

EXTRACTION OF GROUND SURFACE ALONG ROADWAY FROM MOBILE LIDAR DATA

M. Yadav ^{1,*}, A. K. Singh ^{2,1}, B. Lohani ³

¹ GIS Cell, Motilal Nehru National Institute of Technology Allahabad, India - ssm Yadav@mnnit.ac.in

² Dept. of Civil Engineering, Motilal Nehru National Institute of Technology Allahabad, India - aksingh@mnnit.ac.in

³ Dept. of Civil Engineering, Indian Institute of Technology Kanpur, India - blohani@iitk.ac.in

Commission V, WG V/7

KEYWORDS: Mobile LiDAR System (MLS), Point Cloud, Ground Filtering, Grids, Standard Deviation, Flatness

ABSTRACT:

High quality digital elevation model (DEM) is obtained from mobile LiDAR data and it is used in various applications like road widening, slope measurement of road side surfaces, and inundation of the roadway evaluation. Two steps algorithm is proposed to filter ground points using mobile LiDAR data. Initially unstructured input data is organized then standard deviation and flatness based approach is used to filter ground points. Proposed algorithm is tested on point cloud of test site located along 800 m of roadway. Type I, Type II and total error are 2.11%, 2.21% and 2.15%, respectively with kappa is equal to 96.61% are computed using ground filtered points and reference data points.

1. INTRODUCTION

Integrated LiDAR system deployed on different platforms, i.e. airborne, terrestrial or mobile has been used for acquiring detailed and accurate geometric (XYZ) and radiometric (intensity) information of earth objects since last few decades. It becomes most efficient surveying technique in recent years in terms of its speed of data acquisition, easy to operate and minimal manual intervention. Size of acquired data is large due to high point density, which is highest in case of terrestrial laser scanning. Mobile LiDAR system (MLS) operates in shorter range compare to airborne LiDAR system, so it maps roadway objects, i.e., trees, building, roadway ground surface, pole-like objects, cars, pedestrians, and power line corridors more accurately. MLS collects data at road speeds and it eliminates traditional surveyor safety issues, keeps roads/lanes open, and minimize impact to traffic (Yadav et al., 2014). Analysis of road infrastructure, bridge structure and overhead clearances, road surface conditions assessment, roadway asset management, finding location of encroaching overhead wires, and detection of pole-shaped objects are important applications of MLS. MLS data is volumetric in size and requires automatic algorithms for segmenting and extracting various objects of interest present in data. But automation is big challenge due to volumetric data size, variable point densities, complicated scene structures and occlusion of features by moving objects. MLS data can be generally classified into ground and non-ground points. Ground filtering of MLS data reduces data volume and allows further processing of ground and non-ground features easy and straightforward.

High quality digital elevation model (DEM) is generated along roadway using MLS data and it is used during road widening, slope measurement of road side surfaces, and inundation of the roadway evaluation.

Existing studies on ground filtering using MLS data are divided into two categories, one focuses solely on extraction of ground points and other focuses on classifying other objects but first step is ground points filtering. Lalonde et al. (2006) filtered ground points from non-ground points. Daniel et al. (2010) identified ground points for operating unmanned ground vehicles. two dimensional (2D) regular grids were generated and lowest height point in each grid was retained. Pu et al. (2011) separated ground points by assuming that ground points lie below the trajectory of laser scanner. Filtered ground points were further segmented into on ground and off ground. They used trajectory information and performed 2D segmentation to filter the ground points. Liu et al. (2013) and Tian et al. (2014) proposed similar three step model for ground extraction. Three dimensional grids were generated using vehicle trajectory, point density and slope. Instead of choosing lowest height point in a grid, collinear condition in horizontal points by assuming that the ground is less steep than a predefined slope. Konolige et al. (2009) used RANSAC algorithm to separate plane was used for seed point estimation. Then a multi-scale neighborhood analysis was performed followed by slope analysis to filter final ground points.

The current researches on mobile or terrestrial point cloud data focus on independent object extraction (Liu et al., 2013; Tian et al., 2014). Yu et al. (2014) used block based

* Corresponding author

elevation method for ground filtering. Ground points were filtered for detecting trees. Zhou et al. (2014) proposed scan line based method for ground filtering using slope and elevation information.

Two steps algorithm is proposed to filter ground points. Initially unstructured input data is organized then standard deviation and flatness based approach is used to filter ground points. Proposed method works directly on unstructured MLS data available in XYZ format, so it is independent on scanning geometry. No any additional information, i.e., trajectory information and training data are required unlike many methods of literature. Proposed algorithm is simple and it is tested efficiently on point cloud of test site located along 800 m of roadway. These exclusive features of proposed algorithm prove its efficiency and uniqueness for ground filtering using mobile LiDAR data.

2. METHODOLOGY

2.1 Data points organization

MTLS data points, $P_i (i = 1, 2, 3, 4, \dots, r)$ in XYZI format are used, where i and r represent arbitrary data point and total number of data points sequentially. Data points are further arranged in a specific pattern to reduce computation time of proposed algorithm for ground filtering. Three sequential steps are applied for organizing the data points: (1) projecting data points on 2D plane, i.e. XY plane; (2) projected data points division into square grids; and (3) vertical segmentation of data points of each grids and separate tagging of each segment data points as blocks. Each step is discussed in detail in the following sections.

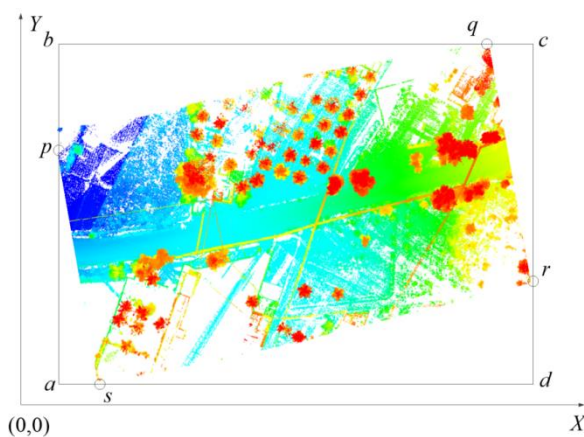


Figure 1. Minimum bounding rectangle of input MLS data

2.1.1 Data points projection

MTLS data points are projected on XY plane and new set of projected data points, P_i^{2D} in XY format only is generated. Minimum bounding rectangle (MBR), say $abcd$ (see Figure 1) is generated using data points of set P_i^{2D} . It is characterized by the 2D coordinates of four vertices, i.e., a , b , c , and d . Initially maximum and minimum values of abscissa (X) and ordinate (Y) from set P_i^{2D} are determined.

The point p has its abscissa minimum (X_{\min}) and similarly point r has maximum (X_{\max}) among points of set P_i^{2D} (see Figure 1). Ordinate of point s is minimum (Y_{\min}) and similarly for point q is maximum (Y_{\max}) among points of set P_i^{2D} (see Figure 1). Using these minimum and maximum values of abscissa and ordinate, the vertices a , b , c and d are determined. The points $a(X_{\min}, Y_{\min})$, $b(X_{\min}, Y_{\max})$, $c(X_{\max}, Y_{\max})$ and $d(X_{\max}, Y_{\min})$ are four vertices of MBR, say $abcd$.

2.1.2 Square gridding

Set of 2D MTLS data points, P_i^{2D} which defines MBR, say $abcd$ are divided into $N (N_x \times N_y)$ regular square grids of size $m \times m$ (see Figure 2). N_x and N_y are calculated using Equation 1.

$$N_x = \begin{cases} [d(a,d)/m] + 1 & \text{if } d(a,d)/m \text{ is non-integer} \\ d(a,d)/m & \text{if } d(a,d)/m \text{ is integer} \end{cases} \quad (1)$$

Where $d(a, d)$ is Euclidean distance between points a and d . $[]$ is symbol for greatest integer function. Similarly N_y is determined using $d(a, b)$.

The following points e , f , g , and h are four vertices of square grid, $N_{l,k}$.

- I. $e(X_{\min} + (l-1) \times m, Y_{\min} + (k-1) \times m)$
- II. $f(X_{\min} + (l-1) \times m, Y_{\min} + k \times m)$
- III. $g(X_{\min} + l \times m, Y_{\min} + k \times m)$
- IV. $h(X_{\min} + l \times m, Y_{\min} + (k-1) \times m)$

$N_{l,k}$ is an arbitrary square grid and four vertices of all the N grids are computed by using different combinations of $l = 1$ to N_x and $k = 1$ to N_y . A new set of points $P_i^{l,k}$ is

generated, which lies inside the square grid $N_{l,k}$. Grids take shape of connected pillars in 3D.

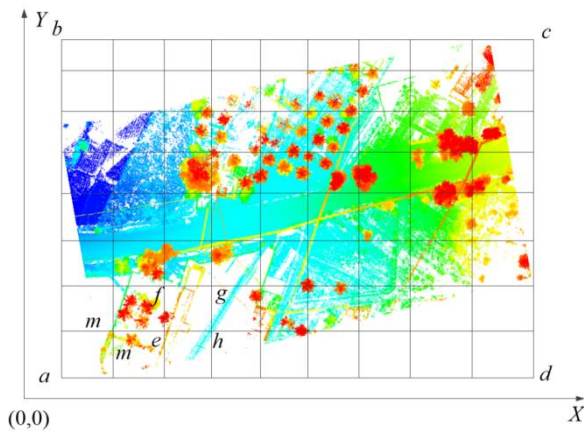


Figure 2. 2D gridding of input MLS data

2.1.3 Vertical segmentation

In this step, MTLs data points $P_i^{l,k}$ ($i = 1, 2, 3, 4, \dots, r^{l,k}$) of arbitrary grid $N_{l,k}$, say $efgh$ are selected (see Figure 2). Further $P_i^{l,k}$ is arranged row-wise in the fashion of increasing of Z values and new set $P_{i,z}^{l,k}$ is generated. Minimum Z (Z_{\min}) and maximum Z (Z_{\max}) are computed by taking average of n_{\min} and n_{\max} number of data points from starting and end of data set $P_{i,z}^{l,k}$. It is done to avoid the effect of outlier.

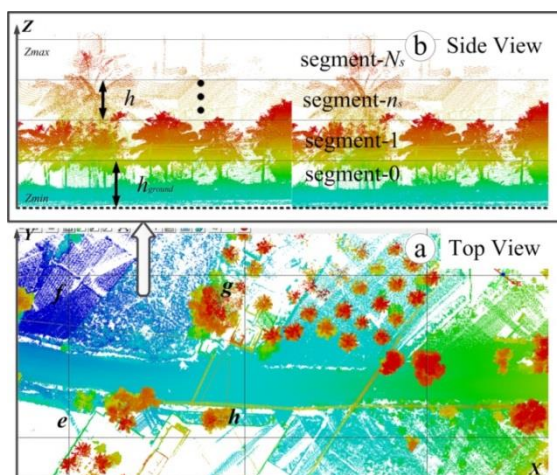


Figure 3. Vertical segmentation of 2D grid

Data set, $P_{i,z}^{l,k}$ is divided into different segment based on the grouping of data points lies in the definite range of Z

values. It is assumed that data points lies within Z_{\min} and $Z_{\min} + h_{\text{ground}}$ contain all the ground points and grouped together as set segment-0 (see Figure 3). The h_{ground} is range of Z values of ground points. Similarly segment- n_s is computed by grouping the data points lies within $Z_{\text{lower}}(n_s)$ and $Z_{\text{upper}}(n_s)$ (Equation 2 & 3).

$$Z_{\text{lower}}(n_s) = Z_{\min} + h_{\text{ground}} + (n_s - 1)h \quad (2)$$

$$Z_{\text{upper}}(n_s) = Z_{\min} + h_{\text{ground}} + n_s h \quad (3)$$

Where, h is the fixed range of Z values for segment- n_s . The n_s vary from 1 to N_s and N_s is total number of segment calculated by Equation 4.

$$N_s = \begin{cases} \lceil M \rceil + 1 & \text{if } M \text{ is non-integer} \\ M & \text{if } M \text{ is integer} \end{cases} \quad (4)$$

Where $M = (Z_{\max} - Z_{\min} - h_{\text{ground}}) / h$.

Vertical segmentation is only performed on different combinations of grid $N_{l,k}$ by considering different values of l and k for which the total number of points $r^{l,k}$ is greater than a user defined threshold value, $r_{th}^{l,k}$.

2.2 Ground points filtering

Rough ground classification is performed by filtering data points of segment-0 of each vertical pillar, which lies within minimum Z and minimum $Z + h_{\text{ground}}$ (see Figure 3). These filtered points are from first segment of vertical pillar having height h_{ground} , and vertical objects such as tress, road signs, light poles, buildings along the route corridor are removed, so it is called ground segment also. But points from base of vertical objects and low vegetation are still present in the filtered data. Each point of ground segment is selected as seed point and its neighbouring points within radius R are computed using 2D k-d tree neighborhood search technique (Yadav et al., 2015). Standard deviation (Z_{std}) of Z values and flatness (see Equation 5) of set of neighbouring points of each seed point are computed.

$$\text{Flatness: } (F) = \alpha_1 / (\alpha_1 + \alpha_2 + \alpha_3) \quad (5)$$

Where α_i for $i=1, 2,$ and 3 are three Eigen values and α_1 is minimum Eigen value of points lie within cylinder. If Z_{std} is less then user defined threshold $Z_{\text{std}}^{\text{ground}}$ and flatness is also less than flatness threshold (F^{th}), then seed point is classified as ground point otherwise non-ground point. The above process is iterated for all the points of

ground segment by assuming them as seed point. Finally ground segment is further classified into ground point and non-ground points.

3. Experiment

3.1 Mapping and Reference Data

Test site (see Figure 4) used for testing proposed method is located along two lane road of 800 m. Road environment of this site is quite complex, having road-side irregularly spaced trees, bushes, low vegetation, tilted and vertical utility poles, buildings, boundary walls, tall telephone tower, overhead water tank, and vertical pillars supporting rooftop of a fuel filling station, overhead power lines and high mast signage pole installed at fuel filling station (see Figure 5(b)). Terrain of the site is not flat as the maximum height difference on the ground surface was 10 m. The 46,192,791 points are acquired by StreetMapper 360 MLS from the test site at 2D point density of 450 points/m². Reference ground points are generated by visual inspection of colored point cloud data of test site and manual filtering of ground point using editing tool of TerraScan module of Terrasolid software. Manually filtered ground points are used for validating ground filtering result. 28,649,142 ground points are manually filtered and used as reference.

3.2 Result

Proposed algorithm is tested on point cloud of test site (see Figure 4) located along 800 m of roadway using thresholds; $m \times m = 3 \text{ m} \times 3 \text{ m}$, $h_{\text{ground}} = 1 \text{ m}$, $R = 20 \text{ cm}$, $Z_{\text{std}}^{\text{ground}} = 5 \text{ cm}$, and $F^{\text{th}} = 0.15$. Ground points (28,455,343) are efficiently filtered out (see Figure 5 (a) & Table 1).

3.3 Discussion

The result obtained (see Table 1) shows a very satisfactory performance of the proposed algorithm, i.e., Type I (see Equation 6), Type II (see Equation 7) and total error (see Equation 8) are 2.11%, 2.21% and 2.15%, respectively with kappa (see Equation 9) is equal to 96.61%. Type I and Type II errors are ground and non-ground points omission error, respectively. The proposed algorithm uses only XYZ coordinates of mobile LiDAR point cloud. It is also independent on the scanning geometry and neighborhood structure in the mobile LiDAR data file. Proposed method does not use training data.

		Filtered	
		Ground points	Non-ground points
Reference	Ground points	a	b
		28,045,218	603,924
	Non-ground points	c	d
		410,125	18,147,573

Table 1. Ground filtering performance of proposed algorithm on chosen test site.

$$\text{Type I error (\%)} = (b / (a + b)) \times 100 \quad (6)$$

$$\text{Type II error (\%)} = (c / (c + d)) \times 100 \quad (7)$$

$$\text{Total error (\%)} = ((b + c) / (a + b + c + d)) \times 100 \quad (8)$$

$$\text{Kappa (\%)} = ((e \times (a + d) - p) / (e^2 - p)) \times 100 \quad (9)$$

Where $e = a + b + c + d$ and $p = ((a + b) \times (a + c)) + ((c + d) \times (b + d))$

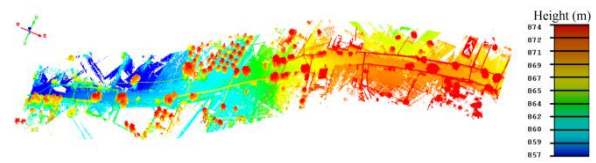


Figure 4. 3D perceptive point cloud view test site

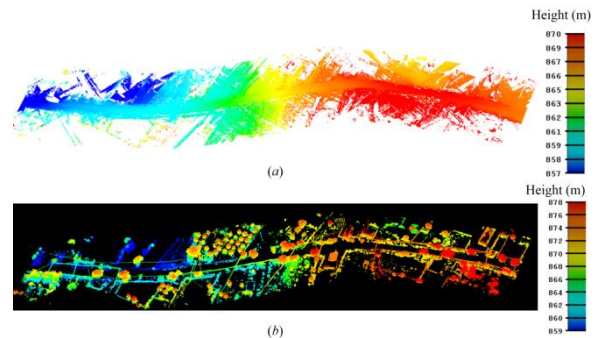


Figure 5. 3D perceptive point cloud view of (a) ground points (b) non-ground points, of test site

4. Conclusions and Future Works

In this study, an automatic algorithm is proposed for ground point filtering from MLS data. Initially, input MLS data are divided into connected pillars followed by filtering of ground segment. Ground points are extracted from ground segment based on the criteria on threshold of standard deviation and flatness computed on neighbors around seed points. Proposed algorithm is tested on MLS data of test site located along 800 m of roadway. Type I, Type II and total error are 2.11%, 2.21% and 2.15%, respectively with kappa is equal to 96.61% are computed using ground filtered points and reference data points.

Using the results obtained in the test site, it can be concluded that the proposed method is more general and it performs efficiently: (i) even in case of complex roadway scene (ii) requiring no training data and additional information, i.e., trajectory data, respectively, (iii) no initial assumptions about the relative location of roadway objects (iv) being independent of the scanning geometry information and only need mobile LiDAR points in XYZ format.

Future works are: (i) to improve the accuracy of ground filtering (ii) modifying proposed algorithm to make it fully automatic and parameter free.

ACKNOWLEDGMENT

The authors would like to thank Geokno India Pvt. Ltd. for providing the StreetMapper 360 mobile LiDAR system and field assistance for capturing the point cloud data used in this study.

REFERENCES

- Daniel, M. W., Nishihata, T., Brooks, C. A., Iagnemma, K., 2010. Ground plane identification using LiDAR in forested environments. IEEE International Conference on Robotics and Automation, Alaska, USA, pp 3831-3836.
- Konolige, K., Agrawal, M., Blas, M. R., Bolles, R. C., Gerkey, B., Sola, J., Sundaresan, A., 2009. Mapping, navigation and learning for Off-road traversal. *Journal of Field Robotics*, 26(1), pp. 88-113.
- Lalonde, J. F., Vandapel, N., Huber, D. F., Hebert, M., 2006. Natural terrain classification using three-dimensional Ladar data for ground robot mobility. *Journal of Field Robotics*, 23(10), pp. 839 – 861.
- Liu, R., Tian, M., Shi, B., Xu, J., 2013. Expressway surface point extraction from mobile laser scanning point clouds. *Journal of Engineering Science and Technology Review*, pp. 15-19.
- Pirotti, F., Guarnieri, A., Vettore, A., 2013. Vegetation filtering of waveform terrestrial laser scanner data for DTM production, *Applied Geomatics*, 5(4), pp. 311–322.
- Pu, S., Rutzinger, M., Vosselman, G., Elbernik, S. O., 2011. Recognizing basic structures from mobile laser scanning data for road inventory studies. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66 (6), pp. 28-39.
- Tian, M., Liu, R., Lu, X., 2014. Ground point filtering method of vehicle-borne laser point cloud in urban street. *Computer Modelling and New technologies*, 18(3), pp. 188-192.
- Vaaja, M., Kukko, A., Kaartinen, H., Kurkela, M., Kasvi, E., Flener, C., Hyypa, H., Hyypa, J., Jarvela, J., Alho, P., 2013. Data processing and quality evaluation of a boat-based mobile laser scanning system. *Sensors*, 13(9), pp. 12497-12515.
- Yadav, M., Goel, S., Singh, A. K., Lohani, B., 2014. Developing Basic Design and Mathematical Framework for a Mobile Mapping System—a Case Study Using Available Sensors, *J. Indian Soc. Remote Sens.*, 42(2), pp. 301–310.
- Yadav, M., Husain, A., Singh, A. K., & Lohani, B., 2015. Pole-shaped object detection using mobile LiDAR data in rural road environments. In: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. II-3/W5, ISPRS Geospatial Week, La Grande Motte, France, pp. 11-16.
- Yu, Y., Li, J., Guan, H., Zai, D., Wang, C., 2014. Automated extraction of 3D trees from mobile lidar point clouds. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XL-5, ISPRS Technical Commission V Symposium, Riva del Garda, Italy, pp. 629-632.
- Zhou, Y., Wang, D., Xie, X., Ren, Y., Li, G., Deng, Y., Wang, Z., 2014. A fast and accurate segmentation method for ordered LiDAR point cloud of large-scale scenes. *IEEE Geoscience and Remote Sensing Letters*, 11(11), pp. 1981-1985