

# Hava LiDAR Verilerininin Basit Üçgen Düzensiz Ağ Algoritması ile Filtrelenmesi

Fırat URAY<sup>1\*</sup>  Abdullah VARLIK<sup>1</sup> 

<sup>1</sup> Necmettin Erbakan University, Faculty of Engineering, Department of Geomatics Engineering, Konya, Türkiye

## Makale Bilgisi

## ÖZET

**Geliş Tarihi:** 10.05.2024  
**Kabul Tarihi:** 25.07.2024  
**Yayın Tarihi:** 31.12.2024

### Anahtar Kelimeler:

Uzaktan Algılama,  
Nokta Bulutu,  
LiDAR,  
Filtreleme,  
Sınıflandırma.

Son yıllarda havadan ve karadan lazer tarama sistemleri jeo-uzamsal bilgi elde etmek için giderek daha popüler hale gelmiştir. Yüksek kaliteli 3B nokta bulutları çok çeşitli uygulamalar için kullanılmakta olup sayısal arazi modelleri (SAM) bu ürünlerden birini temsil etmektedir. Light Detection and Ranging (LiDAR) verileri, planlama uygulamalarının temel unsurları olan sayısal yükseklik modellerinin (SYM) üretilmesinde yaygın olarak kullanılmaktadır. LiDAR nokta bulutlarından zemin dışı nesnelerin çıkarılması, DTM üretim iş akışının ana sorunudur. Üçgenleştirilmiş düzensiz ağ (DÜA) yoğunlaştırma, LiDAR filtreleme algoritmaları arasında klasik bir tekniktir. Bu çalışmada, klasik DÜA yoğunlaştırmadan türetilen basit DÜA yoğunlaştırma (sTIN) adlı basit bir filtreleme algoritması önerilmiştir. sTIN'in performansı, uyarlanabilir üçgenleştirilmiş düzensiz ağ, basit morfolojik filtre ve geliştirilmiş aşamalı DÜA yoğunlaştırma olmak üzere üç filtre ile test edilmiştir. Önerilen algoritmanın ortalama tip I hata oranı %5,6, ortalama tip II hata oranı %10,42 ve ortalama toplam hata oranı %8,2'dir. Ayrıca, algoritmaların güçlü ve zayıf yönleri, karesel ortalama hata (KOH) değerleri ve SAM'ların görsel analizleri açısından incelenmiştir.

## Filtering Airborne LIDAR Data Using Simple Triangulation Irregular Network

## Article Info

## ABSTRACT

**Received:** 10.05.2024  
**Accepted:** 25.07.2024  
**Published:** 31.12.2024

### Keywords:

Remote Sensing,  
Point Cloud,  
LiDAR,  
Filtering,  
Classification.

In recent years, airborne and terrestrial laser scanning systems have become increasingly popular for obtaining geospatial information. High-quality 3D point clouds are used for a wide range of applications, with digital terrain models (DTMs) representing one of these products. Light Detection and Ranging (LiDAR) data is widely used in producing digital elevation models (DEM), which are fundamental elements of planning applications. Removing non-ground objects from LiDAR point clouds is the main problem of the DTM production workflow. Triangulated irregular network (TIN) densification is a classical technique among LiDAR filtering algorithms. In this study, a simple filtering algorithm entitled simple TIN densification (sTIN) is proposed, which is derived from classic TIN densification. The performance of sTIN is tested with three filters, namely, the adaptive triangulated irregular network, the simple morphological filter and the improved progressive TIN densification. The proposed algorithm has an average type I error rate of 5.6%, an average type II error rate of 10.42% and an average total error rate of 8.2%. In addition, the strengths and weaknesses of the algorithms are examined with regards to the root-mean-square error (RMSE) values and visual analyses of DTMs.

### To cite this article:

Uray, F. & Varlık, A. (2024). Filtering airborne LIDAR data using simple triangulation irregular network. *Necmettin Erbakan University Journal of Science and Engineering*, 6(3), 495-511. <https://doi.org/10.47112/neufmbd.2024.61>

\*Sorumlu Yazar: Fırat Uray, [furay@erbakan.edu.tr](mailto:furay@erbakan.edu.tr)



## INTRODUCTION

Surveying technologies rapidly grown in last twenty years and contains various aspects of map making from using different approaches for geographical information systems and cadastre applications [1-2] to create 3D models using images [3] to analyzing land cover and land use with satellite images [4]. LiDAR technology allows the creation of spatial twins of any region, from small to large scale. Airborne LiDAR technology can acquire high-precision and high-density point cloud data of terrain [5]. The point cloud is the most basic form of LiDAR data and which is composed of three-dimensional points and associated attributes, like the intensity of points or color information when laser scanners are integrated with digital cameras. The generation of high-resolution DEM, which is a detailed representation of the shape of the terrain, is one of the important applications of LiDAR technology [6]. Many filtering methods have been proposed by researchers to retain terrain. Zhang et al. created ALDPAT (Airborne LiDAR Data Processing and Analysis Tools), which is open-source software used to filter LiDAR point cloud data with different five algorithms. LiDAR filtering methods can be classified into four categories; triangular irregular network (TIN) based, morphology based, slope based, and segmentation based.

One of the most widely used and reliable algorithms for ground filtering is the triangulation-based method from Axelsson known as the adaptive triangulated irregular network (ATIN) [7-8]. The ATIN uses seed points in the original point cloud data and a TIN surface is built based on the seed points. Other points are tested against these triangles with parameters that are the angle and distance between the point and triangle. If the parameters of the algorithm are below the thresholds, the algorithm iteratively adds points [9-11]. Out of eight algorithms tested, Axelsson's algorithm performed the best for twelve out of fifteen samples [12]. The ATIN's excellent performance can be attributed to its ability to retain and handle all point cloud data prior to producing a DTM. Recent attempts to surpass the performance of the ATIN method have had mixed results [6]. A triangulation-based algorithm known as, virtual deforestation (VDF), uses the smoothness of the ground. This method presumes that bare earth is generally made up of smooth surfaces with no sharp corners. Thus, object points are selected among strong curvatures points. This method first generates a TIN model and then creates a point database and converts the TIN model into a grid. The central point of the grid is then allocated the mean value inside the window. The curvature is defined as the elevation difference between the mean value and the corresponding value in the point database that was converted from the model. Two threshold values are given, one of these is to filter these points above the terrain and the other one is for removing negative outliers. The algorithm labels the non-ground points as object and builds a new TIN with the filtered ground points. This process continues until no more points are removed [11]. Zhao's proposed algorithm improves the selection of potential ground points by applying morphological operations and increases the performance of the classic TIN algorithm [12].

Furthermore, many LiDAR filtering algorithms have been developed based on mathematical morphology, which utilizes object shapes or shape measurements. Experiments on LiDAR data have shown that a morphological filter can remove non-ground objects [13]. Kilian used a progressive morphological filter on a gridded surface model, which was based on a sequence of opening operations. [14]. This method was later developed into a working algorithm by [15]. Chen proposed an improved technique that searches for large features to identify the potential non-ground points. Chen's algorithm was tested against Axelsson's algorithm with fifteen ISPRS sample LiDAR point cloud datasets and showed improved results for seven of fifteen samples [16]. Many other improved versions of progressive morphological algorithms have been proposed [17-18]. An alternative method based on geodesic transformations of mathematical morphology has also been proposed [19].

Another morphological-based algorithm is the simple morphological filter (SMRF) from Pingel.

This method has been adapted from a technique used by the methods of Zhang and Chen. This algorithm consists of four stages. The first stage includes the generation of the minimum surface from the lowest points within each cell. Applying a filter to the minimum surface is the second step of the SMRF process. In this stage, a vector of window sizes based on the supplied maximum is created. The distinctive feature of the SMRF arises in this stage. The window size is increased by one pixel per iteration up to the maximum iteration value defined by the user. The user defined slope tolerance value multiplied by the window radius and the cell size of the grid is equal to an elevation threshold value. A disk-shaped element is used to apply a morphological opening to the minimum surface to creating the next surface. The radius of this element is equal to the current window size. If the elevation difference of any of the cells between the minimum surface and the next surface is greater than the threshold, they are flagged as object cells. The next surface acts as the “minimum surface” for the next iteration. For each window size in the vector, this process continues. In the third stage, a binary grid is generated from the results of the iteration process. Each cell is classified as being bare earth or an object. A morphological mask is applied to the minimum surface to clear object cells. A provisional DEM is produced from the bare earth cells and SMRF uses advanced image inpainting techniques to interpolate the empty cells. The last step of the SMRF is the identification of the bare-earth and object points based on their relation to the interpolated DEM. The distance difference between each LiDAR point's elevation and the temporary DEM is calculated, and the result is compared to the threshold that was set. Pingel suggested one more parameter, namely the transforming threshold to a slope-dependent value to increase the threshold on steep slopes. The elevation threshold is then equal to a fixed distance plus the slope of the DEM at each point. The parameters and detailed description of the SMRF can be found in Pingel's paper [6].

Vosselman et al. proposed the first slope-based filtering algorithm. The slope between two neighbouring points is tested against the pre-defined threshold. If the slope value is greater than the threshold, one of the subject points is classified as a non-ground point [20]. However, this method does not show good performance when applied on rough terrain, such as cliffs. Susaki improved Vosselman's technique by applying changeable threshold values based on terrain types [21]. A multi-directional ground filtering algorithm (MGF) has been proposed by Meng. MGF uses the advantages of directional scanning, and it considers the slopes for neighbouring pixels in up to four directions and the elevation difference between the pixels [22]. Chen proposed a multi-resolution hierarchical classification algorithm that filters LiDAR point cloud in levels of hierarchy. This method is designed to directly and iteratively filter the point cloud data using thin plate spline-based surface interpolation. However, this method suffers from difficulties in filtering non-ground points in steep slope areas [17].

Segmentation-based methods consist of two main processing steps. Firstly, some segmentation techniques are applied to the data, such as clustering point clouds. Secondly, traditional filtering algorithms are used on the segmented data. This kind of filtering method generally performs better performance than other techniques if segmentation step is sufficiently successful. [9] proposed a region growing-based filtering algorithm based on linear prediction.

An improved progressive TIN densification filtering algorithm (IPTD) combined with a morphological method is one of the algorithms we use in this study to compare our method. The IPTD is derived from Axelsson's technique for better filtering performance for complex forested sites [18]. IPTD was created to address the considerable obstacle that for LiDAR filtering algorithms, particularly in places that are both environmentally and topographically complex. The algorithm focuses on three aspects: (1) employing the morphological method to locate possible ground seed points rather than the grid's lowest points; (2) using a buffer zone to obtain ground seed points to improve the quality of TIN to avoid the formation of unsuitable triangles; (3) the use of upward densification after downward densification to improve the ability of the IPTD to deal with slope variations [18].

In recent years, the application of machine learning techniques to image classification and recognition tasks has been highly successful. Researchers have progressively turned to employing them for the analysis of 3D point clouds. For these approaches, various supervised techniques have been adopted. Discriminant features and representative samples are two essential components that have a major influence on how effectively standard supervised classification performs. Unfortunately, the former has received relatively little attention in the published studies, as most of them validate their performances using benchmark point clouds, where the labels of the training data are typically available. However, an exception to this trend is the work of Feng and Guo [23], which presents an automatic method for sample selection that utilizes 2D land cover maps and a constructed topological graph. For the feature selection, geometric and eigen-based features are commonly used. For a representation of the local properties of points, point-based and segment-based methods extract properties based on 3D covariance matrix including 3D coordinates of points clusters.

Zhang et al. utilized a support vector machine (SVM) algorithm to classify point clouds with 13 features [24]. Lopatin et al. employed a random forest approach to classify LiDAR point clouds [25]. Niemeyer applied conditional random fields, while Shapovalov applied Markov random fields to address the point cloud classification problem [26-27]. Jahromi tested their artificial neural network-based algorithm against four samples and found an improvement over Axelsson's technique in three cases [28]. More researchers have proposed popular deep learning techniques to solve filtering problems [29-31].

In this study, we adapted Axelsson's approach to our simple TIN densification (sTIN) algorithm. In the classic PTD algorithm, every point classified as a ground point is included in the TIN construction process. Unlike Axelsson's technique, our algorithm triangulates only the lowest points in each grid instead of all classified ground points in the last iteration. In addition, the parameters are calculated only when the grid size decreases instead of at the end of each iteration. Therefore, the memory occupation decreases, and the processing capability of the algorithm increases. Before the selection of seed points, a noise-point filtering process was applied to the data to avoid false selection of ground points. Thus, it allows for the creation of a better initial surface for the start of the filtering process. Axelsson's mirroring technique was also added to the sTIN filter for the detection of non-ground points on the edge of the hillsides.

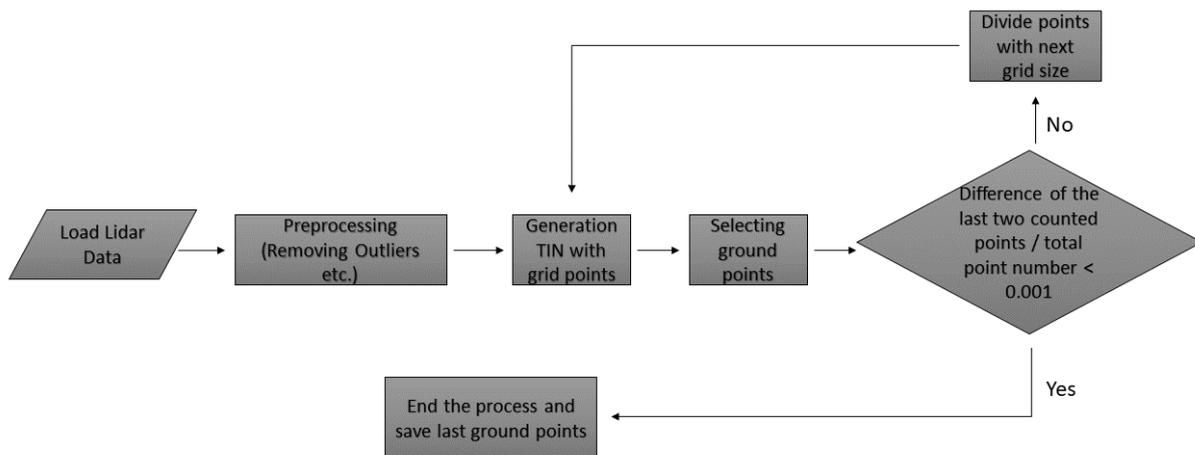
## **MATERIALS AND METHODS**

### **Principle of sTIN**

In this study, the sTIN filter was developed with two main differences from the classic TIN densification algorithm. The first difference is in the method of constructing the TIN. In sTIN, instead of using all obtained ground points in the last iteration, like the ATIN technique, classified ground points are divided into grids again and only the lowest points in the grids are used for reconstructing the next TIN in every iteration. The second difference is selection method of ground points. The ATIN technique uses changeable threshold values during the filtering process whereas our proposed method uses constant threshold values until the grid size is changed. The first step of sTIN is to remove the outliers from the entire dataset before beginning the filtering process. Outliers can be extreme points that are generated from the scanner, and it affects the selection of the lowest points in the cells. This process is necessary to improve filtering performance. LiDAR data is divided into 10-m grids and the average heights in the grid cells were used to generate a second-order polynomial surface for this purpose. Our decision to apply 10-m grids was related to generating time of the polynomial surface. As the grid size decreases, the number of points used to generate the polynomial surface increases, so the processing time also grows. To better represent topography, the grid size can be reduced when applying the outlier point filter algorithm on sloping land datasets. For the datasets used in this study, a 10-m grid size is

sufficient. The generation time of the surface is the main reason behind the selection of the second-order polynomial technique. According to [32], the performance of this method was 72% and is sufficient for filtering out outlier points from point cloud data. Outlier points can be determined by the distance between points and the polynomial surface. Two different predefined threshold values are used for the rural and urban areas, 5m and 10m, respectively [33]. The points that exceed the threshold from below and above the surface are removed from the data. Removing the outliers from the data improves the ground filtering and is a helpful stage for better triangulation from ground points.

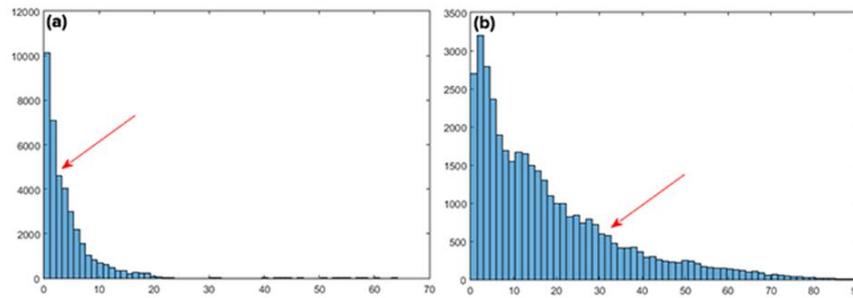
The technique then builds a buffer zone to that widens the point cloud's boundaries. The purpose of extending the boundaries is to ensure that each point belongs to a triangle. To achieve this, four points at the corners of study area are created with distance of 10m from edges of LiDAR data and adding these points in the process of generating triangles. After that, all point cloud data is divided into grids with a grid size that is larger than the maximum object size in the data. That grid size also named initial grid size. The maximum object size value should be close to the size of the largest building in the point cloud. The initial triangulation is generated by classifying all the grids' lowest points as ground points. Other non-ground points are tested with thresholds. Points that belong to the triangles are not tested and accepted as ground points. Thresholds are estimated from histogram values at the end of each iteration in classic ATIN algorithm. In sTIN, the thresholds are constant until the grid size is decreased. Thresholds are recalculated from histogram values when grid size becomes smaller than the last one. Until the final grid size is 2m, next grid size will be halved. The triangulation process is repeated for each grid size and a new TIN is created. The difference between the number of ground points from the previous triangulation and the next triangulation was calculated. If the ratio of the difference to the total number of measured points in the data is below the constant threshold, the filtering process is stopped. This threshold is set to 0.001 for our technique. The workflow of proposed technique is displayed in Figure 1.



**Figure 1**  
Flow chart of the proposed algorithm

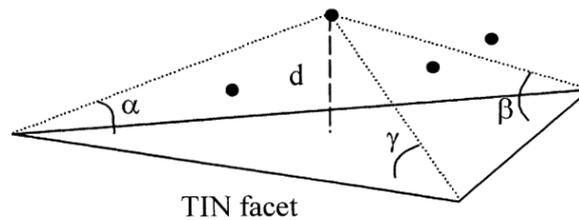
Four parameters are used for the algorithm, namely, grid size, maximum terrain angle, distance and angle between points and triangle nodes. The optimum parameters can be estimated according to the histograms in Figure 2. The histograms give the threshold values for distance and angles. These threshold values can be chosen by the operator considering the terrain types and the sharp jump values on the graph. Maximum terrain angle or slope is determined by the ratio between distance and height of the lowest and highest points in the study area. The distance ( $d$ ) and angle ( $\alpha, \beta, \gamma$ ) parameter are referred range between the triangle surface and potential ground points. The relationship among triangle and the points is given in Figure 3. The obtained ground points in the last iteration are divided into grids again. Thereafter, each lowest point in the grids is used for the construction of the TIN instead of using all

ground points. This is the main difference between sTIN and the ATIN algorithm. This technique may help to reduce the processing time.



**Figure 2**

*Histogram of (a) the distance between the points (x-axis) and the surface of triangle and (b) maximum angle between points and triangle nodes (x-axis) for sample dataset. The number of points is shown along the y-axis. Selected threshold values are marked on the graphic.*



**Figure 3**

*The distance ( $d$ ) and angle ( $\alpha$ ,  $\beta$ ,  $\gamma$ ) parameters on the triangle surfaces [8]*

In the classic ATIN, many object points that are close to the ground can be easily classified wrongly as ground points. This is because the angle between the object point and the triangle remains below the threshold. The ATIN algorithm has a weakness while classifying points on hillsides because points on the high slopes can easily exceed the threshold values. To overcome this problem, a mirroring technique is used in the triangles, which exceed predetermined maximum terrain angle for filtering.

The basis of the technique is to mirror original candidate point,  $P_c (X_c, Y_c, Z_c)$ , by the nearest point of the triangle,  $P_n (X_n, Y_n, Z_n)$ , which is located on the slope. The mirror point,  $(X_{\text{mirror}}, Y_{\text{mirror}}, Z_{\text{mirror}})$ , is then also tested with the distance and angle parameters of the algorithm (Equations 1-3) [8]. If the parameters of the mirror point are below the threshold, this candidate point is classified as a ground point. This technique is shown in Figure 4.

$$X_{\text{mirror}} = 2X_n - X_c \quad (1)$$

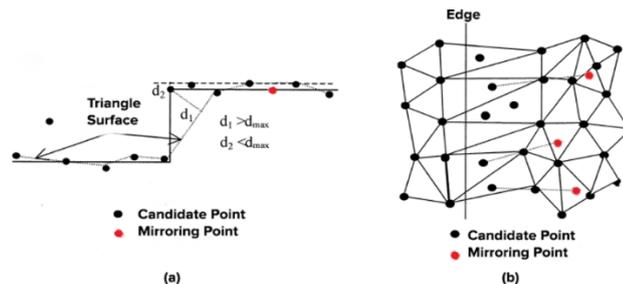
$$Y_{\text{mirror}} = 2Y_n - Y_c \quad (2)$$

$$Z_{\text{mirror}} = Z_p \quad (3)$$

### Study Area

A total of fifteen benchmark LiDAR point cloud datasets were provided by ISPRS. LiDAR filtering algorithms may show variable performance when applied to different types of terrain [13]. Therefore, five of the benchmark datasets, which represent different types of terrain, were tested with the sTIN filter in this study (Table 1). The filtered ground points for the test datasets were already provided by ISPRS for the performance assessment of filtering accuracy.

The datasets were collected with an Optech ALTM system in 2000. The average point density and spacing are 1 points/m<sup>2</sup> and 0.97m, respectively. The benchmark airborne LiDAR datasets along with


**Figure 4**

(a) Profile view of the slope and (b) mirroring by the nearest triangle points

the manually collected reference data for ground points were downloaded from a web portal [34] DEMs of the benchmark point clouds are shown in Figure 5.

**Table 1**

Attributes of test datasets

Data Set	Size	Terrain type	Slope	Elevation Range (m)
SAMP11	134 x 303 m	High slope, buildings and vegetation on the hill.	Hillside	295 - 404
SAMP12	205 x 270 m	Partially flat, contain building and road.	Low	325 - 357
SAMP23	150 x 206 m	Flat, contain nested large buildings.	Low	284 - 327
SAMP52	450 x 300 m	A River and flats with steep slope.	High	249 - 347
SAMP71	400 x 220 m	Brigde and flat terrain.	Flat	293 - 310

### Experiments and performance analysis

As discussed above, the proposed filtering algorithm requires four parameters. These parameters are set to the same values for both the triangulation-based filters and the morphological filter. For SMRF, Pingel's optimized parameters were used, as shown in Table 2 [6]. First parameter of SMRF is slope tolerance that governs the elevation threshold that is after used to calculate non-ground points. Elevation threshold used to detect ground or non-ground points is calculated by multiplying the given slope tolerance by the window radius and cell size. Windows radius is used applying morphological operations in SMRF algorithm. In addition, the same angle threshold and suggested threshold values for other parameters were used for the IPTD algorithm [10]. IPTD uses nearest points ( $k$ ) to identify non-ground points in initial potential ground points. These nearest points then used to fit a local plane ( $F$ ) on the potential points. Nearest points and distance between local plane and each point are used to calculate the parameter NormaError ( $r$ ). All the parameters were determined by the operator's judgement on the LiDAR data, rather than by a trial-and-error method, which is meaningful for the various types of applications [35].

A prototype software for filtering LiDAR datasets was developed on a computer with Intel Core i5-6500 3.2 GHz CPU and 16 GB RAM using Matlab (The Mathworks) under the Windows 10 operating system.

#### Metrics for Performance Assessment

The RMSEs of DTMs generated from the ground points with a resolution of 1m were used to evaluate the volumetric quality of the filtered data. DTMs were created using Kriging interpolation algorithm. Many factors influence the creation of a DTM, including the interpolation method, resolution, and sampling error. Additional information about these parameters can be found in [36], although it is outside the scope of this study.

$$RMSE_Z = \sqrt{\frac{\sum_{i=1}^n (Z_i - \hat{Z}_i)^2}{n}} \quad (4)$$

The RMSE was then calculated from the deviations of elevation values in a filtered DTM ( $Z_i$ )

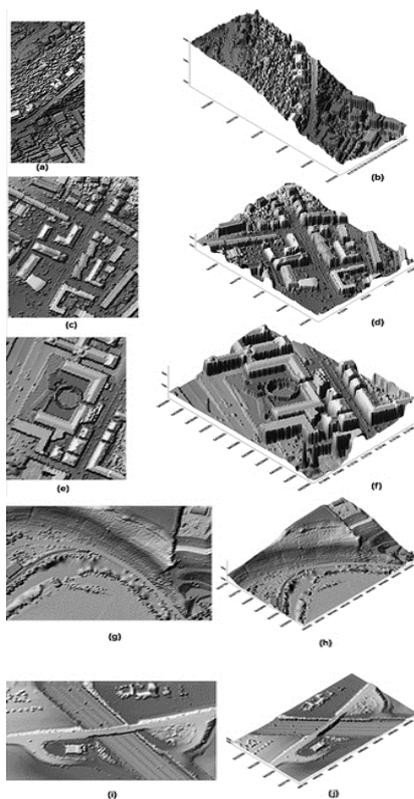
from its reference DTM ( $\hat{Z}_i$ ). The total number of points in the study area is represented by (n). Using knowledge of the terrain and accessible aerial photos, manual filtering was used to create the reference data [12]. The literature reports that the accuracy of a DTM produced by LiDAR is in the order of  $\pm 15$  cm [37-39]. When filtering LiDAR data, there are two common errors that can be committed. One is to classify ground points as non-ground measurements and the other is to select non-ground points as ground measurements. The former is known as an omission error (type I) while the latter is a commission error (type II) [40]. In addition, in this paper, we used RMSE values for the performance assessment. Moreover, omission and commission errors are represented by deviations in RMSEs. Both positive and negative volumetric errors can result from omission errors [41]. Airborne LiDAR's vertical accuracy in low-vegetative areas (such as open terrain or concrete) ranges from 0.15 to 0.25 m RMSE, according to Eq. (4) [42–46], with normally distributed errors (mean error approaching zero).

**Table 2**  
*Selected parameters for ATIN, sTIN, SMRF and IPTD*

	SAMP11	SAMP12	SAMP23	SAMP52	SAMP71
<b>ATIN and sTIN</b>					
Initial Grid Size (m)	30	30	40	35	25
Max. Terrain Slope (%)	17	3	15	10	5
Distance (m)	2	2	4	2.4	4
Angle (°)	30	45	45	25	20
<b>SMRF</b>					
Slope Tolerance (%)	20	18	27	13	13
Windows Radius (m)	16	12	13	13	15
Elevation Threshold (m)	0.45	0.30	0.50	0.25	0.75
Scaling Factor	1.20	0.95	0.90	2.20	0.00
<b>IPTD</b>					
k Nearest	300	300	600	200	300
NormaError (r)	0.5	0.5	0.8	0.8	0.8
Angle (°)	30	45	45	25	20
Distance (m)	2	2	1.5	2.4	3

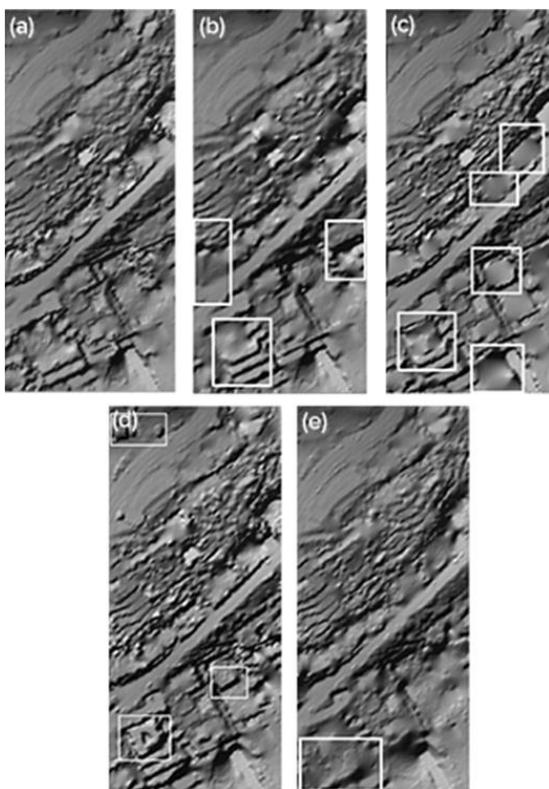
### *Visual Analyses*

For the Sample11 dataset, the ATIN shows a partially good performance in terms of visual evaluation, as shown in Figure 6. Points that belong to the vegetation layer were detected reasonably well on the high slope; nevertheless, on the hillside, it was hard to identify nested buildings for the algorithm. The SMRF algorithm also shows a close performance to the ATIN algorithm. As can be seen in Figure 7, vegetation layer on the high slope was filtered acceptably but some nested buildings could not be identified. Some of the buildings on the hillside were not detected completely while the vegetation layer was filtered successfully by sTIN. Our method was also not able to remove all the ditch slopes on the edge of the roads. sTIN achieves highest type I and type II error rates, 10.9% and 22%, respectively (Table 3). The total error rates of ATIN, sTIN, SMRF and IPTD for this dataset are 10.76%, 15.69%, 8.28 % and 13%, respectively.



**Figure 5**

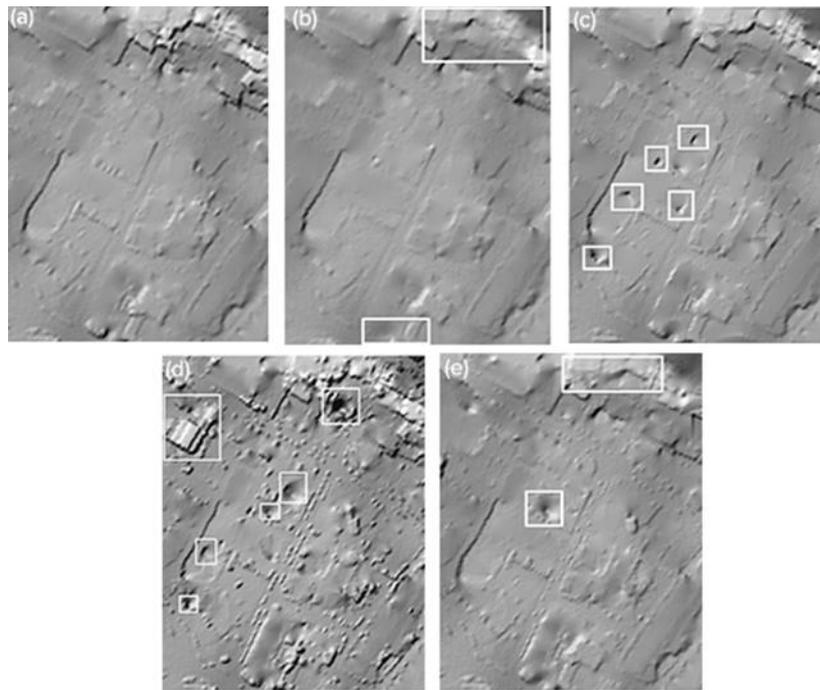
Datasets are displayed in shaded model maps of DEMs with cell size 1m. (a) SAMP11 in 2d (b) SAMP11 in 3d (c) SAMP12 in 2d (d) SAMP12 in 3d (e) SAMP23 in 2d (f) SAMP23 in 3d (g) SAMP52 in 2d (h) SAMP52 in 3d (i) SAMP71 in 2d (j) SAMP71 in 3d



**Figure 6**

DTMs produced for SAMPLE11 using filtered data. DTMs from reference data are shown in (a) and DTMs from test filters are shown in (b) ATIN algorithm, (c) SMRF, (d) IPTD, (e) sTIN algorithm.

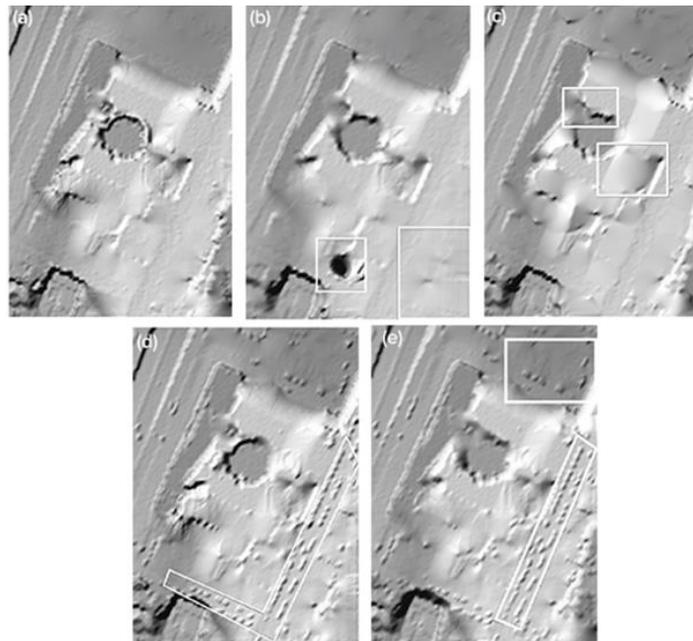
Sample12 contains a partially flatter surface than other datasets as shown in Figure 7. It has different sizes and types of buildings with a high density of cars. This means that there are relatively small elevation differences between ground and non-ground points. The SMRF was able to identify and filter buildings. However, the SMRF and IPTD were not able to filter outliers in the dataset, such as the other two algorithms. In addition, IPTD was the only algorithm that could not filter out the building in the north-east area of the dataset and resulted in a larger RMSE error. The ATIN and sTIN partially smoothed the abrupt slope changes in the terrain located in the north-east area of the dataset. The SMRF showed the best performance with type I and type II error rates of 3% and 7.5%, respectively (Table 3). The total error rates of ATIN, sTIN, SMRF and IPTD for this dataset are 3.25%, 5.3%, 2.92% and 9%, respectively. In terms of the RMSE values of DTMs, decent results were achieved by applying four filters to this dataset, as shown in Figure 11.



**Figure 7**

*DTMs produced for SAMPLE12 using filtered data. Visible filtering errors are shown in the white boxes. DTM from reference data are shown in (a) and DTMs from test filters are shown in (b) ATIN, (c) SMRF, (d) IPTD, (e) sTIN*

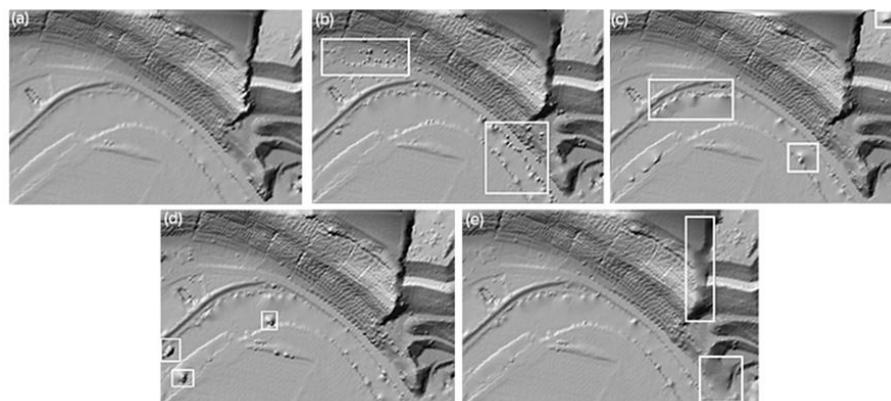
Dataset Sample23 has nested buildings in an urban area. The SMRF, IPTD, and sTIN filtered the sloping region located in south-east area of the dataset reasonably well. Furthermore, these algorithms maintain the surface of the terrain. The ATIN failed to keep the true elevation of the terrain. While the SMRF, IPTD and sTIN were successfully filtered outliers, ATIN was not able to remove all outlier points. ATIN and SMRF have a capability to identify and remove the vehicles on the roads but IPTD and sTIN were failed to detect those objects. The total error rates of ATIN, sTIN, SMRF and IPTD for this dataset are 4%, 11.5%, 4.61% and 7%, respectively (Table 3). IPTD achieved the best RMSE value in four algorithms with 0.71m, as shown in Figure 8.



**Figure 8**

*DTMs produced for SAMPLE23 using filtered data. Visible filtering errors are shown in the white boxes. DTM from reference data are shown in (a) and DTMs from test filters are shown in (b) ATIN, (c) SMRF, (d) IPTD, (e) sTIN*

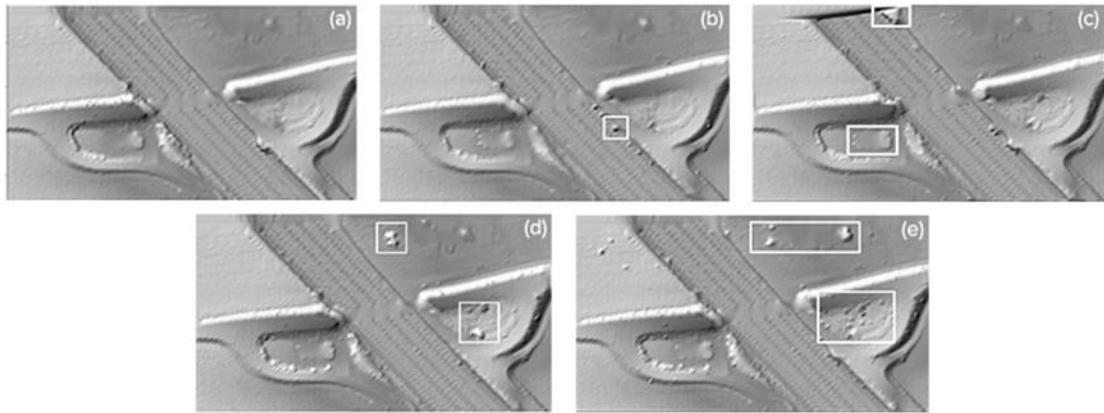
Dataset Sample52 is located in a riverside area and contains a large slope. There is a terraced field with large elevation differences located in east area and the vegetation layer near the river in the south. Figure 9 shows the reference DTM and the filtered DTM by four filters. While sTIN filtered the vegetation and trees on the riverside very well, ATIN, SMRF, and IPTD could not perform as well as our algorithm. In this dataset, the TIN-based filters showed better performance than the morphological-based filter. At the start of the slope, sTIN and SMRF were able to detect and remove the vegetation surface, but the same thing cannot be said for ATIN filter. The SMRF was the only algorithm that could not identify the building that is located on the hill. Other algorithms were able to identify and remove the building. IPTD achieved best RMSE value in this dataset than all other techniques. In addition, sTIN has achieved better performance than the SMRF algorithm in terms of RMSE values. The total error rates of ATIN, sTIN, SMRF and IPTD for this dataset are 3.07%, 6%, 3.82% and 4%, respectively (Table 3).



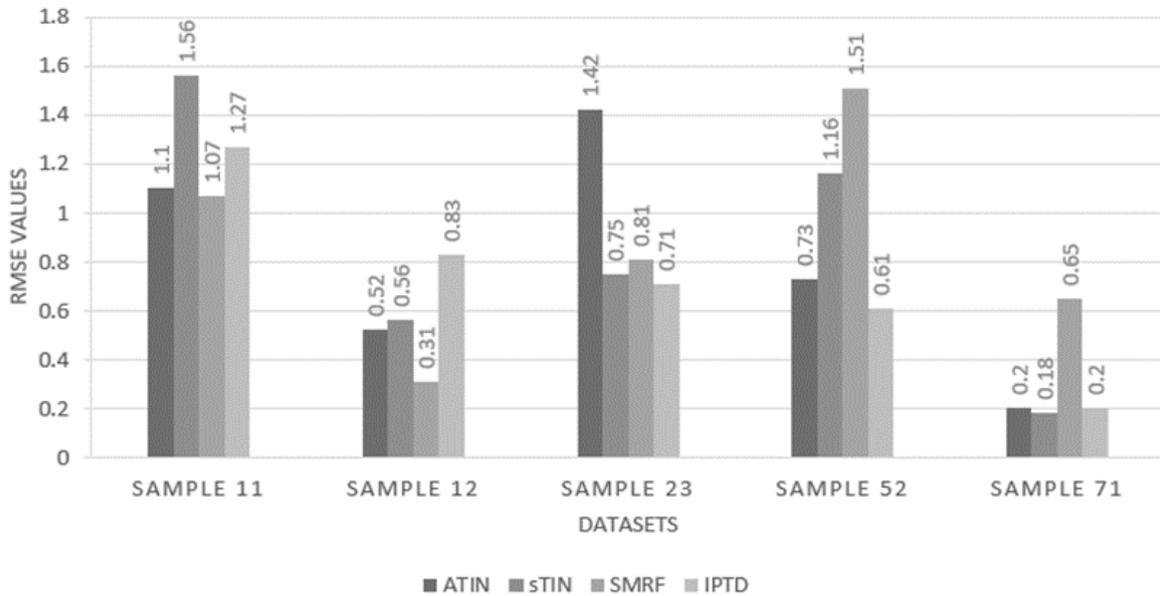
**Figure 9**

*DTMs produced for SAMPLE52 using filtered data. Visible filtering errors are shown in the white boxes. DTM from reference data are shown in (a) and DTMs from test filters are shown in (b) ATIN, (c) SMRF, (d) IPTD, (e) sTIN*

As can be seen in Figure 11, among the dataset, the most accurate filtering results were obtained from Sample71. This dataset's primary advantage is that it lacks a thick covering of vegetation. There is an overpass and a few buildings that need to be removed. ATIN, IPTD, and sTIN performed well with this dataset and resulted in smaller RMSEs than the SMRF algorithm. The ATIN and SMRF were not able to identify only small vegetation on the edge of the road. sTIN successfully filtered the building near the overpass, but IPTD and sTIN failed to remove the building on the flat terrain that located in the north-east area of the dataset, as can be seen in Figure 10. The total error rates of ATIN, sTIN, SMRF and IPTD for this dataset are 1.63%, 4.3%, 1.65% and 2%, respectively (Table 3).



**Figure 10**  
DTMs generated from filtered data for SAMPLE71. Visible filtering errors are shown in the white boxes. DTM from reference data are shown in (a) and DTMs from test filters are shown in (b) ATIN, (c) SMRF, (d) IPTD, (e) sTIN



**Figure 11**  
RMSE values (m) of DTMs for each filtering algorithm respectively

**Table 3**  
Accuracy of the sTIN algorithm

		Reference Data					
		Ground	Object	Identified	Error	Total	
Classification results	Sample11	Ground	18211	2390	20601	10.9%	38010
		Object	3574	13835	17409	22%	
	Sample12	Ground	24761	826	25587	3%	52119
		Object	1930	24602	26532	7.5%	
	Sample23	Ground	11515	1189	12704	9%	25095
		Object	1708	10683	12391	14.3%	
	Sample52	Ground	19319	559	19878	2.7%	22474
		Object	792	1804	2596	33.5%	
	Sample71	Ground	13349	151	13500	1%	15645
		Object	526	1619	2145	29.7%	

## RESULTS

The main goal of this study was to show the performance of the classic ATIN algorithm when the TIN construction method changes. For this purpose, the sTIN uses only initial ground points that were selected from the lowest point in each grid for triangulation process instead of using all ground points at each iteration. In addition, the sTIN includes a different approach from ATIN, using fixed parameters to remove outlier LiDAR points. The elimination of outlier points is a crucial pre-process step to generate the better initial triangulated surface. Nevertheless, the sTIN did not show better filtering results regarding RMSE values than the ATIN algorithm in three out of five datasets. Therefore, the algorithm's processing capacity has been increased and the use of memory has decreased by changing the TIN construction method and setting parameters. Further studies could investigate the utility of this filtering method in larger and density areas. In addition, morphology-based filtering algorithms were proven to be powerful and efficient. However, it is easy to cause misjudgement in protruding terrains [40]. The SMRF resulted in better RMSEs than TIN based algorithms in flat terrain areas. Pingel also reported that the SMRF algorithm is successful even when using a single set of parameters against all samples, suggesting that novice users can achieve good results with it [6].

## CONCLUSIONS

Selection of the filtering parameters has a great influence on the removal of non-ground objects. Future work will address the optimizing and add various parameters for better extraction of bare earth surface, especially in dense forest areas. Many studies were carried out for ground surface extraction from dense forests. For generating terrain model in a forested environment, the choice of first return or last return from LiDAR data is still a matter of debate [45-46]. The functionality of filters using slope factors can be increased by using slope information specific to the area being studied. In triangular-based filters, performance can be improved by selecting the first reference ground point from blocks of appropriate size, thereby reducing the distance between the triangle corners. Consequently, classification errors are reduced and processing time improved.

In recent years, machine learning techniques, especially deep learning, showed quite remarkable performance in 3D point cloud segmentation problems. The big drawback of these learning process is that huge datasets are required to be trained, which brings with it the problem of availability of labeled datasets. From the perspective of success of the learning process, it is essential to use a class-balanced dataset. Despite these disadvantages, the success of these techniques has left classical methods behind. Concepts on this technology can be found in review papers [47-48].

### **Ethical Statement**

This work has been derived from first author's MSc thesis titled "*Hava lidar nokta bulutu verileri filtreleme algoritmalarının geliştirilmesi ve performanslarının karşılaştırılması*" under the supervision of Abdullah VARLIK.

### **Authorship contribution statement**

Research Design (CRediT 1) Fırat URAY (%50) – Abdullah VARLIK (%50)

Data Collection (CRediT 2) Fırat URAY (%70) - Abdullah VARLIK (%30)

Research - Data Analysis - Validation (CRediT 3-4-6-11) Fırat URAY (%70) - Abdullah VARLIK (%30)

Writing the Article (CRediT 12-13) Fırat URAY (%100)

Revision and Improvement of the Text (CRediT 14) Fırat URAY (%50) – Abdullah VARLIK (%50)

### **Conflicts of Interest**

The authors declare no conflict of interest.

## REFERENCES

- [1] M. A. Sayar, H. Z. Selvi & İ. Buğdaycı, Determination of suruç tent city area by analytic hierarchy method, *Necmettin Erbakan University Journal of Science and Engineering*. 1 (2019), 20–31.
- [2] E. Jonuzi, S.S. Durduran & T. Alkan. North Macedonian cadastre towards cadastre 2034, *Necmettin Erbakan University Journal of Science and Engineering*. 4 (2022), 26-44. doi:10.47112/neufmbd.2022.3.
- [3] M. Erdönmez & A. Varlik. Oblique photogrammetry application and accuracy analysis with unmanned aerial vehicles in built areas, *Necmettin Erbakan University Journal of Science and Engineering*. 2 (2020), 1-11. doi:10.47112/neufmbd.2020.1.
- [4] B. Keleş. & S.S. Durduran. In terms of land use and land cover change using remote sensing technique: case of study in Osmaniye city, *Necmettin Erbakan University Journal of Science and Engineering*. 1 (2019), 32-52.
- [5] Z. Lijian, L. Zulong, L. Yingcheng, X. Yanli, L. Ming, W. Zhuolei & L. Xiaolong. Application and Analyses of Airborne Lidar Technology in Topographic Survey of Tidal Flat and Coastal Zone, *In the International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2008, 1–4.
- [6] T.J. Pingel, K.C. Clarke & W.A. McBride. An improved simple morphological filter for the terrain classification of airborne LIDAR data. *ISPRS Journal of Photogrammetry and Remote Sensing*. 77 (2013), 21-30. doi:10.1016/j.isprsjprs.2012.12.002.
- [7] P. Axelsson. Processing of laser scanner data—algorithms and applications. *ISPRS Journal of Photogrammetry and Remote Sensing*. 54(2-3) (1999), 138-147. doi:10.1016/S0924-2716(99)00008-8.
- [8] Axelsson, P. DEM Generation from Laser Scanner Data Using adaptive TIN Models. *International Archives of Photogrammetry and Remote Sensing*. 33 (2000), 110-117. doi:10.1016/j.isprsjprs.2005.10.005.
- [9] K. Kraus & N. Pfeifer. Determination of terrain models in wooded areas with airborne laser scanner data. *ISPRS Journal of Photogrammetry and Remote Sensing*. 53 (1998), 193-203. doi:10.1016/S0924-2716(98)00009-4.
- [10] N. Pfeifer, P. Stadler, & C. Briese. Derivation of Digital Terrain Models in The Scop++ Environment. *OEEPE Workshop on Airborne Laserscanning and Interferometric SAR for Digital Elevation Models*, 2001, 67-80.
- [11] G. Sohn, & I. Dowman. Terrain Surface Reconstruction by the Use of Tetrahedron Model with the Mdl Criterion. *International Archives of Photogrammetry and Remote Sensing, Remote Sensing and Spatial Information Sciences XXXIV (Pt. 3A)*, 2022, 336–344.
- [12] G. Sithole, & G. Vosselman. Experimental comparison of filter algorithms for bare-Earth extraction from airborne laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*. 59, (2004), 85-101. doi:10.1016/j.isprsjprs.2004.05.004.
- [13] X. Meng, N. Currit, & K. Zhao. Ground filtering algorithms for airborne LiDAR data: A review of critical issues. *Remote Sensing*. 2(10) (2010), 833-860. doi:10.3390/rs2030833.
- [14] J. Kilian, N. Haala, & M. Englich. Capture and evaluation of airborne laser scanner data. *International Archives of Photogrammetry and Remote Sensing*. 31 (1996), 383–388.
- [15] K. Zhang, S.C. Chen, D. Whitman, M. L. Shyu, J. Yan, & C. Zhang. A progressive morphological filter for removing nonground measurements from airborne LIDAR data. *IEEE Transactions on Geoscience and Remote Sensing*. 41 (2003), 872-882. doi:10.1109/TGRS.2003.810682.
- [16] Q. Chen, P. Gong, D. Baldocchi, & G. Xie. Filtering airborne laser scanning data with morphological methods. *Photogrammetric Engineering & Remote Sensing*. 2 (2007), 175-185. doi:10.14358/PERS.73.2.175.

- [17] C. Chen, Y. Li, W. Li, & H. Dai. A multiresolution hierarchical classification algorithm for filtering airborne LiDAR data. *ISPRS Journal of Photogrammetry and Remote Sensing*. 82 (2013), 1-9. doi:10.1016/j.isprsjprs.2013.05.001.
- [18] D. Mongus, N. Lukač, & B. Žalik. Ground and building extraction from LiDAR data based on differential morphological profiles and locally fitted surfaces. *ISPRS Journal of Photogrammetry and Remote Sensing*. 93 (2014), 145-156. doi:10.1016/j.isprsjprs.2013.12.002.
- [19] Y. Li, B. Yong, P. van Oosterom, M. Lemmens, H. Wu, L. Ren & J. Zhou. Airborne LiDAR data filtering based on geodesic transformations of mathematical morphology. *Remote Sensing*. 9 (2017), 1104. doi:10.3390/rs9111104.
- [20] G. Vosselman. Slope based filtering of laser altimetry data. *International Archives of Photogrammetry and Remote Sensing*. 33 (2000), Part B3/2, 678-684. doi:10.1016/S0924-2716(98)00009-4.
- [21] J. Susaki. Adaptive slope filtering of airborne lidar data in urban areas for Digital Terrain Model (DTM) generation. *Remote Sensing*. 4(6) (2012), 1804-1819. doi:10.3390/rs4061804.
- [22] X. Meng, L. Wang, J.L. Silván-Cárdenas, & N. Currit. (2009). A multi-directional ground filtering algorithm for airborne LIDAR. *ISPRS Journal of Photogrammetry and Remote Sensing*. 64 (2009), 117-124. doi:10.1016/j.isprsjprs.2008.09.001.
- [23] C.-C. Feng & Z. Guo. Automating parameter learning for classifying terrestrial LiDAR point cloud using 2D land use maps. *Remote Sensing*. 10 (2018), 1192-1214. doi:10.3390/rs10081192.
- [24] J. Zhang, X. Lin & X. Ning. SVM-based classification of segmented airborne LiDAR point clouds in urban areas. *Remote Sensing*. 5(8) (2013), 3749–3775. doi:10.3390/rs5083749.
- [25] J. Lopatin, K. Dolos, H. J. Hernández, M. Galleguillos & F. E. Fassnacht. Comparing generalized linear models and random forest to model vascular plant species richness using LiDAR data in a natural forest in central Chile. *Remote Sensing of Environment*. 173 (2016), 200-210. doi:10.1016/j.rse.2015.11.029.
- [26] J. Niemeyer, F. Rottensteiner & U. Soergel. Contextual classification of LiDAR data and building object detection in urban areas. *ISPRS J. Photogramm. Remote Sensing*. 87 (2014), 152–165. doi:10.1016/j.isprsjprs.2013.11.001.
- [27] R. Shapovalov, A. Velizhev & O. Barinova. Non-associative Markov networks for 3D point cloud classification. *International Archives of Photogrammetry and Remote Sensing, Remote Sensing and Spatial Information Sciences* 2018, 103–108.
- [28] A.B. Jahromi, M.J.V. Zoj, A. Mohammadzadeh, & S. Sadeghian. A novel filtering algorithm for bare-earth extraction from airborne laser scanning data using an artificial neural network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. 4 (2011), 836-843. doi:10.1109/JSTARS.2011.2132793.
- [29] X. Zhao, Q. Guo, Y. Su, & B. Xue. Improved progressive TIN densification filtering algorithm for airborne LiDAR data in forested areas. *ISPRS Journal of Photogrammetry and Remote Sensing*. 117 (2016), 79-91. doi:10.1016/j.isprsjprs.2016.03.016.
- [30] P. Ghamisi, Y. Chen, & X. X. Zhu. A Self-Improving Convolution Neural Network for the Classification of Hyperspectral Data. *IEEE Geoscience and Remote Sensing Letters*. 13 (2016), 1537-1541. doi:10.1109/LGRS.2016.2595108.
- [31] L. Mou, L. Bruzzone, & X. Zhu. X. Learning spectral-spatial features via a recurrent convolutional neural network for change detection in multispectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*. 57 (2019), 924-935. doi:10.1109/TGRS.2018.2863224.
- [32] B. Erol. An automated height transformation using precise geoid models. *Scientific Research and Essays*. 6 (2011), 1351-1363. doi:10.5897/SRE10.1119.
- [33] P. Rashidi, & H. Rastiveis. Ground filtering LiDAR data based on multi-scale analysis of height

- difference threshold. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. 42, (2017), 7–10. doi:10.5194/isprs-archives-XLII-4-W4-225-2017.
- [34] G. Sithole, & G. Vosselman. ISPRS comparison of filters. *ISPRS Commission III*, 2003, 71-78. doi:10.1093/intimm/dxs147.
- [35] J. Zhang, & X. Lin. Filtering airborne LiDAR data by embedding smoothness-constrained segmentation in progressive TIN densification. *ISPRS Journal of Photogrammetry and Remote Sensing*. 81 (2013), 44-59. doi:10.1016/j.isprsjprs.2013.04.001.
- [36] S. Smith, D. Holland, & P. Longley. Investigating the spatial structure of error in digital surface models derived from laser scanning data. *Int. Arch. Photogramm. Remote Sensing*, 2003, 7.
- [37] E.P. Baltsavias. A comparison between photogrammetry and laser scanning. *ISPRS Journal of Photogrammetry & Remote Sensing*. 54 (1999), 83-94. doi:10.1016/S0924-2716(99)00014-3.
- [38] L. Pereira, & G. Gonçalves. (2010). Accuracy of a DTM derived from full-waveform laser scanning data under unstructured eucalypt forest: A case study. *FIG Congress, 2010*, 16.
- [39] S.E. Reutebuch, R.J. McGaughey, H.E. Andersen, & W.W. Carson. Accuracy of a high-resolution lidar terrain model under a conifer forest canopy. *Canadian Journal of Remote Sensing*. 29 (2003) 527-535. doi:10.5589/m05-016.
- [40] R.G. Congalton. A review of assessing the accuracy of classifications of remotely sensed data. *Remote Sensing of Environment*. 37 (1999), 35-46. doi:10.1016/0034-4257(91)90048-B.
- [41] S. Seo, & C.G. O'Hara. Parametric investigation of the performance of lidar filters using different surface contexts. *Photogrammetric Engineering and Remote Sensing*. 74 (2008), 343-362.
- [42] M.E. Hodgson, & P. Bresnahan. Accuracy of Airborne Lidar-Derived Elevation. *Photogrammetric Engineering & Remote Sensing*. 3 (2004), 331-339. doi:10.14358/PERS.70.3.331.
- [43] H. Mitasova, M.F. Overton, J.J. Recalde, D.J. Bernstein, & C.W. Freeman. Raster-Based Analysis of Coastal Terrain Dynamics from Multitemporal Lidar Data. *Journal of Coastal Research*. 252 (2009), 507-514. doi:10.2112/07-0976.1.
- [44] Z. Hui, Y. Hu, Y.Z. Yevenyo, & X. Yu. An improved morphological algorithm for filtering airborne LiDAR point cloud based on multi-level kriging interpolation. *Remote Sensing*. 8(1) (2016), 35. doi:10.3390/rs8010035.
- [45] S.C. Popescu, & K. Zhao. A voxel-based lidar method for estimating crown base height for deciduous and pine trees. *Remote Sensing of Environment*. 112 (2008), 767-781. doi:10.1016/j.rse.2007.06.011.
- [46] K. Zhao, S. Popescu, & R. Nelson. Lidar remote sensing of forest biomass: A scale-invariant estimation approach using airborne lasers. *Remote Sensing of Environment*. 113 (2009), 182-196. doi:10.1016/j.rse.2008.09.009.
- [47] S. A. Bello, S. Yu, C. Wang, J. M. Adam & J. Li. Review: Deep learning on 3D point clouds. *Remote Sensing*. 12 (2020), 1729. doi:10.3390/rs12111729.
- [48] Y. He, H. Yu, X. Liu, Z. Yang, W. Sun, Y. Wang., Q. Fu, Y. Zou & A. Mian. Deep Learning based 3D Segmentation: A Survey. *arXiv*. 2021. doi:10.48550/arXiv.2103.05423.