www.cafetinnova.org

ISSN 0974-5904, Volume 10, No. 03

DOI:10.21276/ijee.2017.10.0329

International Journal of Earth Sciences and Engineering

June 2017, P.P. 683-687

An Automated Approach for Contour Lines Generation using Terrestrial Laser Scanner Data

ARSHAD HUSAIN[1], RAKESH CHANDRA VAISHYA[2] AND SHAILENDRA CHAUDHARY[2]

[1]GIS Cell, MNNIT, Allahabad 211004, U.P., INDIA

[2]Department of Civil Engineering, MNNIT, Allahabad 211004, U.P., INDIA

Email: *rgi1501@mnnit.ac.in, rcvaishya@mnnit.ac.in, rce1504@mnnit.ac.in*

Abstract: *Contouring is commonly used method for terrain mapping. Contour lines illustrate the profile (topography) of the land surface. In the process of contour lines generation, elevation data of terrain is mandatory. Light Detection and Ranging (LiDAR) or laser scanning technology is acknowledged as a remote sensing technique, to generate massive amounts of densely spaced geo-referenced three dimensional points that sample the elevation of a terrain. In present study, an automated method for generating the contour lines using Terrestrial Laser Scanner (TLS) data has been proposed. Proposed method consist four steps, first step divides the TLS data into regular two dimensional square grids. Generation of depth image is performed in second step of proposed method. In third step, pixels having same depth are identified at pre-defined contour interval. Contour lines are generated in forth step, by joining the same depth pixels with the help of convex-hull.*

Keywords: *Lidar, Terrestrial Laser Scanner, Contour Lines, Depth Image*

1. Introduction

Typically, contour lines are used in order to demonstrate the profile of the terrain (topography) on a map. Contour lines are useful in various areas such as determination of catchment area, finding water flow direction, locating bunds, dams and to find out flood levels. Also, the contours lines are help full for deciding the routes of the railway, road, canal or sewer lines so as to minimize and balance earthworks. A map illustrated with contour lines is called contour map, a topographic map is an example of contour map. Contour lines are created by connecting the points of equal elevation on contour map. Contour lines are equally separated with each other by distance on contour maps called contour interval, represents the difference in elevation between two successive contour lines. Base contour is the position from which contouring begins. The contour interval is always remains constant for each map. On the basis topography (terrain nature) of the mapped area, the contour interval is chosen for a map. The contour interval is chosen usually larger in high relief mapped areas, to prevent the contour map from too many contour lines. Otherwise, it would make the contour map difficult to read [1]. Typically, every fifth contour line of contour map is marked as an index contour line, which makes the map easier to read [2]. Labels of elevation map are also a vital element. The reader can quickly interpret the shape of the terrain, if the contour map is properly labeled. The terrain is steep, if the labeled numbers are placed close to each other. Contours lines deal with the elevation of terrain. Therefore, for exact representation of shape of terrain, accurate height (elevation) information is mandatory. The beginning of the laser scanning or Light Detection and Ranging (LiDAR) surveying was a technological revolution in procuring three-dimensional (3D) data. LiDAR has become a reliable, swift and precise technique for collection of topographic data about surface objects in a rapid and cost-effective manner. LiDAR provides accurate elevation information of the top surface and ground terrain. LiDAR technology incomes less human and atmospheric interaction as compare to conventional surveying and mapping systems including photogrammetry and total station surveying [3,4]. LiDAR is capable of capturing the range of target from the source device and scan angle. Further, these two parameters range and scan angle are converted in the X, Y and Z coordinate values with the help of mathematical transformation called geo-location. GPS and Inertial Navigation System (INS) are used for the transformation from the local coordinate values to global coordinate values. Therefore, LiDAR generates the 3D geo-referenced point cloud [5]. LiDAR is also capable of sensing some other essential information of the target such as intensity, number of returns, return numbers, etc.

Point cloud generated by the laser scanner is further used to generate other geospatial information such as Digital Surface Models (DSMs) and Digital Terrain Models (DTMs). These generated models ensure higher accuracy and higher resolution as compare to the traditional model generation methods. DTM is very expensive element for topographic maps and it is used to generate contour lines. DTM generated by aerial image is quite slow due to the manual workflow while, laser scanner is powerful enough for quick generation of DTM. Many algorithms were explored by the researchers to process LiDAR point cloud and filter ground points to build DTMs and DSMs.

Received: January 11, 2017; Accepted: May 29, 2017; Published: June 30, 2017
International Journal of Earth Sciences and Engineering, 10(03), 683-687, 2017, DOI:10.21276/ijee.2017.10.0329

Vosselman [9] proposed an algorithm based on mathematical morphology to filter the ground and non-ground points from the scanned point cloud but the proposed algorithm was unable to handle objects like large building or dense forest (Chang et. al [8]). Axelsson [10] introduced a novel approach in order to generate the Digital Elevation Model (DEM) based on progressive densification of a Triangular Irregular Network (TIN). A linear prediction and hierarchic robust interpolation method was proposed by Kraus and Pfeifer [11] to generate the DTM using LiDAR data. Jacobsen and Lohmann [7] developed a classification method to classify the segments into ground or non-ground points. Developed method uses the concept of height differences to classify the objects. In this research, a novel method for contour lines generation using TLS data is proposed. Captured TLS data has been divided into regular two dimensional square grids. Further, a depth image is generated by using the gridded data. Equal depth pixels of generated depth image are identified and joining of these pixels is performed with the help of convex-hull.

2. Proposed Method

Proposed method for generating contours lines using TLS data is divided into four steps (Fig. 1). The steps are namely gridding of data, depth image generation, finding pixels having same depth and labeling and joining of same depth pixels. All these steps are discussed in detail in the following sections.

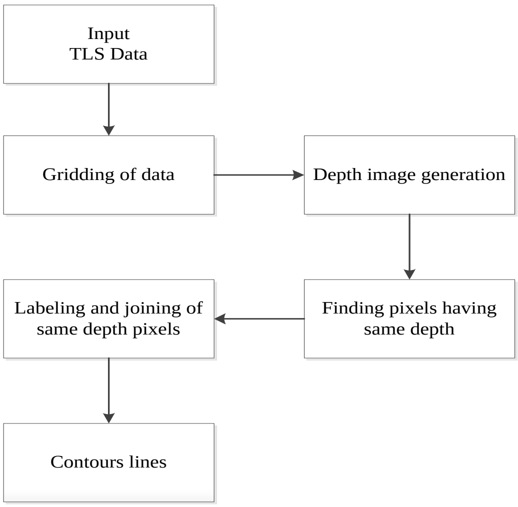
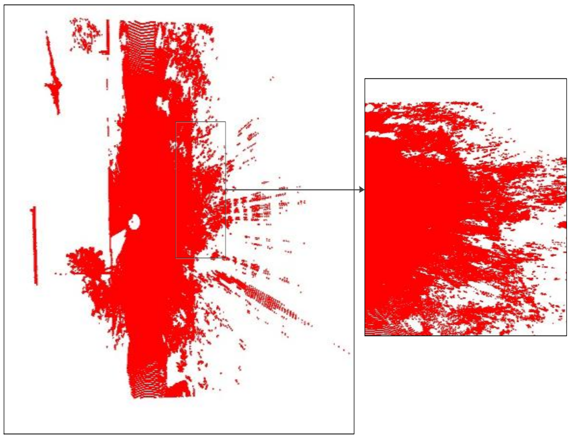


Fig. 1 *Flow chart of proposed method*

2.1 Gridding of Data

X, Y, and Z coordinates of each TLS point is taken as input. Further, two dimensional projection has been performed at input dataset. For projecting the input dataset, Z coordinate value of each TLS point is ignored (set to zero). Now, each point has only X and Y coordinate values. XY plane is chosen as projection plane. The objective of two dimensional (2D) projection is, to create a minimum bounding box at input dataset. The amount of distortion in area of input dataset is very less in case of XY plane projection, therefore, XY plane is chosen as 2D projection plane. Fig. 2(a) shows the two dimensionally projected dataset and Fig. 2(b) shows the closer view of projected dataset.



(a) ***(b)***

Fig. 2 (a) *2D projected dataset* ***(b)*** *closer view of 2D projected dataset*

After projecting the dataset, points with minimum X, minimum Y, maximum X and maximum Y coordinates (four points) are identified. Points *a*, *b*, *c* and *d* in Fig. 3 shows the identified points having minimum X, maximum X, maximum Y and minimum Y coordinate values respectively. Bottom left and top right corners of bounding box are defined with the help of these four identified points. Coordinate of bottom left and top right corners of bounding box are (minimum X, minimum Y) and (maximum X, maximum Y) respectively. In Fig. 3, point *p* and point *r* show the bottom left and top right corner of bounding box. Likewise, other two corners, bottom right and top left are defined. The coordinate of these two points are (maximum X, minimum Y) and (minimum X, maximum Y) respectively. Point *q* and point *s* in Fig.3 represent these corners.

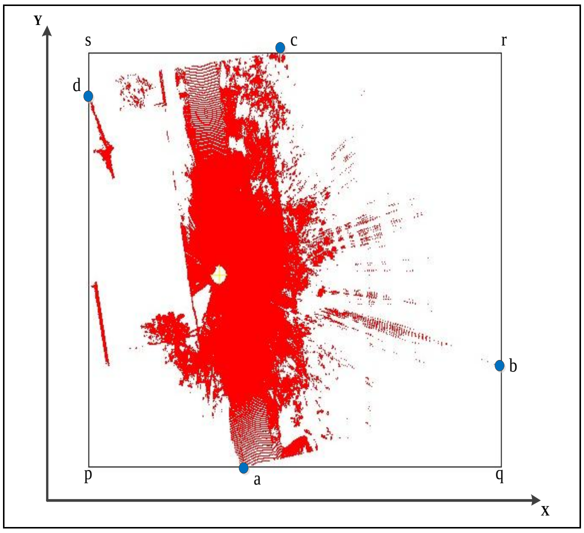


Fig. 3 *Bounding box (p, q, r, s) at 2D projected dataset*

After creation of bounding box, projected dataset has been divided into consecutive regular square grids (Fig. 4). For the division of dataset in regular square grids, difference between the minimum and maximum values of X coordinate is used as the range for the X axis (1). Similarly, the range of Y axis is calculated (2). These calculated ranges are divided into smaller segments of *m* meters and a mesh grid has been created (fig. 4). Each grid has the dimension of *m*×*m* square meters.

$$Length\ of\ bounding\ box(bl) = \left[\begin{matrix}(maximum\ X - minimum\ X)\\ or\ (Q_x - P_x)\ or\ (R_x - S_x)\end{matrix}\right] \quad (1)$$

$$Width\ of\ bounding\ box(wl) = \left[\begin{matrix}(maximum\ Y - minimum\ Y)\\ or\ (S_y - P_y)\ or\ (R_y - Q_y)\end{matrix}\right] \quad (2)$$

$$Number\ of\ rows\ (f) = \left\lceil \frac{wl}{m} \right\rceil \quad (3)$$

$$Number\ of\ coloumns\ (g) = \left\lceil \frac{bl}{m} \right\rceil \quad (4)$$

Where, P_x, Q_x, R_x, S_x are the X coordinate value of bounding box corner points and P_y, Q_y, R_y, S_y are the Y coordinate value of bounding box corner points.

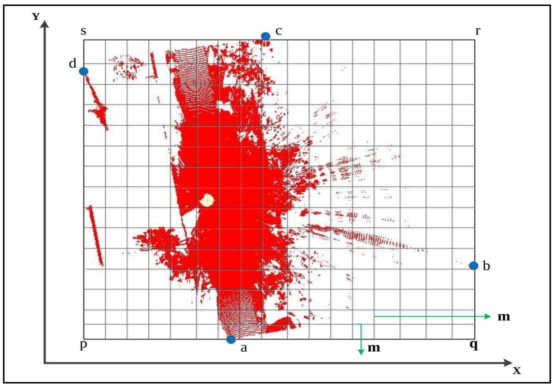


Fig. 4 *Two dimensional grid formation at 2D projected dataset*

2.2 Depth Image Generation

For each grid, points belonging to the same square grid are taken collectively and average of their Z coordinate value is calculated (5). These values are treated as Digital Number (DN) value of depth image for corresponding grid. Each pixel of depth image is represented by a particular grid. The number of pixels in depth image is depend on the grid size (*m*) (3,4,6).

$$DN_p = \left\lfloor \frac{\sum_{i=1}^{n} Z_i}{n} \right\rfloor \quad (5)$$

$$Number\ of\ pixel\ in\ depth\ image = f \times g \quad (6)$$

Where DN_p is Digital Number value for the pth grid, *f* shows the number of rows and *g* shows the number of columns of generated depth image. Fig. 5 shows the generated depth image.

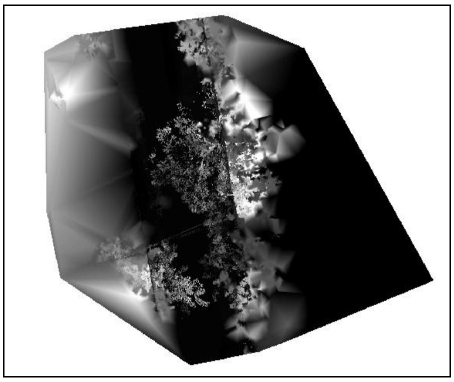


Fig. 5 *Generated depth image*

2.3 Finding the pixels having same depth

The vertical distance or difference in elevation between two consecutive contour lines is called contour interval, a contour interval (*cntr_i*) has been defined before initiating the step. Initially, DN values of depth image are sorted in decreasing order. Pixel with the highest DN value is chosen as the seed pixel and all other reaming pixels having the same DN value (equal to seed pixel DN value) are identified. Now, all these pixels are taken collectively. Similar, procedure is applied at next seed points. Next seed point is calculated by taking the difference between previous seed point and contour interval (7). The same procedure is repeated until the seed point is reached at the lowest DN value (8).

$$(Seed\ Point)_{i+1} = (Seed\ Point)_i - (cntr\_i) \quad (7)$$

$$Number\ of\ contour\ layer\ (cl) = \left\lceil \frac{maximum\ z - maximum\ z}{cntr\_i} \right\rceil \quad (8)$$

2.4 Labeling and joining of same depth pixels

All the pixels having same depth values (collected in previous step) are labeled with their depth (DN) value and the pixels with equal label are joined to each other by creating the convex hull. Convex hull is created with the help of Graham's scan algorithm.

3 Experimental Study

3.1 Test Data

The TLS dataset used for the testing of proposed method is captured from the campus of Motilal Nehru National Institute of Technology (MNNIT), Allahabad city, India (25° 29’ 39.2064’’ N, 81° 51’ 43.326’’ E). Dataset is captured with help of FARO Focus3D X 330 along the corridor of Yamuna gate road. Fig. 6 shows the Google Earth image of the corresponding location along with side and top view of captured dataset. Red rectangular area (Fig. 6) shows the particular Yamuna gate road. Statistical specification of the captured dataset is shown in Table 1. Captured dataset has no road curbs or road marking. It includes utility poles attached to power lines, trees, dense vegetation and building facades etc.

Table 1: *Statistical specification of captured dataset*

Name	File Size	No. of Points	Area ($meter^2$)	Point Density/$meter^2$
Dataset	306 MB	5492422	1971.1690	2786.3780

File size in Mega-Bytes (MB), point density is in per meter square (/$meter^2$) and total covered area by the captured dataset is in meter square ($meter^2$).

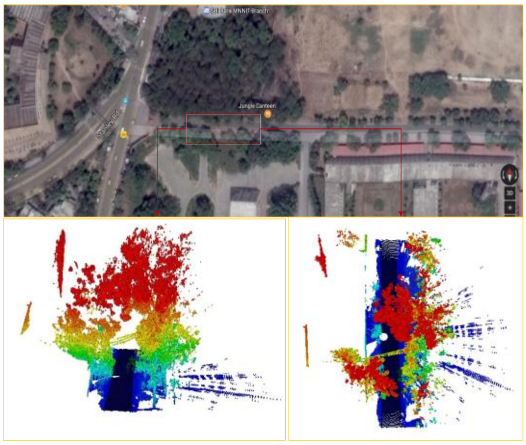


Fig. 6 *Google Earth image along with captured dataset*

3.2 Results

Proposed method is tested at captured TLS point cloud dataset (Fig. 6). Statistical specification of the same is shown in Table 1. Methodology uses two parameters (*m*, *cntr_i*) in different steps. The threshold (used) values of these parameters are shown in Table 2.

Table 2: *Parameters and their used values in different steps of proposed method*

Step	Parameter(s)	Value(s)
Gridding of data	*m*	0.1 meter
Finding the pixels having same depth	*cntr_i*	2 meter

3.3 Output Dataset

Fig. 7 shows the generated contour lines after applying the proposed method at captured TLS data. ArcScene 10.3 (ESRI) has been used to visualize the generated contours lines in three dimensional (3D) modes.

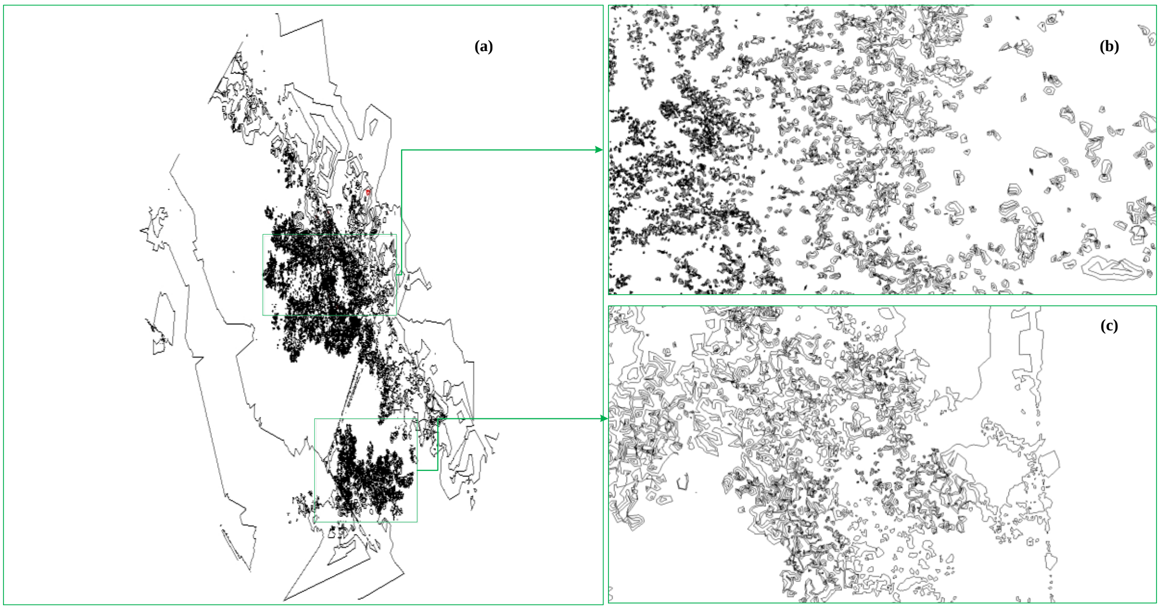


Fig. 7 (a) *Perspective view of generated contour lines* ***(b, c)*** *zoomed view of generated contour lines*

4. Discussion

Proposed method uses only X, Y, and Z coordinate of each TLS point. So, the method does not depend on additional information such as intensity; scan line, number of returns, return numbers, GPS time, etc. Proposed method is independent at point density.

4.1 Execution Time Analysis

Proposed method is coded at Matlab2013a installed on Sony Vaio E Series notebook (OS: Windows7 64bit, CPU: Intel Core i3@2.4GHz, RAM: 3GB). Execution time of the proposed method at standard parameters values (Table 2) is 108.14 seconds. Execution time of the proposed method depends on the grid size (*m*) and contour interval (*cntr_i*). If the value of grid size decreases, number of formed square grids increases (9). Therefore, lots of processing time will be consumed in creation of grid itself; also more number of grids takes more execution time (Fig. 8). Likewise, if the value of grid size increases, number of formed square grids decreases (9). Therefore, time consumption in creation of square grid decreases, also less number of grids takes less execution time (Fig. 8). In case of contour interval (*cntr_i)* parameter, if interval size decreases, more number of contour layers (*cl*) are generated (8). Therefore, exaction time of proposed method will increase (Fig. 9). Similarly, increased contour interval value takes less execution time (Fig. 9).

$$Number\ of\ square\ grid = \left\lceil \frac{Total\ area\ of\ bounding\ box}{Area\ of\ single\ grid\ (m \times m)} \right\rceil \quad (9)$$

Where, *total area of bounding box* represents the total area formed by the bounding region *p, q, r,* and *s* (Fig. 4).

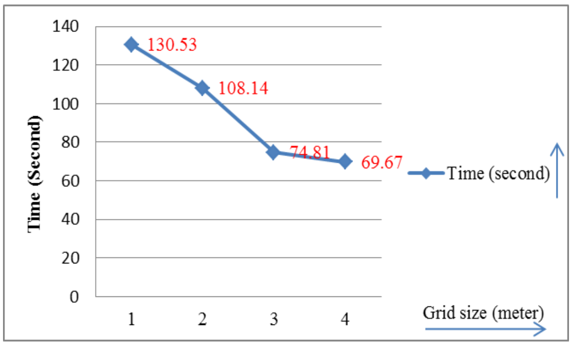


Fig. 8 *Run time performance of proposed method at different value of grid size (m)*

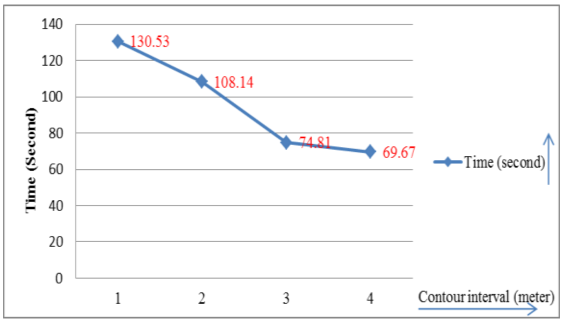


Fig. 9 *Run time performance of proposed method at different contour interval (cntr_i)*

5. Conclusion and Future Recommendations

In present study a novel automated method for the generation of contour lines using TLS data has been proposed. Proposed method consist four steps in order to generate the contour lines. Only the X, Y, and Z coordinate values each TLS point is used. Convex-Hull is used to connect the same depth pixels. Proposed method does not use any training data. No assumption has been established for generation of contour lines. Method has been tested at captured TLS dataset and corresponding contour lines are generated. Proposed method has been implemented at Matlab2013a and run time analysis is also performed. Future work will be focused on developing such methodologies to automatically determine the optimized value of employed parameters.

References

[1] Kang, T. C., "Introducation to geographic information system" 4th ed., Tata Mcgraw hill, New York 2007.

[2] Endre, K., "Contour line thinning and multigrid generation of raster based digital elevation models", Institutes of informatics, University of Szeged.

[3] Liu, X., "Airborne Lidar for DEM generation: some critical issues", Prog. Phys. Geog. 32, PP. 31-49, 2008.

[4] Awano, T., Kunii Y., "Survey Technique of Historic Garden by Using 3D and Direct Measurements in the Case Study of Seikan-Tei Garden", JILA Technical Reports of Landscape Architecture No.7, PP. 126-129, 2013.

[5] Kukko, Hyyppä, J., "Small-footprint laser scanning simulator for system validation, error assessment, and algorithm development", Photogramm. Eng. Remote Sens. 75, PP. 1177-1189, 2009.

[6] Kunii, Y., Yanagi, T., Yamazaki, M., "Grasping of Landscape in Campus by using Terrestrial Laser Scanner and its Application", Journal of Agriculture Science, Tokyo University of Agriculture Vol. 55 No.2: PP. 199-204, 2010.

[7] Jacobsen, K., Lohmann, P., "Segmented filtering of laser scanner DSMs" In: The International Archives of Photogrammetry and Remote Sensing, 34 (3/W13), 2003.

[8] Chang, Y.C., Habib, A.Y., Lee, D.C., Yom, J.H., "Automatic Classification of LIDAR Data Into Ground and Non-Ground Points", In: The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Beijing, Vol. XXXVII. Part B4, 2008.

[9] Vosselman, G., "Slope based filtering of laser altimetry data", In: The International Archives of Photogrammetry and Remote Sensing, 33(B3), PP. 935–942, 2000.

[10] Axelsson, P., "DEM generation from laser scanner data using adaptive TIN models", In: The International Archives of the Photogrammetry and Remote Sensing, 33 (B4/1), PP. 110– 117, 2000.

[11] Kraus, K., Pfeifer, N., "Advanced DTM generation from LiDAR data", In: The International Archives of Photogrammetry and Remote Sensing, 34(3/W4), PP. 23–30, 2001.