

## NEW AND RAPID METHODS FOR GRIDDING OF POINT CLOUDS

A. Mahphood<sup>1,2\*</sup>, H. Arefi<sup>1,3</sup>

<sup>1</sup> School of Surveying and Geospatial Engineering, College of Engineering, University of Tehran, Tehran, Iran - (ahmadmahphood, hossein.arefi)@ut.ac.ir

<sup>2</sup> Faculty of Surveying and Geomatic Engineering, Tishreen University, Latakia, Syria - ahmadmahphood@tishreen.edu.sy

<sup>3</sup> i3mainz – Institute for Spatial Information and Surveying Technology, Mainz University of Applied Sciences, Germany - hossein.arefi@hs-mainz.de

### Commission IV, WG IV/3

**KEY WORDS:** Point Cloud, Grid Network, Grid Points, Mesh, Resampling, Voxel.

### ABSTRACT:

Point clouds are among the most important photogrammetry and remote sensing data that provide three-dimensional information about objects and features. These points have unorganized, irregular structures and do not have uniform spacing. In this paper, two innovative methods for converting the point cloud into a grid are presented. The first method is called the mesh method. It generates all possible 3D grid points covering the point cloud. Then, the distances between the generated grid points and the original points are measured. Based on these distances, the required grid points are determined. The second method is the resampling method. It uses three equations to resample the original point so that the points are moved to new coordinates to create the grid points. The methods are tested on three different point clouds. The results show that the two methods are very fast. However, the resampling method is much faster, especially for very large areas.

### 1. INTRODUCTION

Nowadays, point clouds are used to create high-quality and high-density features and models. These datasets are acquired by laser scanning techniques such as Airborne LiDAR (Light Detection And Ranging) or ALS, Terrestrial Laser Scanning (TLS), and Mobile Mapping System (MMS). They are also created using photogrammetric methods and datasets (Alidoost and Arefi, 2015, Mahphood et al., 2019) to provide efficient data for various applications.

The point clouds have an unstructured irregular structure, which leads to high time consumption in processing. Therefore, in some studies they are converted into constructed regular networks (in point or pixel space) or irregular networks (such as Triangulated Irregular Network (TIN)) (Chen et al., 2006, Awrangjeb, 2016). By using these networks, the processing speed is increased. Thus, the raster technique creates regular points from irregular points and reduces the size of the point cloud, increasing the processing speed for very dense point clouds. This technique can be divided into the pixel-based technique (raster space) and the point-based technique (vector space). In the pixel-based technique, the points are converted into raster images such as a digital surface model (DSM) (Brédif et al., 2013, Yu et al., 2021) or a binary image (Galvanin and Dal Poz, 2011, Yang et al., 2013) or a normalized digital surface model (nDSM) (Haithcoat et al., 2001), or a 3D raster (Gorte and Pfeifer, 2004). In the point-based technique, points are divided into a regular grid of cells in 2D space (Mahphood and Arefi, 2017, Mahphood and Arefi, 2022) or a grid of voxels (cubic) in 3D space (Xiong et al., 2021, Xu et al., 2021). For the 2D grid, a point is selected instead of the points within the cell. For the 3D grid, one point is selected instead of the points within the voxel. Usually, the selected points are the centers of the cells or voxels. Nevertheless, the point-based grid technique is used in many applications, such as filtering ground

points (Qin et al., 2018, Mahphood and Arefi, 2020a), building boundary extraction (Zhou and Neumann, 2008, Mahphood and Arefi, 2022), digital terrain model (DTM) generation (Mongus and Žalik, 2012), segmentation (Deschaut and Goulette, 2010, Wang and Tseng, 2011), forest canopy vertical space characterization (Chasmer et al., 2004, Popescu and Zhao, 2008), skeletonization (Bucksch et al., 2009), and point cloud registration (Xiong et al., 2021).

Few studies explain how to generate grid points from a point cloud, as they may use the axiomatic method (Weishampel et al., 2000, Popescu and Zhao, 2008). In this method, a horizontal window moves in horizontal space with a vertical interval (or bin) to detect the points within that interval. Xiong et al. (2021) created the grid of point clouds by finding the minimum values of the coordinates in the X, Y, and Z directions. Then, these values are subtracted from the coordinates of the points, and the resulting values are divided by a grid distance (or voxel size). This method creates a grid point in a local coordinate system. Gorte and Pfeifer (2004) converted the point cloud into a 3D raster. This 3D raster represents the 3D grid in raster space with elements called voxels. In this method, voxels have two values: 0 and 1. If a voxel has 0, it is an empty voxel; if it has 1, it contains points.

In this paper, we present two new and fast methods for converting point clouds into 3D grid points. The first method creates grid points in the first step. Then the required grid points are selected. This method is called the mesh method. In the second method, the points are resampled in x, y, and z directions. The resampled points are the required grid points. Therefore, this method is called the resampling method. Then, the two methods are compared in terms of the speed and size of the generated grid points. In this work, the programming code was written in Matlab R2020b.

\* Corresponding author

## 2. THE PROPOSED METHODS

Figure 1 illustrates the flowchart of the proposed methods. First, the mesh method is implemented by creating the covering grid points. Then, the final grid points are selected based on the distances between the covering grid points and the original points. In the resampling method, the grid points are generated using three equations. Then, the duplicate points are removed. Finally, these methods are compared with each other.

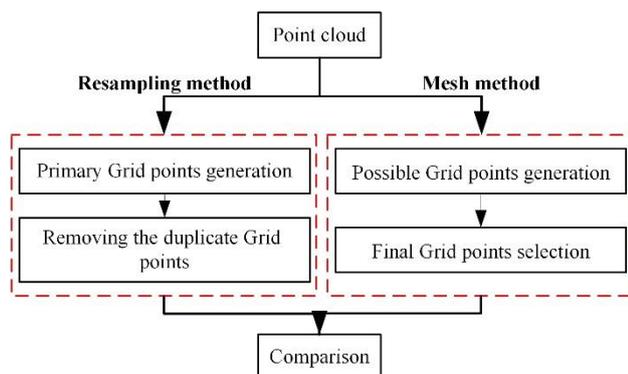


Figure 1. Workflow of the proposed methods.

### 2.1 Overview

There are two cases in which a point cloud can be converted into a grid in point space: 2D space and 3D space (Figure 2). The two cases are similar in terms of the working mechanism. In the first case, the point cloud is divided into a mesh of cells with an equal grid distance  $G_d$  (squares) (Figure 2 (a)). The required cells are the cells that contain at least one point (the black cells in Figure 2(a)). In contrast, the empty cells are invalid and are removed from the post-processing (red cells in Figure 2(a)). In the end, the centers of the remaining cells are selected as the required grid points. In the second case, the procedure follows the exact mechanism. The voxels with at least one point are detected, and the remaining ones are removed.

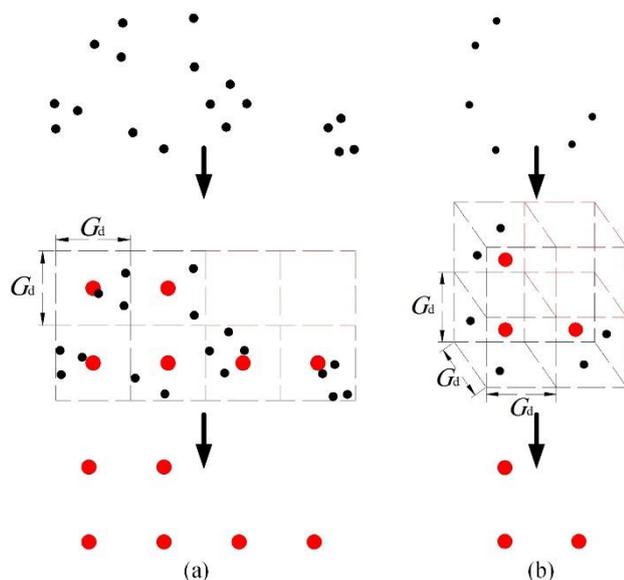


Figure 2. Examples of converting a point cloud into a grid. a) 2D space, b) 3D space. Top) the point clouds, middle) the boundaries of the grid, bottom) the resulting grid points. The black points are the point cloud. The red points are the grid

points. The dashed black lines are the non-empty cells or voxels. The dashed red lines are the empty cells or voxels.

### 2.2 Mesh Method

This method initially generates all possible grid Points without using the coordinates of the original points. Then, it selects the required grid points. Subsequently, this method has two steps:

**2.2.1 Possible Grid Points Generation:** In this step, all possible grid points for the point cloud are generated. First, the extreme values of the point cloud are extracted: min x, min y, min z, max x, max y, and max z. The two extreme values, min and max are used for each axis to generate a vector of points separated by a distance equal to  $G_d$ . A 3D grid is then generated using the *meshgrid* function in Matlab (Figure 3(b)). The resulting points are all possible grid points for the point cloud.

**2.2.2 Final Grid Points Selection:** Each generated grid point is checked to see if it should be kept or removed. To do this, the distances between the generated grid points and the original points are measured using the K-nearest neighbor method. Thus, for each point in the point cloud, the nearest grid point is determined. The result of this process is an index vector that indicates the number of grid points closest to each point in the point cloud. This index contains duplicate numbers because grid points are inevitably located near multiple points. Subsequently, the duplicate numbers are removed from this vector. The resulting index vector represents the final grid points (Figure 3(c)).

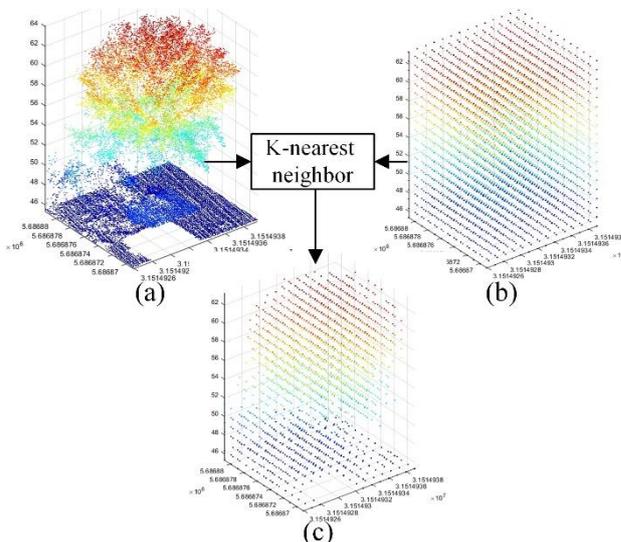


Figure 3. Mesh method. a) Point cloud, b) Possible grid points generation, c) final grid points selection.

### 2.3 Resampling Method

This method works in reverse compared to the first method. The grid points are not generated, but the original points are shifted towards the grid points. The original points will be converted into grid points through the resampling process of the original points. The point cloud is resampled using the equations (1)-(3). Mahphood and Arefi (2020b) used the equations (1)-(2) to resample the point cloud and extract the virtual first and last pulse within each local area. As the points that have the same resampled horizontal coordinates are detected. Then the highest point is detected as the first pulse, and the lowest is detected as the last pulse.

$$x_G = \text{round}(x/G_d) \times G_d \quad (1)$$

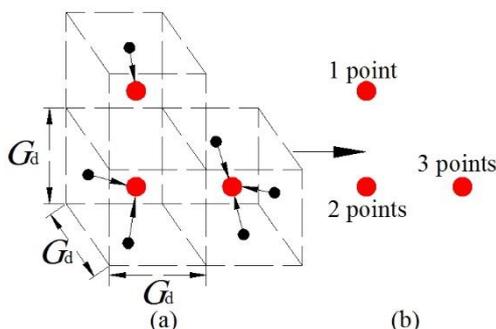
$$y_G = \text{round}(y/G_d) \times G_d \quad (2)$$

$$z_G = \text{round}(z/G_d) \times G_d \quad (3)$$

where  $x$ ,  $y$ , and  $z$ : the coordinates of the point cloud.  
 $x_G$ ,  $y_G$  and  $z_G$ : the coordinates of the grid points.

This paper presents a new method for generating a grid network using the equations (1)-(3). The resampling procedure needs a sampling distance. Therefore, the sampling distance is the same as the grid distance  $G_d$ . Thus, using the equations (1)-(3), primary grid points (resampled points) will be generated. In other words, the original points inside each voxel will shift towards the voxel center to generate the primary grid points (Figure 4(a)). But, after resampling, there are duplicate grid points at the same locations (Figure 4(b)). Thus these points are removed by detecting the points with the same coordinates. The remaining grid points are the final ones.

To explain the resampling mechanism, from the equations (1)-(3), the coordinates of the original points are rounded to the nearest number of the specified grid distance. In other words, the original points are resampled to the grid points in  $x$ ,  $y$ , and  $z$  directions using the equations (1)-(3), so that the coordinates are transferred to the nearest grid distance  $G_d$ . The three expressions of  $x/G_d$ ,  $y/G_d$ , and  $z/G_d$  are calculated and then rounded to the integer values. Finally, the values are multiplied by  $G_d$  in order to give grid point coordinates  $x_G$ ,  $y_G$ , and  $z_G$ .



**Figure 4.** Resampling method. a) Shifting the original points inside each voxel towards the voxel center, b) duplicate grid points. The black points are the point cloud. The red points are the grid points.

Data	Dimension (m <sup>2</sup> )	Number of points (point)	Terrain Features
Area 1	42*82	297082	Trees, cars, small buildings.
Area 2	106.3*83.7	986442	A mixture of vegetation and small buildings, power lines.
Area 3	140.3*133.7	1751014	Cars, complex and large buildings, power lines.

**Table 1.** Characteristics of the studied areas.

### 3. EXPERIMENTAL RESULTS AND ANALYSIS

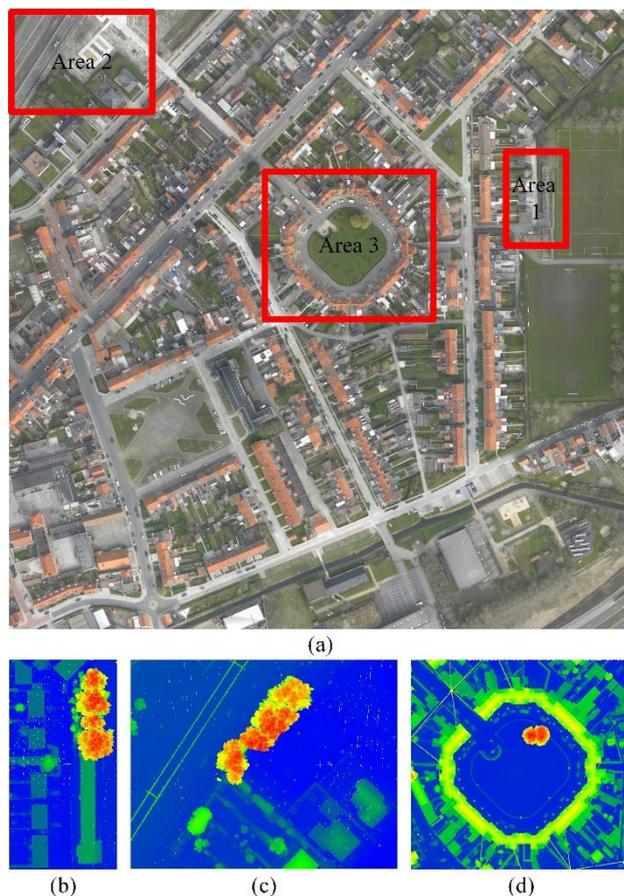
#### 3.1 Datasets

To evaluate the results, the LiDAR point cloud of the 2015 IEEE (Institute of Electrical and Electronics Engineers) GRSS dataset is used (Figure 5(a)). The data cover an urban and harbour area in Zeebrugge, Belgium. The density of the LiDAR point cloud is

approximately 65 points/m<sup>2</sup>, which is related to the point spacing of about 10 cm. Three areas are extracted from this data to evaluate the proposed methods (Figure 5(b)-(d)). These areas differ in the number of points, dimensions, and Terrain features (trees, buildings, land, etc.). The characteristics of the areas are shown in Table 1.

#### 3.2 Results, Evaluation, and Comparison

The two proposed methods are performed in areas with different grid distances. The distances used are 0.5, 0.75, and 1 m. The results are shown in Figure 6, Table 2, and Table 3. From the visual interpretation of Figure 6, we notice that the results of the methods have almost the same shape and structure for all the areas and used grid distance. However, there are some differences between the generated grid points in some regions. In Figure 6 (b) and (c), the regions that are inside the red circles are a little different from each other. In other words, the structure of the grids differs in these regions between the two proposed methods. For accuracy, Table 2 shows that the number of generated grid points for the proposed methods is slightly different for corresponding regions and grid distance. This confirms the previous result. Moreover, the difference between the number of corresponding grid points increases when the number of area points increases.



**Figure 5.** a) IEEE dataset of the Zeebrugge, Belgium area, b) area 1, c) area 2, d) area 3.

For quantitative evaluation, Table 2 shows the number of grid points generated for each area, method, and grid distance. This number clearly decreases as the grid distance increases. Table 3 shows that the two proposed methods are very fast in terms of processing time. For the mesh method, the processing time

increases when the grid distance decreases or the data size increases. The maximum running time is about 6.5 seconds for the largest area (area 3) and the smallest grid distance (0.5 m). The minimum runtime is about 0.5 seconds for the smallest area (area 1) and the maximum grid distance (1 m). The main disadvantage of this method is that it consumes a lot of time for large areas with minimum grid distance. This disadvantage results from measuring the distances between the original points and the possible grid points. When resampling distance, it can be seen that this method is not affected by the grid distance. This is because the times are almost the same. So this is one of the main advantages of this method. For area size, the time increases by a tiny amount as the area size increases. The maximum run time is about 0.5 seconds for the largest area (area 3), and the minimum run time is about 0.08 seconds for the smallest area (area 1). This is another advantage of this method.

Data	Mesh method			Resampling method		
	Grid distance (m)					
	0.5	0.75	1	0.5	0.75	1
Area 1	38046	18808	10458	38826	17991	10563
Area 2	89541	42454	23082	90654	41914	22464
Area 3	145710	65071	37016	143559	70740	41462

**Table 2.** The number of grid points for each gridded area and each method.

Data	Mesh method			Resampling method		
	Grid distance (m)					
	0.5	0.75	1	0.5	0.75	1
Area 1	2.929	0.729	0.564	0.13	0.109	0.081
Area 2	6.045	2.363	1.609	0.312	0.275	0.269
Area 3	6.573	3.153	2.339	0.475	0.445	0.503

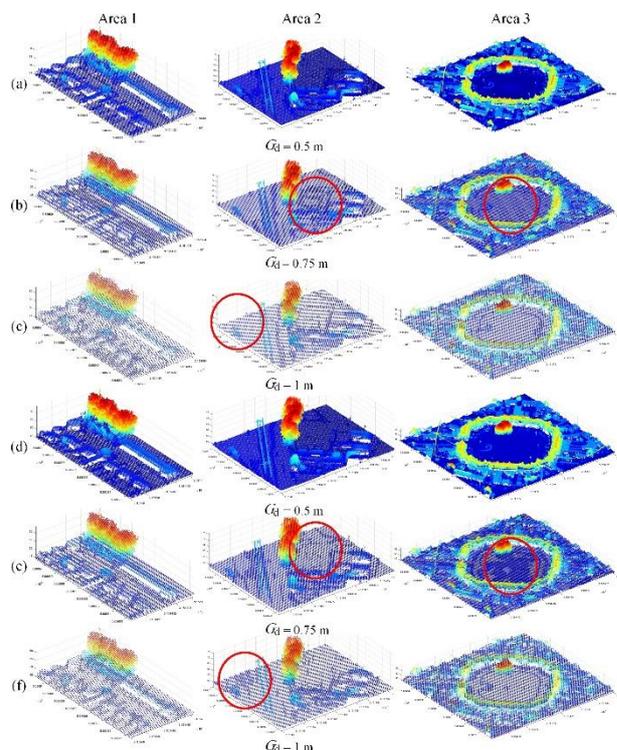
**Table 3.** The processing time of the methods (in second).

### 3.3 Discussion

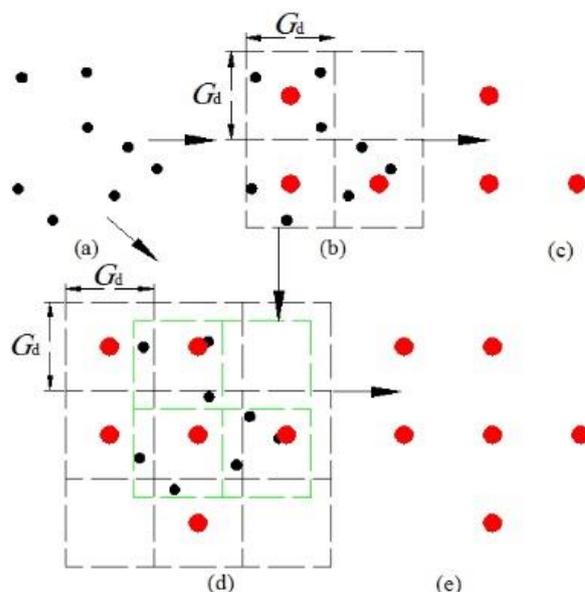
From the previous section, it can be seen that the two methods differ in the number of generated grid points, as shown in Figure 6. This difference is due to the different structures of the grid, i.e., the positioning of the grid lines according to the point cloud. The position of these lines has a great influence on the number of generated grid points. From Figure 7, it can be seen that there are two cases where the grid lines are positioned according to the same point cloud (Figure 7(b) and (d)). In the first case, the generated grid points are three points (Figure 7(b) and (c)). The second case is created by shifting the grid lines of the first case in x and y directions (Figure 7(d)). Thus, the generated grid points are six points (Figure 7(e)). Thus, the difference between the two cases is significant because only half as many grid points were generated in the first case as in the second case. At the same time, the mesh structure of the mesh method is related to the extreme coordinates of the point cloud. On the other hand, the resampling method is related to the grid distance. As a result, the grid lines of the two methods may not be identical. Thus, the number of generated grid points will be different.

In terms of processing time, we find that the two proposed methods generate grid points quickly. However, the resampling method is much faster than the mesh method because it is not

affected by the grid distance or the area size. From Table 3, the maximum running time for the mesh method is about 6.5 seconds for area 3, and the grid distance is 0.5m. In contrast, the corresponding time for the resampling method is about 0.48 seconds, which is 13 times less. This is a big difference, which becomes even bigger for larger areas.



**Figure 6.** The results of gridding the point clouds of the areas: 1, 2, and 3. a), b) and c) The results of the mesh method. d), e) and f) The results of the resampling method.



**Figure 7.** The effect of the grid network structure on the number of the generated grid points. a) The point cloud, b) the first structure of the grid network, c) the generated grid points of the first structure (three points), d) the second structure of the grid network by shifting the first one in the x and y directions, e) the generated grid points of the second structure (six points). The red points are the grid points. The dashed black lines are

the grid lines. The dashed green lines are the grid lines of the first case.

#### 4. CONCLUSION

In this paper, two novel and fast methods for gridding point clouds are proposed. The first method is the mesh method, which first generates all possible grid points and then determines the final points. This method is fast, but it is strongly affected by the size of the data used and the grid distance  $G_d$ , as it becomes slow for large data and small spacing. Since the time required multiplies with the change of the previous variable, this method is not recommended for large areas.

The second method is the resampling method, which uses three equations ((1)-(3)) to convert the points to grid points. This method is very fast and is not affected by the size of the data used and the grid distance  $G_d$ . Therefore, it is faster than the previous method, especially for large areas. Ultimately, the two methods are recommended for small areas (less than one million points), since the time difference between the two methods will not be too large. In contrast, the resampling method is recommended for large areas, as the difference will be noticeable.

#### REFERENCES

- Alidoost, F. and H. Arefi, 2015. An image-based technique for 3D building reconstruction using multi-view UAV images. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* 40(1): 43. DOI: 10.5194/isprsarchives-XL-1-W5-43-2015.
- Awrangjeb, M., 2016. Using point cloud data to identify, trace, and regularize the outlines of buildings. *International Journal of Remote Sensing* 37(3): 551-579. DOI: <https://doi.org/10.1080/01431161.2015.1131868>.
- Brédif, M., O. Tournaire, B. Vallet and N. Champion, 2013. Extracting polygonal building footprints from digital surface models: A fully-automatic global optimization framework. *ISPRS Journal of Photogrammetry and Remote Sensing* 77: 57-65. DOI: <https://doi.org/10.1016/j.isprsjprs.2012.11.007>.
- Bucksch, A., R. C. Lindenbergh and M. Menenti, 2009. SkelTreat fast skeletonisation for imperfect point cloud data of botanic trees. *Proceedings of the 2nd Eurographics conference on 3D Object Retrieval*: 13-20. DOI: <http://dx.doi.org/10.2312/3DOR/3DOR09/013-020>.
- Chasmer, L., C. Hopkinson and P. Treitz,(2004). Assessing the three-dimensional frequency distribution of airborne and ground-based lidar data for red pine and mixed deciduous forest plots. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*
- Chen, L.-C., T.-A. Teo, C.-H. Hsieh and J.-Y. Rau, 2006. Reconstruction of building models with curvilinear boundaries from laser scanner and aerial imagery. *Pacific-Rim Symposium on Image and Video Technology*: 24-33. DOI: [http://doi.org/10.1007/11949534\\_3](http://doi.org/10.1007/11949534_3).
- Deschaut, J.-E. and F. Goulette,(2010). A fast and accurate plane detection algorithm for large noisy point clouds using filtered normals and voxel growing. *3DPVT*, Hal Archives-Ouvertes Paris, France.
- Galvanin, E. and A. P. P. Dal Poz, 2011. Extraction of building roof contours from LiDAR data using a Markov-random-field-based approach. *IEEE Transactions on Geoscience and Remote Sensing* 50(3): 981-987. DOI: <http://doi.org/10.1109/TGRS.2011.2163823>.
- Gorte, B. and N. Pfeifer,(2004). Structuring laser-scanned trees using 3D mathematical morphology. *International Archives of Photogrammetry and Remote Sensing*.
- Haithcoat, T. L., W. Song and J. D. Hipple, 2001. Building footprint extraction and 3-D reconstruction from LIDAR data. *IEEE/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas (Cat. No. 01EX482)*: 74-78. DOI: <https://doi.org/10.1109/DFUA.2001.985730>.
- Mahphood, A. and H. Arefi, 2017. A DATA DRIVEN METHOD FOR FLAT ROOF BUILDING RECONSTRUCTION FROM LiDAR POINT CLOUDS. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* 42: 167–172. DOI: <https://doi.org/10.5194/isprs-archives-XLII-4-W4-167-2017>.
- Mahphood, A. and H. Arefi, 2020a. Tornado method for ground point filtering from LiDAR point clouds. *Advances in Space Research* 66(7): 1571-1592. DOI: <https://doi.org/10.1016/j.asr.2020.06.030>.
- Mahphood, A. and H. Arefi, 2020b. Virtual first and last pulse method for building detection from dense LiDAR point clouds. *International Journal of Remote Sensing* 41(3): 1067-1092. DOI: <https://doi.org/10.1080/01431161.2019.1655176>.
- Mahphood, A. and H. Arefi, 2022. Grid-based building outline extraction from ready-made building points. *Automation in Construction* 139: 104321. DOI: <https://doi.org/10.1016/j.autcon.2022.104321>.
- Mahphood, A., H. Arefi, A. Hosseinaveh and A. Naeini, 2019. DENSE MULTI-VIEW IMAGE MATCHING FOR DSM GENERATION FROM SATELLITE IMAGES. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* DOI: 10.5194/isprs-archives-XLII-4-W18-709-2019.
- Mongus, D. and B. Žalik, 2012. Parameter-free ground filtering of LiDAR data for automatic DTM generation. *ISPRS Journal of Photogrammetry and Remote Sensing* 67: 1-12. DOI: <https://doi.org/10.1016/j.isprsjprs.2011.10.002>.
- Popescu, S. C. and K. Zhao, 2008. A voxel-based lidar method for estimating crown base height for deciduous and pine trees. *Remote sensing of environment* 112(3): 767-781. DOI: <https://doi.org/10.1016/j.rse.2007.06.011>.
- Qin, H., H. Guan, Y. Yu and L. Zhong, 2018. A VOXEL-BASED FILTERING ALGORITHM FOR MOBILE LIDAR DATA. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* 42(3). DOI: <http://doi.org/10.5194/isprs-archives-XLII-3-1433-2018>.
- Wang, M. and Y. H. Tseng, 2011. Incremental segmentation of lidar point clouds with an octree-structured voxel space. *The Photogrammetric Record* 26(133): 32-57. DOI: 10.1111/j.1477-9730.2011.00624.x.
- Weishampel, J. F., J. B. Blair, R. G. Knox, R. Dubayah and D. B. Clark, 2000. Volumetric lidar return patterns from an old-growth tropical rainforest canopy. *International Journal of Remote*

*Sensing* 21(2): 409-415. DOI:  
<http://doi.org/10.1080/014311600210939>.

Xiong, B., W. Jiang, D. Li and M. Qi, 2021. Voxel Grid-Based Fast Registration of Terrestrial Point Cloud. *Remote Sensing* 13(10): 1905. DOI: <https://doi.org/10.3390/rs13101905>.

Xu, Y., X. Tong and U. Stilla, 2021. Voxel-based representation of 3D point clouds: Methods, applications, and its potential use in the construction industry. *Automation in Construction* 126: 103675. DOI: <https://doi.org/10.1016/j.autcon.2021.103675>.

Yang, B., W. Xu and Z. Dong, 2013. Automated extraction of building outlines from airborne laser scanning point clouds. *IEEE Geoscience and Remote Sensing Letters* 10(6): 1399-1403. DOI: <https://doi.org/10.1109/LGRS.2013.2258887>.

Yu, J.-W., Y.-W. Yoon, W.-K. Baek and H.-S. Jung, 2021. Forest Vertical Structure Mapping Using Two-Seasonal Optic Images and LiDAR DSM Acquired from UAV Platform through Random Forest, XGBoost, and Support Vector Machine Approaches. *Remote Sensing* 13(21): 4282. DOI: <https://doi.org/10.3390/rs13214282>.

Zhou, Q.-Y. and U. Neumann, 2008. Fast and extensible building modeling from airborne LiDAR data. *Proceedings of the 16th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*: 1-8. DOI: <http://doi.org/10.1145/1463434.1463444>.