Cloud Publications
International Journal of Advanced Remote Sensing and GIS
2016, Volume 5, Issue 1, pp. 1500-1506, Article ID Tech-446
ISSN 2320 - 0243

Research Article **Open Access**

Ground and Non-Ground Filtering for Airborne LIDAR Data

Suchita Yadav

National Institute of Technology, Allahabad, Uttar Pradesh, India

Publication Date: 9 February 2016

Article Link: http://technical.cloud-journals.com/index.php/IJARSG/article/view/Tech-446

Abstract Automatic ground filtering for Light Detection and Ranging (LIDAR) data is a critical process for Digital Terrain Model and three-dimensional urban model generation. Various methods have been proposed in literature to separate ground from non-ground, but sometimes problem has been occurred due to the similar characteristics possessed by ground and non-ground objects. The proposed approach in this paper is based on neighborhood based approach. Hierarchy of preprocessing is done for LIDAR data using various essential tools. K-D tree is used to distinguish the bare ground and non-ground objects using nearest neighbor search. Experimental results show the effectiveness of the proposed approach.
Keywords *LiDAR, Airborne LiDAR, K-D tree, Point Clouds*

1. Introduction

LiDAR (Light Detection and Ranging) could be a remote sensing technique supported optical device technology. It measures the period of time of the emitted optical device pulses to work out the space between the device and also the ground. Combined with a world Positioning System (GPS) and a mechanical phenomenon measure Unit (IMU), measuring system will generate a three-dimensional (3D) dense, geo-referenced purpose clouds for the reflective parcel of land surface. Mobile lightweight detection and travel (LiDAR) has bit by bit become a thought tool for aggregation elevation info of surface targets by scheming the time of flight taken for optical device pulse travel between a measuring system device and a target scene. Compared with ancient photogrammetric technique. Measuring system will penetrate through vegetation to get parcel of land elevation in biological science areas. The raw measuring system information contains tremendous points came from numerous objects, like buildings, trees, vehicles etc. measuring system has 2 major blessings over photogrammetric systems: (1) the acquisition of vertical info over an oversized space is additional cost-effective; and (2) there are fewer necessities for information preprocessing.

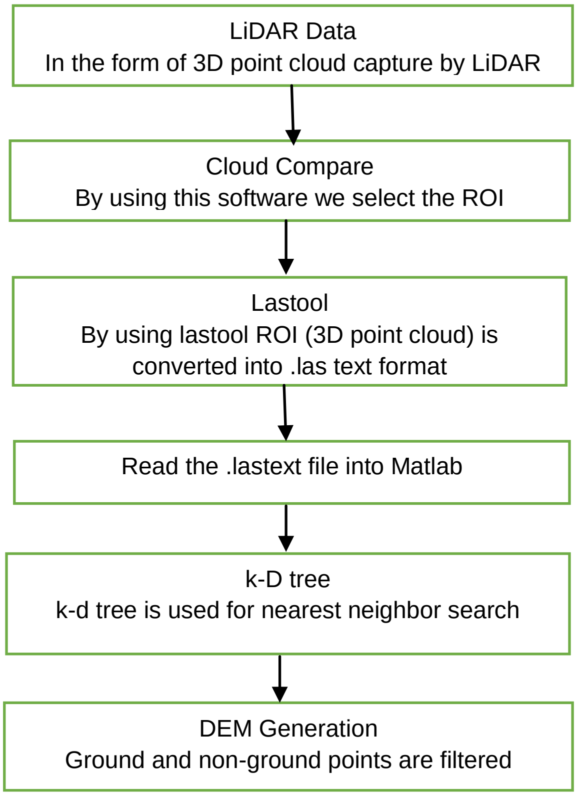
In terms of DTM creation, measuring system has taken the place of ancient photogrammetric ways and become the first technique for manufacturing regional or national DTMs in some countries.

Modern mobile measuring system technology will map the Earth's surface at a 15–20-cm horizontal resolution, and future generations of measuring system scanners are expected to come up with even higher resolution maps. The big volume of scanned information that are manipulated once process a measuring system purpose cloud has been one amongst the main challenges in processing.

In raw measuring system information, each bare-ground and non-ground objects, like trees, buildings, vehicles, and electrical wires, generate disperse. Non-ground points got to be known and eliminated from measuring system measurements before constructing added product like DTMs [3]. Likewise, ground points got to be removed to accurately establish non-ground objects, like buildings and trees. In either case, an economical and correct ground filtering is needed. Existing algorithms have achieved some success, however typically have problem on steep slopes or ridges. To the present finish, our goal is to develop a stronger ground filtering algorithmic program to facilitate DTM creation. Ground filtering algorithms treat either raw measuring system purpose clouds or gridded elevation values [3] that are derived by interpolation of data. However sorting out neighbors in an irregular basis distributed purpose cloud will be long and hard, particularly once users apply the algorithms to broader areas. Most ground filters are supported the idea that natural parcel of land variations ar gradual, instead of abrupt. Therefore, ground elevation and slope ought to vary swimmingly once moving from one ground purpose to a different neighboring ground purpose. In distinction, the boundary between ground and non-ground points ought to exhibit an abrupt amendment in elevation and slope. The joint use of slopes, elevation variations, and native elevations will discriminate ground points from non-ground points [4].

2. Proposed Approach

Proposed approach is briefly described in following chart.



Step used in proposed approach is described following:

Step 1

First we take the raw LiDAR input data which is captured by airborne LiDAR. The input is in the form of 3D point cloud. These point clouds consists the points of both ground and non-ground objects.

Step 2

We select ROI (region of interest) from taken input with the help of Cloud Compare software. Selected ROI must consist both ground and non-ground point, which is helpful for further processing.

Step 3

We have to convert the ROI which is in 3D point cloud, into text format so that we can easily read the input data by using Matlab and can process our algorithm on that data.

Step 4

Now we use k-D tree data structure which is useful for nearest neighbor search. k-D tree is the type of multidimensional BST. A k dimensional tree is a space partitioning data structure in a k-dimensional space.

2.1. Nearest Neighbor Search

The nearest neighbor search formula aims to seek out purpose within the tree that's nearest to a given input point.

1) Starting with the foundation node the formula moves down the tree recursively, within the same manner that it might if the search purpose were being inserted (goes left or right, depends).

2) Once the formula reaches a leaf node, it that leaf node purpose because the "current best".

3) The formula unwinds the rule of the tree, activity the subsequent steps at every node:

    a. If the present node is nearer than the present best, then it becomes the present best.
    b. The formula checks whether or not there may well be any purposes on the opposite aspect of the ripping plane that are nearer to the search point than the present best. In concept, this is often done by crossed the ripping hyper plane with a hyper sphere round the search purpose that encompasses a radius up to the present nearest distance. Since the hyper planes are all axis-aligned this is often enforced as a straightforward comparison to check whether or not the distinction between the ripping coordinate of the search purpose and current node is lesser than the space (overall coordinates) from the search purpose to the present best. (1) If the hyper sphere crosses the plane, there may well be nearer points on the opposite aspect of the plane, therefore the formula should move down the opposite branch of the tree from the present node probing for nearer points, following identical algorithmic method because the entire search. (2) If the hyper sphere does not cross the ripping plane, then the formula continues walking up the tree, and also the entire branch on the opposite aspect of that node is eliminated.

4) 4- Once the formula finishes this method for the foundation node, then the search is complete.

Range search: a spread search searches for ranges of parameters as an example, if a tree is storing values akin to financial gain and age, then a spread search can be one thing like probing for all members of the tree that have associate age between twenty associated fifty years and an financial gain between 50,000 and 80,000. Since k-d trees divide the vary of a site in at every level of the tree, they're helpful for activity vary searches.

2.2. Algorithm

Input: 3 Dimensional Coordinates of all 3d points
Output: Bare Ground with removed object
Assumption: (1) ***x, y, z, β and μ*** are NULL vectors.

(2) ***R*** is radius that is 0.3.
(3) ***n*** is number of 3d points.
(4) ***α*** is object created by K-D Tree Algorithm.

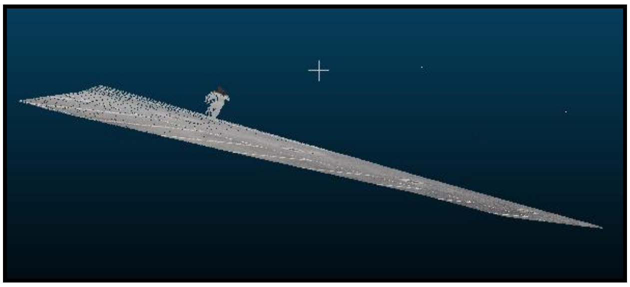
2.3. Procedure

OBJECT REMOVAL (All Coordinates)

x ← X coordinates of all 3d points
y ← Y coordinates of all 3d points
z ← Z coordinates of all 3d points
α ← **KD Tree**(x,y,)
For i ← 1 to *n*
β← **RangeSearch**(***α, α (i),R***)
s← z(*β*(1,1))
μ← Standard Deviation(*s* (1,1))
If *μ* ≤ *R1*
x (i)← *x(i)*
y(i) ← *y(i)*
z(i) ← *z(i)*
End If
End For
End Procedure

3. Experimental Result

3.1. Input Data



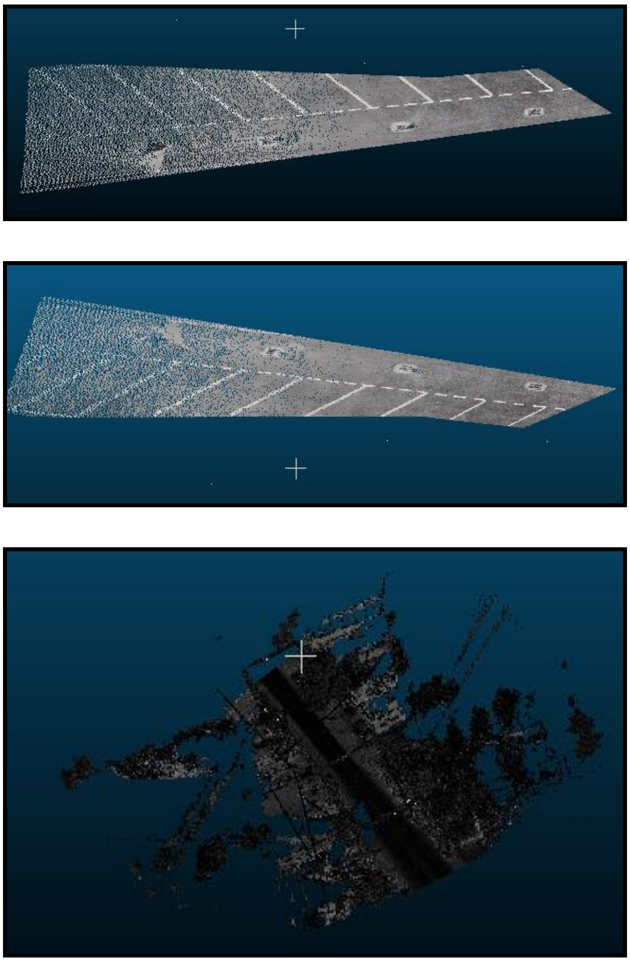


Figure 1: *3D Data Point Cloud Captured by Airborne LiDAR*

3.2. Output

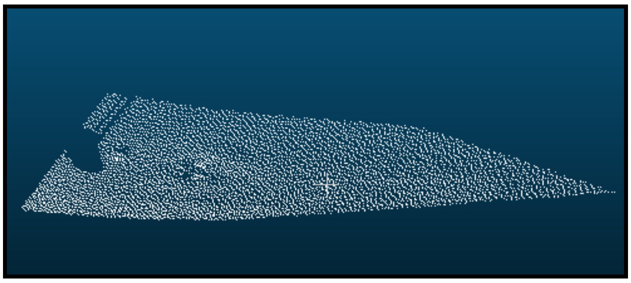


Figure 2: *3D Data point Cloud Output*

4. Dataset and Accuracy Measure

The dataset was sampled by ALTM scanner, and both the first and last return was available. For the study site, the ground point are already identified and available for assessing accuracy. We use the ground reference data to optimize our selection of slope and elevation difference thresholds and test the sensitivity of our algorithms to changes in thresholds.

Table 1: *Kappa Coefficient for Various Algorithms*

Sample Data	Kraus and Pfeifer	Sithole and Vosselman	Zhang and Whitman	Jie Shan and Sampath	Meng et al.	Ground and Non-Ground Filtering
Data1	65.05	51.24	80.01	78.48	91.24	90.63
Data2	90.10	78.86	56.78	81.97	60.13	89.14
Data3	91.01	90.44	92.51	74.71	54.89	67.73
Data4	47.09	87.61	86.11	36.29	89.17	98.7
Data5	75.26	41.21	61.21	49.00	90.05	88.03
Data6	66.75	19.81	46.31	54.21	31.22	41.19

5. Conclusion

The proposed approach is based on neighborhood of the cloud points. Hierarchy of preprocessing is done for LIDAR data using various essential tools. ROI is selected and cropped by a specific tool. Coordinates of all three dimensions of cloud points are processed by nearest neighborhood processing algorithm. K-D tree is used to distinguish the bare ground and non-ground objects using nearest neighbor search. Experiments have been performed on various LIDAR data sets. Kappa coefficient is used to compare the accuracy and effectiveness of the proposed approach. Experimental results show the effectiveness and usefulness of the proposed approach.

References

[1] Kraus and Pfeifer. *Determination of Terrain Models in Wooded Areas with Airborne Laser Scanner Data.* ISPRS Journal of Photogrammetry & Remote Sensing. 1998. 53; 193-203.

[2] Keqi Zhang et al. *A Progressive Morphological Filter for Removing Non-Ground Measurements from Airborne LIDAR Data.* IEEE Transactions on Geoscience and Remote Sensing. April 2003. 41 (4).

[3] Sithole and Vosselman*. Experimental Comparison of Filter Algorithms for Bare-Earth Extraction from Airborne Laser Scanning Point Clouds.* ISPRS Journal of Photogrammetry & Remote Sensing. 2004. 59; 85-101.

[4] Zhang and Whitman. *Comparison of Three Algorithms for Filtering Airborne LiDAR Data.* Photogrammetric Engineering & Remote Sensing. March 2005. 71 (3) 313-324.

[5] Jie Shan and Aparajithan Sampath. *Urban DEM Generation from Raw LiDAR Data: A Labeling Algorithm and Its Performance.* Photogrammetric Engineering & Remote Sensing. February 2005. 71 (2) 217-226.

[6] Xulien Meng et al. *A Multi-Directional Ground Filtering Algorithm for Airborne LIDAR.* ISPRS Journal of Photogrammetry and Remote Sensing. 2009. 64; 117-124.

[7] Xulien Meng et al. *Ground Filtering Algorithms for Airborne LiDAR Data: A Review of Critical Issues.* Remote Sensing. 2010. *2*; 833-860.

[8] Dehvari and Heck. *Removing Non-Ground Points from Automated Photo-Based DEM and Evaluation of Its Accuracy with LiDAR DEM.* Computers & Geosciences. 2012. 43; 108-117.

[9] Chen and Li. *A Robust Method of Thin Plate Spline and Its Application to DEM Construction.* Computers & Geosciences. 2012. 48; 9-16.

[10] Chen, Z., Devereux, B., Gao, B. and Amable, G. *Upward-Fusion Urban DTM Generating Method Using Airborne LiDAR Data.* ISPRS Journal of Photogrammetry and Remote Sensing. 2012. 72; 121-130.

[11] Chen et al. *A Multiresolution Hierarchical Classification Algorithm for Filtering Airborne Lidar Data.* ISPRS Journal of Photogrammetry and Remote Sensing. 2013. 82; 1-9.