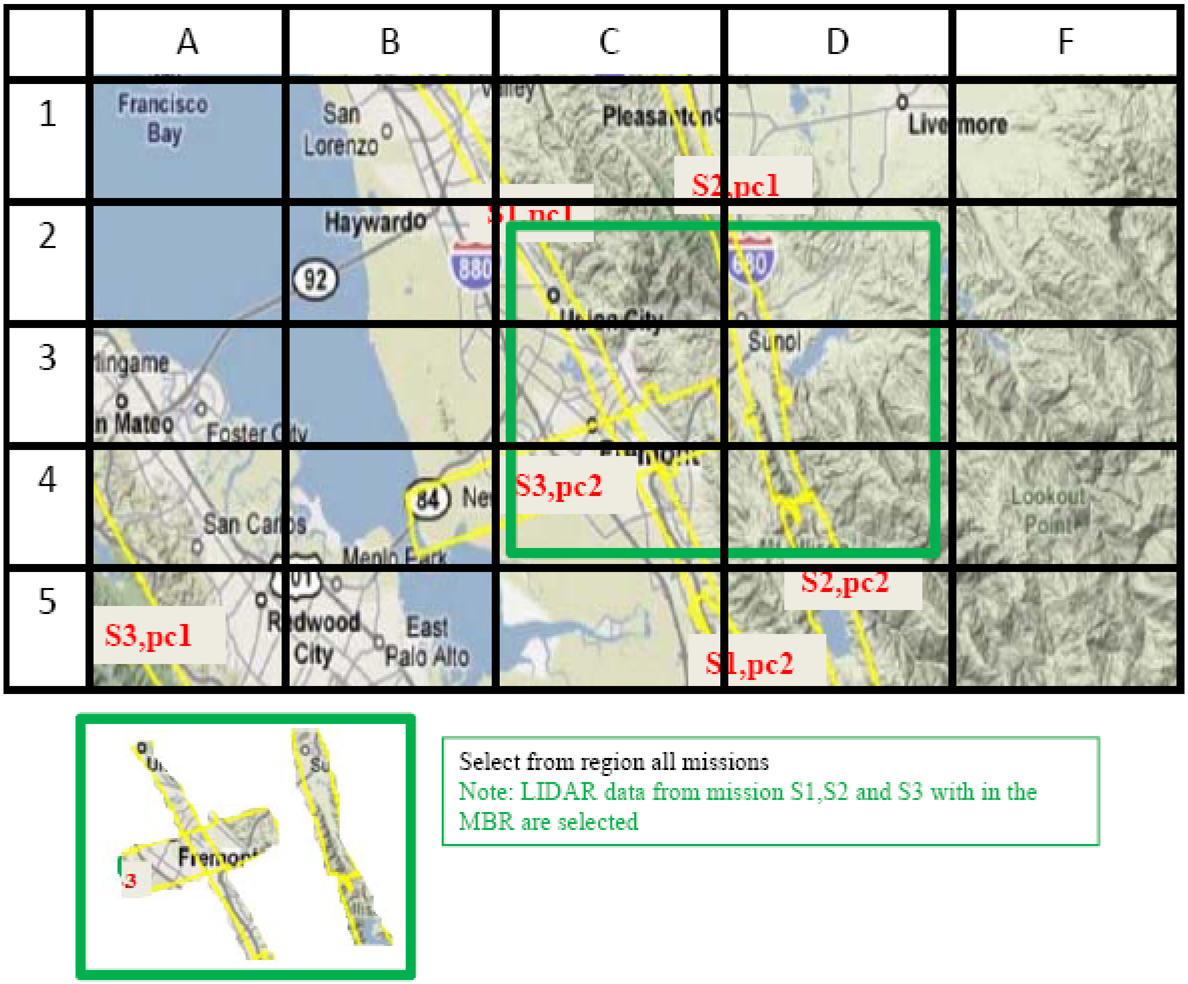
Access Block Index:
< Mission >< Flightline >< Block Geographics >< Block ID >

where Block ID is the index that points to the database location of the datapoints that are contained within a particular Block and Block Geographics are the coordinates of the footprint of that Block data. Going down this path requires that the database now has a function that can take the Mission / Flightline IDs and a set of coordinates of interest and generate the Block Geographics information such that this set of records can be used to retrieve the data. Of course, one would also need an inverse function to be able to ask the question of what flight lines are contained within a given block.

EXAMPLES

Example 1, Figure 5, illustrates a selection for a region in the green minimum bounding rectangle (MBR) at a particular date for a topographic application. In this case, all of the mission data was used, and if the topographic DEM does not exist, then available tools are selected and the DEM is created per flow chart illustrated in Figure 1.

Access Block Index: e.g. 1
< S1 & S3 >< s1(pc1) & S3(pc2) >< Green MBR >< 9-11-09 >< [(2,C),(2,D),(3,C)(3,D),(4,C),(4,D)] >



Select from region all missions
Note: LIDAR data from mission S1,S2 and S3 with in the MBR are selected

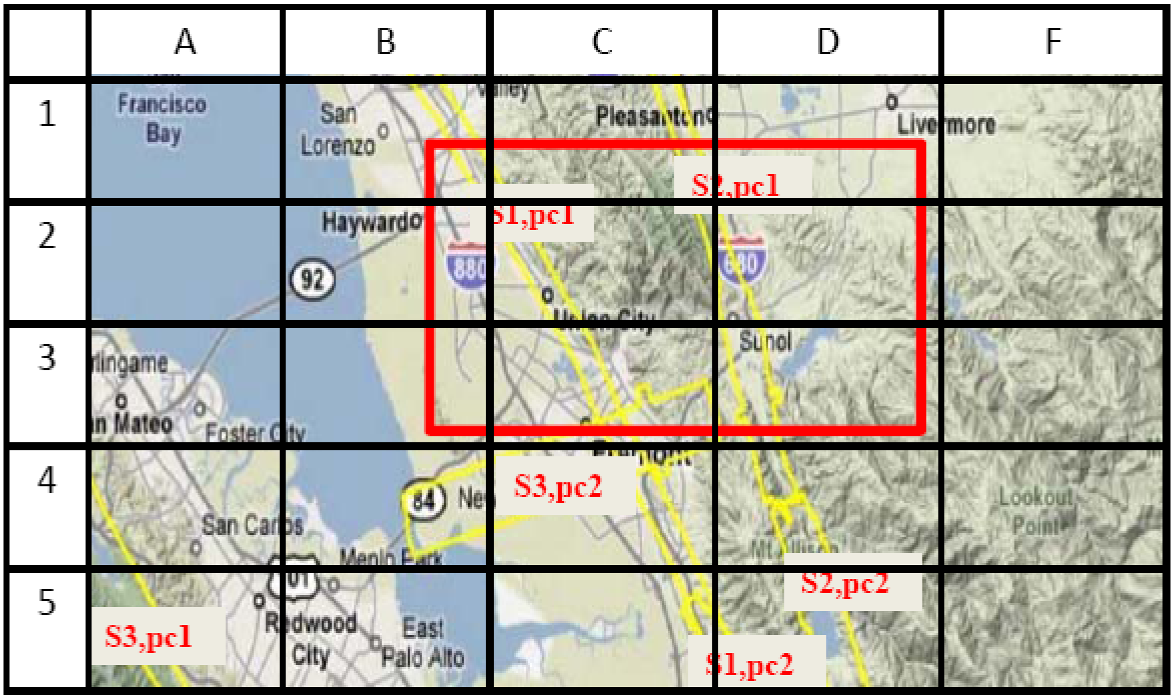
Figure 5. Example 1 for topographic application.

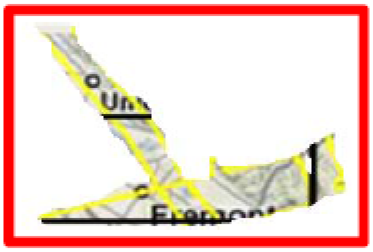
The query for the above example will be of the form :

> *Select sensor = 'Dragon Eye' and flightline = 'pc1' or sensor ='ALTM Gemini' and flightline = 'pc1', point.time = '9-11-09', point.PointCloud, point.X, point.Y, point.Z, block within [(2,C),(2,D),(3,C)(3,D),(4,C),(4,D)] from LIDAR_discovery*

Example 2, Figure 6, illustrates a selection for a region in the red MBR. This could be a particular missions with the sensors set for high accuracy. As an example, this could be to detect feature extraction around airport areas and it is know that mission S2 is not flown around the airport radius because it is not very accurate.

Access Block Index: e.g. 2 < S1 &S3>< S1(pc1) & S3(pc2) >< Red MBR > < 9-11-09 >
<[(1,B),(1,C),(1,D),(2,B),(2,C),(2,D),(3,B),(3,C)(3,D)]>





Select in region from Mission S1& S3 missions
Result – Part of S1(pc1) and S3(pc2)
Note region from Mission S2 were not picked

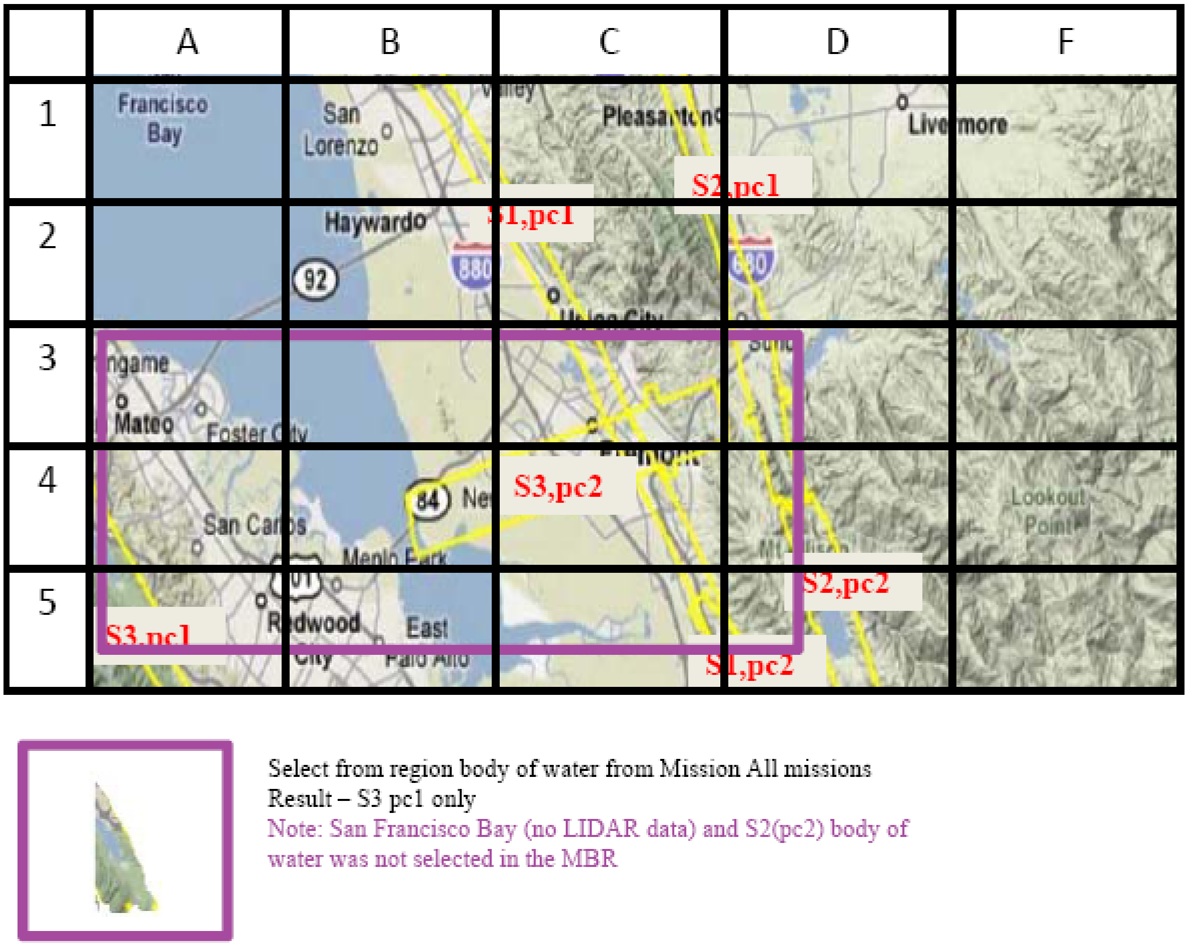
Figure 6. Example 2 for Line of Sight application.

The query for the above example will be of the form :

Select sensor = 'Dragon Eye' and flightline = 'pc1' or sensor ='ALTM Gemini' and flightline = 'pc2' ', point.time = '9-11-09', point.PointCloud, point.X, point.Y, point.Z, block within [(1,B),(1,C),(1,D),(2,B),(2,C),(2,D),(3,B),(3,C)(3,D)] from LIDAR_discovery

Example 3, Figure 7, illustrates a selection for a region in the purple MBR. This could be for the Flood Risk mapping application. LIDAR data does not distinguish bodies of water and the tool, query and analysis should account for that. Also note that, even though there is body of water in S2(pc2) it is not in the MBR and hence it was not selected. Besides those facts the San Francisco Bay does not have LIDAR data and hence should either be out side the calculation or other data should be used as a supplement to correct that void.

Access Block Index: e.g. 3 <All>< S3(pc2) >< Purple MBR >< 9-11-09 >
<[(3,A), (3,B),(3,C),(3,D),(4,A),(4,B),(4,C),(4,D),(5,A),(5,B),(5,C),(5,D)]>



Select from region body of water from Mission All missions
Result – S3 pc1 only
Note: San Francisco Bay (no LIDAR data) and S2(pc2) body of water was not selected in the MBR

Figure 7. Example 2 for Flood Risk mapping application.

The query for the above example will be of the form :
Select from ', point.time = '9-11-09', point.PointCloud, point.X, point.Y, point.Z, block within [(3,A),(3,B),(3,C),(3,D),(4,A),(4,B),(4,C),(4,D),(5,A),(5,B),(5,C), (5,D)] from LIDAR_discovery

SUMMARY

The research investigation to deal with the issues of storing and retrieving the information from multidimensional spatial data sets will be performed, and the result of the three examples above will be recorded in a follow up papers by the author. Problems that may arise depending on LIDAR data only will be addressed for reliable discovery of data in an area of interest. Anticipated problems include:

1. Point cloud data may not cover the entire area of interest therefore data might need to be supplemented for completeness
2. Data obtained may come from different missions, different densities, integrity, classification or accuracy and therefore data might need preprocessing before integration

The work reported in this paper is the first step in a long term project that can help define, manage and optimize database structures for use with the volumes of LIDAR data that are expected in the years to come.

BIBLIOGRAPHY

ASPRS, 2008. LAS Specification Version 1.2., April 28.

ASPRS, 2009. LAS Specification Version 1.3., July 14.

Beaulieu, B.R., 2004, JUL/AUG. Homeland Security through The Palanterra, *GeoIntelligence*, Wasington, D.C.: www.geointelmag.com.

Brown, M.Z., 2009. *LIDAR and Compressive Sensing*, Reston, VA: Work performed under contract to NGA HM1582-0-R-0004.

Carbonell, A., 2009. LIDAR applications, (H. Ferede, Interviewer), August 20.

Cary, T., 2009. Lidar Market: Status and Growth Trends, *International Lidar Mapping Forum.*

Cary, T., 2009. *The Global Market for Airborne Lidar Systems and Services*, Retrieved August 3, 2009, from Cary and Associates: Global Market for Airborne Lidar Systems and Services: http://www.caryandassociates.com/geostore.html

Crosby, C. J., 2006. A Geoinformatics Approach to LiDAR Data Distribution, *Arizona State University, August 2006*, Arizona, US: Master's Thesis.

Schiess, P., & F. Krogstad, 2003. Topography (DEM/DSM generation). Bar Harbor: *Proceedings of the 26th Annual Meeting of the Council on Forest Engineering*.

Schön, B., M. Bertolottoa, D.F. Laefer, & M.W. Seán, 2009. Storage, manipulation, and visualization of lidar data, *3D-ARCH 2009.*

Stennett, T.A., 2004. Lidar: Strap in tight, and prepare to go vertical, *Photogrammetric Engineering & Remote Sensing*, 545-548.

Tracy, Rex, 2009. LIDAR applications, (H. Ferede, Interviewer), August 3.