

Article

Filtering Airborne LiDAR Data Through Complementary Cloth Simulation and Progressive TIN Densification Filters

Shangshu Cai ^{1,2}, Wuming Zhang ^{1,*}, Xinlian Liang ², Peng Wan ¹, Jianbo Qi ¹, Sisi Yu ^{3,4}, Guangjian Yan ¹ and Jie Shao ¹

¹ State Key Laboratory of Remote Sensing Science, Jointly Sponsored by Beijing Normal University and Institute of Remote Sensing and Digital Earth of Chinese Academy of Sciences; Beijing Engineering Research Center for Global Land Remote Sensing Products, Institute of Remote Sensing Science and Engineering, Faculty of Geographical Science, Beijing Normal University, Beijing 100875, China; sscail@mail.bnu.edu.cn (S.C.); wanpeng@mail.bnu.edu.cn (P.W.); qjbj@mail.bnu.edu.cn (J.Q.); gjyan@mail.bnu.edu.cn (G.Y.); shaojie@mail.bnu.edu.cn (J.S.)

² Department of Remote Sensing and Photogrammetry, Finnish Geospatial Research Institute, 02431 Masala, Finland; xinlian.liang@nls.fi

³ Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, Beijing 100101, China; yuss2017@radi.ac.cn

⁴ University of Chinese Academy of Sciences, Beijing 100049, China

* Correspondence: wumingz@mail.bnu.edu.cn; Tel.: +86-10-5880-9246

Received: 18 February 2019; Accepted: 25 April 2019; Published: 1 May 2019



Abstract: Separating point clouds into ground and non-ground points is a preliminary and essential step in various applications of airborne light detection and ranging (LiDAR) data, and many filtering algorithms have been proposed to automatically filter ground points. Among them, the progressive triangulated irregular network (TIN) densification filtering (PTDF) algorithm is widely employed due to its robustness and effectiveness. However, the performance of this algorithm usually depends on the detailed initial terrain and the cautious tuning of parameters to cope with various terrains. Consequently, many approaches have been proposed to provide as much detailed initial terrain as possible. However, most of them require many user-defined parameters. Moreover, these parameters are difficult to determine for users. Recently, the cloth simulation filtering (CSF) algorithm has gradually drawn attention because its parameters are few and easy-to-set. CSF can obtain a fine initial terrain, which simultaneously provides a good foundation for parameter threshold estimation of progressive TIN densification (PTD). However, it easily causes misclassification when further refining the initial terrain. To achieve the complementary advantages of CSF and PTDF, a novel filtering algorithm that combines cloth simulation (CS) and PTD is proposed in this study. In the proposed algorithm, a high-quality initial provisional digital terrain model (DTM) is obtained by CS, and the parameter thresholds of PTD are estimated from the initial provisional DTM based on statistical analysis theory. Finally, PTD with adaptive parameter thresholds is used to refine the initial provisional DTM. These contributions of the implementation details achieve accuracy enhancement and resilience to parameter tuning. The experimental results indicate that the proposed algorithm improves performance over their direct predecessors. Furthermore, compared with the publicized improved PTDF algorithms, our algorithm is not only superior in accuracy but also practicality. The fact that the proposed algorithm is of high accuracy and easy-to-use is desirable for users.

Keywords: airborne LiDAR data; progressive TIN densification; cloth simulation; filtering

1. Introduction

During previous decades, airborne light detection and ranging (LiDAR) technology, which is an efficient and rapid remote sensing technology for collecting three-dimensional (3D) point clouds over a large area, has been widely employed in various fields, such as digital terrain model (DTM) generation [1–5], forest ecosystem investigation [6–12], and 3D building modeling [13–16]. In most applications, a filtering operation for separating point clouds into ground and non-ground points is a preliminary and essential step. Many filtering algorithms have been proposed to automatically filter ground points, and these algorithms can be mainly grouped into three categories, namely, slope-based methods [17–21], mathematical morphology-based methods [22–27], and surface-based methods [28–44]. An experimental comparison of eight filtering algorithms was performed by Sithole and Vosselman [2]. They concluded that the surface-based methods generally performed better than other filtering methods because they used more context information. The basic principle of surface-based methods is to gradually approximate the bare earth using a parametric surface, such as triangulated irregular network (TIN) model, weighted linear least-squares interpolation model, active shape model, thin plate spline (TPS), and cloth simulation model. For this type of method, the ground seed points are obtained and then densified iteratively to create a provisional DTM that gradually refines the ground surface based on certain criteria (e.g., elevation). Among the surface-based methods, progressive TIN densification filtering (PTDF) is one of the typical methods that construct the initial TIN-based DTM (i.e., the initial provisional DTM) from ground seed points, which are the lowest points in each grid cell of an entire region dataset. The grid cell size (c) is usually larger than the size of the maximum non-ground object to minimize the influence of non-ground points [33,36]. Then, remaining ground points are progressively detected from the unfiltered points to update the initial provisional DTM based on elevation and angle criteria [40,41]. The filter has been widely used in the scientific community and the engineering field because of its promising performance [37,38,42,45]. Moreover, it has been integrated into multiple software packages, such as Terrasolid, Lastools, and ALDPAT [37,45,46].

However, the performance of PTDF usually depends on the detailed initial terrain and the cautious tuning of parameters to cope with various terrains [22,37–43]. Specifically, when the initial terrain is not detailed enough, this algorithm usually degrades rugged terrain characteristics (e.g., mountain peaks) for mountainous areas, which is the classical cut-off problem [47]. An example of this “over filtering” can be observed in Figure 1. The heavy bias towards type I error (the percentage of ground points incorrectly classified as non-ground points) may result in sparse ground points, failing to describe terrain morphology. Therefore, in the application of PTDF, one of the key factors for satisfactory results is to acquire more and evenly distributed ground seed points to generate a detailed initial terrain. Improvements have been proposed to enhance the effectiveness and robustness of PTDF. Zhang and Lin [37,38] used smoothness-constrained segmentation to expand the set of ground seed points. This improvement can effectively reduce type I error. However, it needs several suitable parameters to achieve satisfactory results. Chen et al. [39] employed two specific characteristics of triangles (i.e., length and slope changes of the triangle) to detect additional seeds at the ridge and used confidence interval estimation theory to eliminate erroneous ground seed points. This algorithm effectively improves filtering accuracy at sharp ridges, whereas its performance, which was only confirmed on a point cloud obtained from aerial digital photogrammetry, remains uncertain for airborne LiDAR point cloud. Zhao et al. [40] detected ground seed points using the morphological opening operation, wherein the result is sensitive to the selection of the optimal window size. To solve this issue, Liu et al. [41] used a progressive morphological filter to select ground seed points. Although the morphology-based method is easy to achieve, its performance intensively relies on a set of parameters, particularly for the maximum window size [22]. To improve the extraction accuracy of ground seed points, two studies [40,41] used the feature (i.e., ground points belonging to planes) to remove error points from ground seed points. However, this approach further increases the number of parameters. In addition, Nie et al. [42] used the Douglas-Peuker principle to densify ground seed

points under the initial terrain. However, the method cannot reduce the type I error considerably because the number of ground seed points above the initial terrain is not increased. To summarize, although the aforementioned improvements have been proven successful, most algorithms require many user-defined parameters when densifying ground seed points. Moreover, these parameters are sensitive to particular scenes.

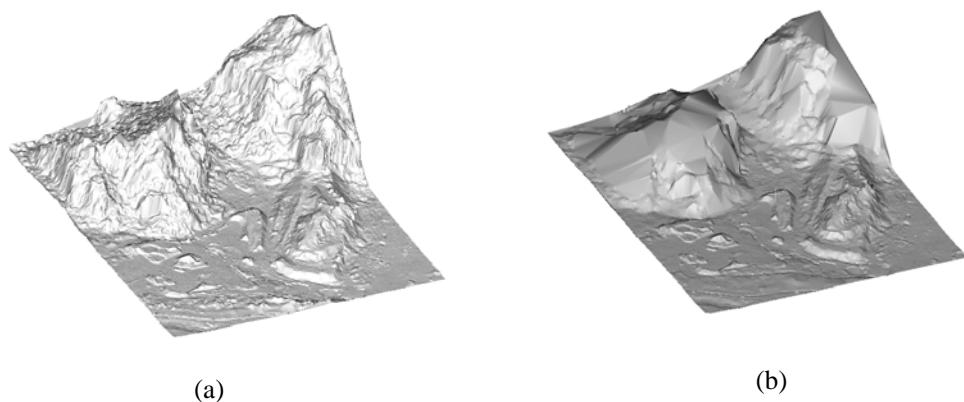


Figure 1. Most ground points on discontinuities and hilltops are not preserved (they are misclassified as “off-terrain”) by progressive TIN densification filtering (PTDF) because of the coarse initial terrain.
 (a) Reference digital terrain model (DTM), (b) DTM generated from PTDF.

Recently, a new surface-based method called cloth simulation filtering (CSF) algorithm has begun receiving attention because its user-defined parameters are few and easy to set [44,48,49]. CSF approximates the initial terrain with simulated cloth (CS) and then adds ground points from the unfiltered points to the initial terrain by using elevation information only, based on the assumption that the actual terrain should be close to a horizontal plane within a certain local area [17,34,39]. This algorithm, which only needs an integer parameter, can obtain an initial terrain that is near the actual terrain. However, unfiltered points are easily misclassified during the refinement of the initial terrain for steep mountainous areas, because the feature differences for ground and non-ground objects on steep surfaces are significantly different from flat surfaces. Sithole and Vosselman [2] suggested using additional contexts to solve this problem. In this sense, PTDF is one of the best implementations [45].

In addition, the challenge with PTDF is that the cautious tuning of parameters is necessary for various landforms. To overcome this limitation, several solutions for parameter threshold adaption have been proposed to improve the automation of PTDF. Axelsson [36] first proposed a statistical analysis-based method for adaptive determination of parameter thresholds. The method used characteristic values of statistics from ground points as the parameter thresholds. Experimental results show that this method can make PTDF achieve promising filtering performance in most types of landscape. However, the details of this method have not been reported. The possible reason is that the academic practitioner tends to maintain work proprietary [37]. Shi et al. [43] used kriging method to predict terrain slope and proposed a parameter threshold estimation formula by analyzing the connection between the terrain slope and the parameters of progressive TIN densification (PTD). However, the method may fail to provide satisfactory results in dense forest areas, because the kriging method cannot accurately predict the terrain slope when the ground seed point is sparse [5]. As mentioned above, estimated parameter thresholds are usually derived from the ground points. In addition, in order to obtain accurate thresholds, it is necessary to acquire as many ground points as possible before PTD.

To summarize, PTDF can acquire a filtering result with high precision, even in steep mountainous areas. The problem is that it usually relies on the detailed initial terrain and suitable parameter thresholds. CSF can obtain a high-quality initial terrain, which also provides a good foundation

for parameter threshold estimation of PTD. However, misclassification can occur easily during the refinement of the initial terrain with respect to steep mountainous areas. To make use of the advantages of PTDF and CSF, this study proposes a novel algorithm to differentiate between ground and non-ground points from airborne LiDAR point clouds in a high-precision and easy-to-use manner by combining cloth simulation (CS) and progressive TIN densification (PTD).

2. Test Data

Two airborne LiDAR point clouds namely ISPRS (International Society for Photogrammetry and Remote Sensing) data and dense point cloud were used to test the performance of the proposed algorithm. ISPRS data, which were collected with an Optech ALTM scanner, include eight sites (named as Site 1–8) consisting of different environments: four urban sites and four rural sites. Fifteen reference samples with different terrain features and land-cover types were selected from Site 1–7 to evaluate the filtering accuracy, as listed in Table 1. Site 8 was excluded due to lack of reference data. The point density is 0.4–1 points/m² for urban sites and 0.08–0.25 points/m² for rural sites, respectively. For each sample, the reference data was manually generated with prior knowledge and available aerial imagery [2].

Table 1. Characteristics of each sample.

Environment	Site	Sample	Features	Reference (Points)	
				Ground	Non-Ground
Urban	1	S11	Mixture of vegetation and buildings on hillside	21,786	16,224
		S12	Mixture of vegetation and buildings	26,691	25,428
	2	S21	Road with bridge	10,085	2875
		S22	Irregularly shaped buildings and bridge	22,504	10,202
		S23	Large, irregularly shaped buildings	13,223	11,872
		S24	Steep slopes	5434	2058
	3	S31	Complex buildings	15,556	13,306
	4	S41	Data gaps, irregularly shaped buildings	5602	5629
		S42	Railway station with trains	12,443	30,027
Rural	5	S51	Vegetation on hillside	13,950	3895
		S52	Steep, terraced slopes	20,112	2362
		S53	Steep, terraced slopes	32,989	1389
		S54	Dense buildings	3983	4625
	6	S61	Data gaps, discontinuity	33,854	1206
	7	S71	Underpass and bridge	13,875	1770

Since the ISPRS data were collected almost two decades ago, the average point density was relatively low ranging from 0.08–1 points/m². Recent advances in LiDAR technology allow continuous improvement of point density. To further test the performance of the proposed algorithm on high density of point cloud, data with an average point density of 31.12 points/m² were used. The data were obtained in 2018 using an unmanned aerial vehicle laser scanning system (SZT-R250) flying about 50 m above the ground level. The data were obtained over 35 ha, 10,912,743 points in total (including 1,749,516 ground points and 9,163,227 non-ground points), and elevations ranging from 0 m to 398.82 m, including the steep slopes, vegetation on hillside and discontinuities which are all typical features of the terrain in mountainous regions, as shown in Figure 2. We produced the reference data using a widely adopted method, which classifies point clouds into ground points and non-ground points by automatic classification tool in Terrasolid, and then the filtering results were further improved manually, through a repeated process in which the results were checked and rectified to ensure accuracy [50–52]. Although the classification results of the reference data are not completely accurate, the misclassification in reference data is much lower than that of the automatic filtering

results [53]. Thus, in this study, we assume that the reference data is a correct classification of the LiDAR data.

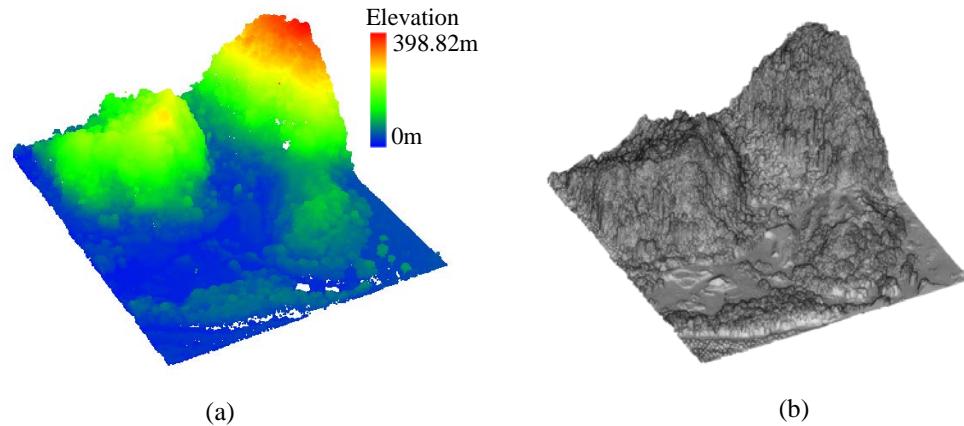


Figure 2. Dense point cloud. (a) Original point cloud, (b) DSM.

3. Methods

A series of critical improvements are implemented to increase the accuracy and practicality of CSF and PTDF in this paper. First, we use CS to select the ground seed points instead of using the lowest points in a user-defined grid cell. This algorithm provides more ground seed points that are almost evenly distributed in general, thereby generating a high-quality initial provisional DTM. Second, parameter thresholds (i.e., the maximum angle (θ) and the maximal terrain slope (s)) for PTD are derived from the initial provisional DTM based on the statistical analysis theory to improve the automation of the filter. Third, the initial provisional DTM is refined by PTD with adaptive parameter thresholds. The entire workflow is composed of three parts, i.e., generation of initial provisional DTM, estimation of parameter thresholds and refinement of the initial provisional DTM, as shown in Figure 3.

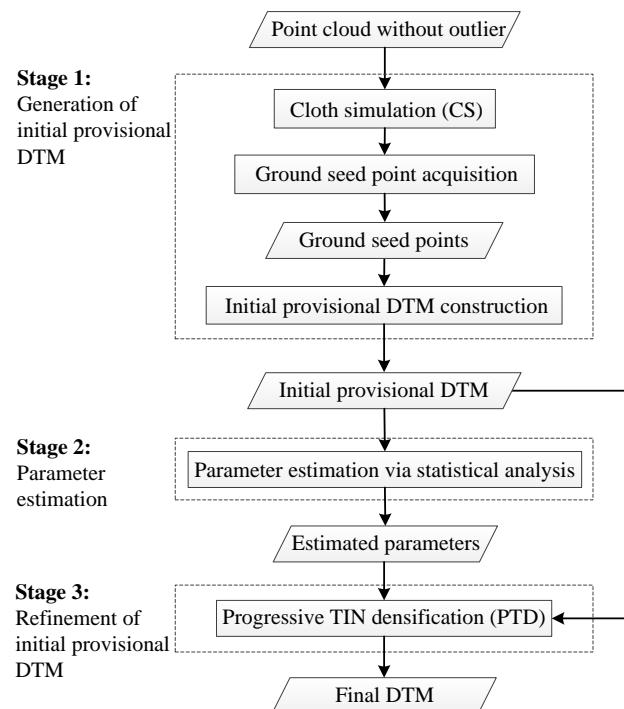


Figure 3. Flowchart of the proposed algorithm.

3.1. Generation of Initial Provisional DTM Based on Cloth Simulation

3.1.1. Cloth Simulation

This method simulates the physical process of cloth-touching objects. After a cloth drops on an inverted (upside-down) landscape, the final shape of the cloth can be confirmed and regarded as an approximated terrain, as shown in Figure 4.

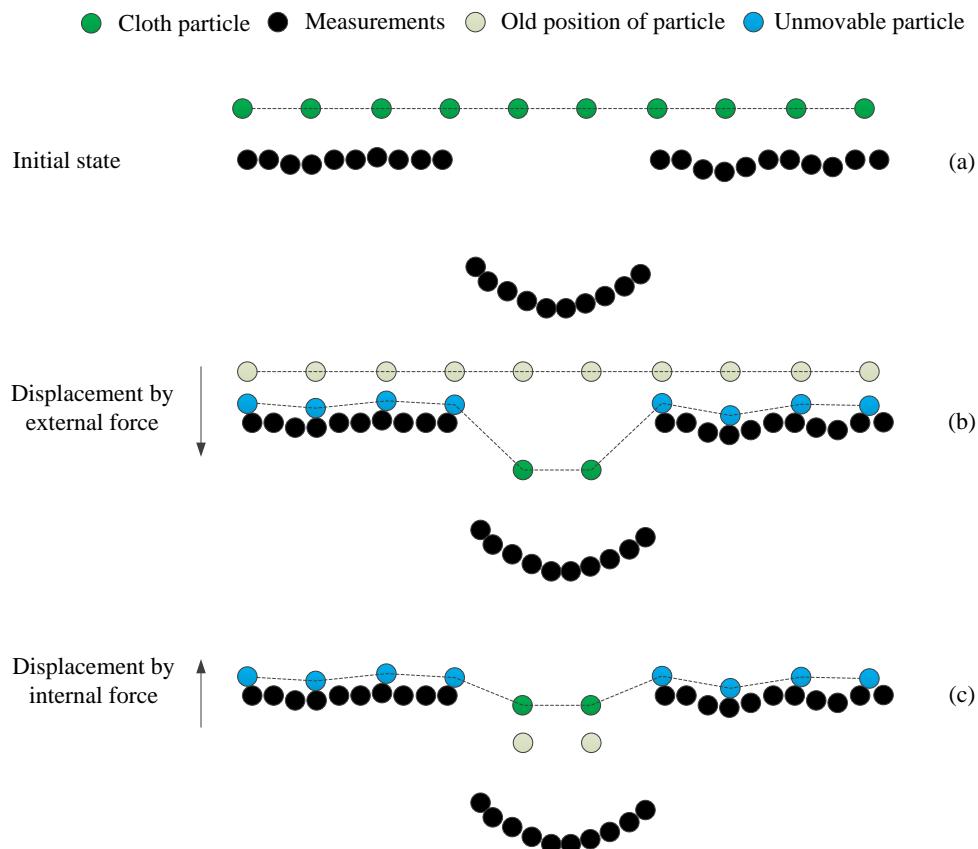


Figure 4. Procedure of cloth simulation (CS): (a) Initial state. A cloth is placed above the inverted measurements. (b) Each particle drops to the inverted measurements under the external force, and their displacement is computed based on the second law of Newton. If the cloth particles collide with the measurements, these particles will stop falling and be labeled unmovable particles. (c) Except for unmovable particles, each particle is moved according to the internal force produced by neighboring particles. Steps (b) and (c) are repeated until the maximum height variation of all particles is sufficiently small.

The process consists of an external force operation followed by an internal force operation. According to the second law of Newton, the relationship between position and forces is decided by Equation (1).

$$m \frac{\partial X(t)}{\partial t^2} = F_{\text{ext}}(X, t) + F_{\text{int}}(X, t) \quad (1)$$

where m represents the mass of the cloth particle, X is the position of a cloth particle at time t , $F_{\text{ext}}(X, t)$ is the external force, that is, gravity, and $F_{\text{int}}(X, t)$ is the internal force that is produced by interconnections between particles.

When the cloth drops to the inverted measurement surface, the displacement of each cloth particle only under the external force is computed by Equation (2).

$$X(t + \Delta t) = 2X(t) - X(t - \Delta t) + g\Delta t^2 \quad (2)$$

where Δt represents the time step, and g is the acceleration. If the cloth particles collide with the inverted surface in the direction of its movement, these particles will stop moving and be labeled unmovable particles.

After cloth particles are moved by an external force, an internal force is exerted to constrain the particles' displacement in the void areas of the inverted surface. The displacement (vector) of each particle is calculated as follows:

$$\vec{d} = \sum_{RT=1}^3 \left(\frac{1}{2}\right)^{RT} b (\vec{p}_i - \vec{p}_0) \Delta \vec{n} \quad (3)$$

where \vec{d} is the displacement vector of a particle, b is zero when the particle is unmovable, otherwise it equals to one, \vec{p}_0 represents the position of the current particle, \vec{p}_i represents the position of the neighboring particle that connects with p_0 , and \vec{n} represents a normalized vector that points to the vertical direction, $\vec{n} = (0, 0, 1)^T$. RT is the repeated time that controls the rigidness of the cloth. As shown in Figure 5, when RT is set to one, the movable particle will move $1/2$ VD between the two particles. When RT is set to two, the movable particle will move $3/4$ VD. When RT is set to three, the movable particle will move $7/8$ VD. When the terrain is extremely steep, RT is specified as a small value (RT = 1). When the terrain is relatively steep, RT is set to a medium value (RT = 2). When handling a flat terrain, a large value (RT = 3) must be set.

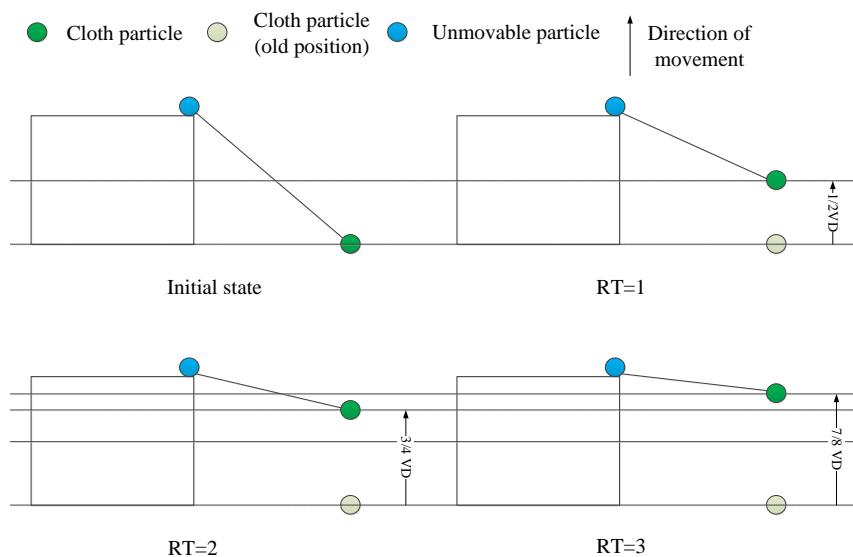


Figure 5. Parameterization of rigidness.

3.1.2. Ground Seed Point Acquisition

When the maximum height variation of all cloth particles is sufficiently small, the cloth is regarded as stationary, and the CS (cloth simulation) process is terminated. The measurements that collide with cloth particles are labeled as ground seed points. The process is presented in Figure 6. The nearest measurement for each cloth particle in the X-Y plane is initially searched. If the heights of the cloth particle and its nearest measurement are equal, then this measurement is considered to collide with the corresponding cloth particle and labeled as a ground seed point. Figure 7 illustrates the labeled results by CS.

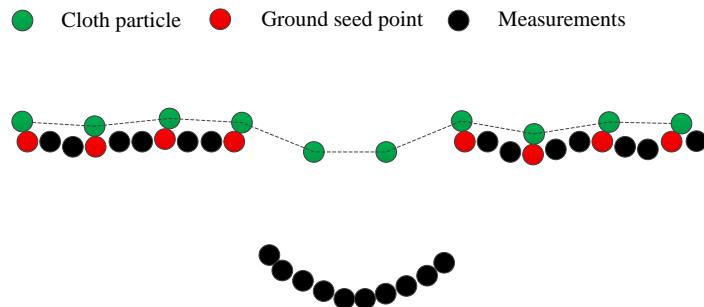


Figure 6. Extraction of ground seed points by collision detection.

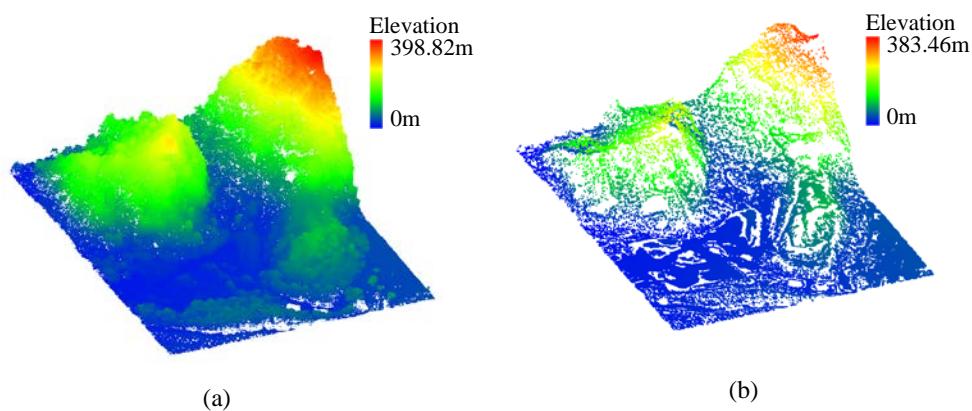


Figure 7. Selection result of ground seed points. (a) Original point cloud, (b) ground seed points.

3.1.3. Initial Provisional DTM Construction

After ground seed points are acquired, the initial provisional DTM is constructed through Delaunay triangulating, as shown in Figure 8. Note that to ensure that each unfiltered point is located in this DTM, the four corners of the study areas are simulated as ground seed points whose heights are equal to the heights of their nearest ground seed points on the X-Y plane [37,38].

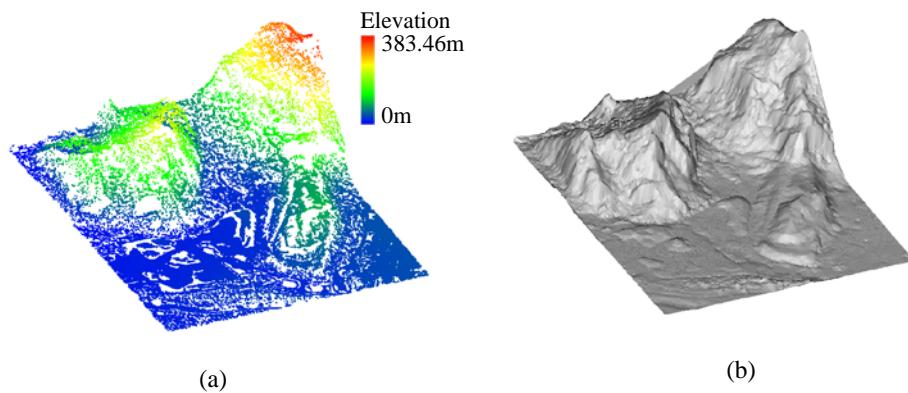


Figure 8. Construction result of initial provisional DTM. (a) Ground seed points, (b) initial provisional DTM.

3.2. Parameter Threshold Estimation Based on Statistical Analysis

The performance of PTD greatly relies on three key parameters, including: the maximum angle (θ), the maximum distance (d) and the maximal terrain slope (s), as shown in Figure 9. The maximum angle (θ) is the maximum angle between the TIN facet and the line that links the candidate point to the facet's closest vertex. When the threshold of the maximum angle (θ) is set to a relatively large value, these non-ground points on the lower objects may be mistakenly classified as ground

points. If the threshold of the maximum angle (θ) is small, many ground points related to key terrain characteristic may be mistakenly classified as non-ground points. The maximum distance (d) is the maximum distance from the candidate point to the corresponding TIN facet. The function of the maximum distance (d) is to guarantee that when the edge length of TIN facet is relatively large, the non-ground points on the low-large areas objects are not mistakenly classified as ground points [37,42]. The maximum terrain slope (s) is the maximum slope between the horizontal plane and the line that links the candidate point to the facet's lowest vertex. The value of the maximum terrain slope (s) decides whether to perform the mirrored operation, which makes the mirrored position of the candidate point is located symmetrically on the other side of the vertex in the corresponding TIN facet, relative to the original position, on the candidate point to handle the discontinuous surface [36–38,42]. More details are provided in Section 3.3.

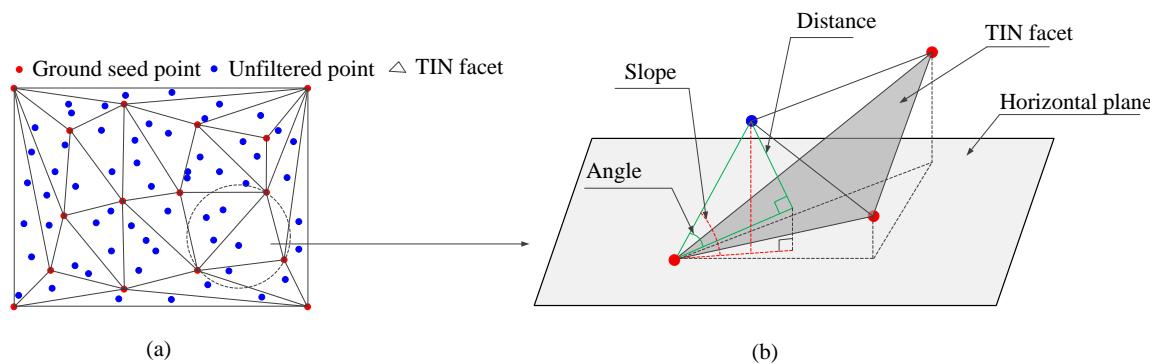


Figure 9. Key parameters in PTD. (a) Initial provisional DTM, (b) key parameters.

In this study, the maximum angle (θ) is determined by statistical analysis, which is inspired by the work by Axelsson [36]. Statistic from initial provisional DTM is collected in the form of a discrete histogram of the terrain slopes, as shown in Figure 10. Finally, θ based on the median value, which is the value separating the higher half from the lower half of a data sample (a probability distribution), is determined from this histogram.

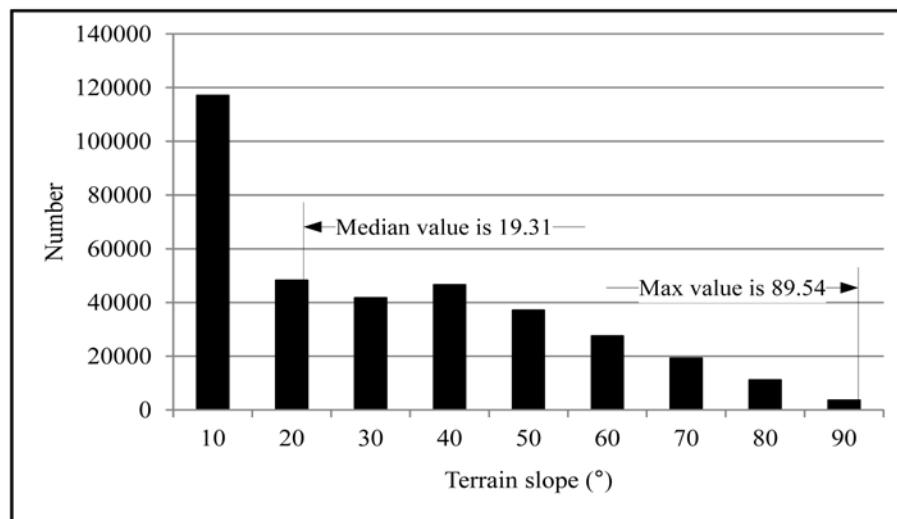


Figure 10. The discrete histogram of terrain slope.

For the maximum distance (d), it is set to the largest elevation difference in the study region. As shown by the example in Figure 11, when the horizontal distance between the non-ground point and the ground seed point is large, d should be set to a suitable threshold to prevent non-ground point from being incorrectly classified as ground point by the maximum angle threshold (θ). In this study,

dense ground seed points generate a relatively short edge length of TIN facet. Thus, d does not work and can be ignored. It usually is set to a rather large value, and we use the largest elevation difference in the study region as a threshold.

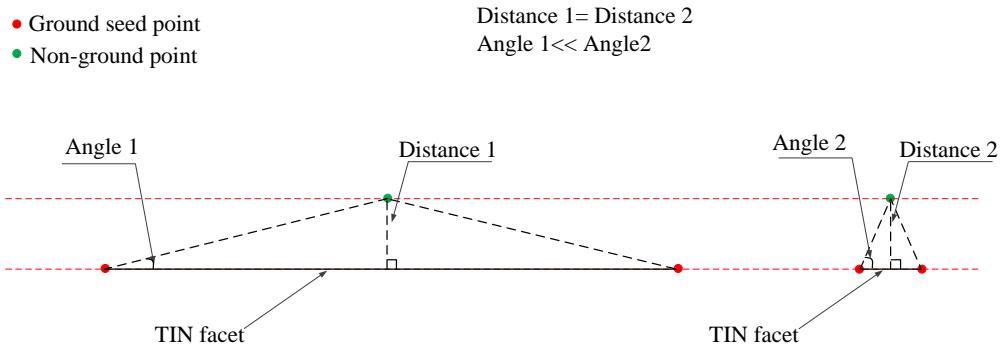


Figure 11. Simple demonstration of the maximum distance threshold setting.

In theory, the maximal terrain slope (s) should be the maximum slope derived from the final DTM. This threshold is usually not available prior to acquirement because ground points are not yet identified [20]. Fortunately, considering the initial provisional DTM is rather close to the actual terrain, we use this provisional DTM to obtain this threshold.

3.3. Refinement of Initial Provisional DTM Based on Progressive TIN Densification

After the initial provisional DTM and key parameters in PTD (progressive TIN densification) are determined, unfiltered points (including ground and non-ground points) are iteratively classified using PTD, and the detected ground points are used to update the initial provisional DTM. The procedure contains six core steps as follows.

Step1: Find the corresponding TIN facet of an unfiltered point.

Step2: Calculate the slope of this TIN facet. If the slope is greater than the maximal terrain slope (s), the unfiltered point will be replaced by the mirrored point in Step 3 to preserve discontinuous terrain. Then, go to Step 4. Otherwise, the unfiltered point will be judged directly in Step 4.

Step3: The mirrored operation is illustrated in Figure 12. Find the vertex $P_{vertex}(x_v, y_v, z_v)$ with the highest elevation value in the corresponding TIN facet of an unfiltered point $P_{unfiltered}(x_u, y_u, z_u)$. The mirrored point $P_{mirrored}(x_m, y_m, z_m)$ of $P_{unfiltered}(x_u, y_u, z_u)$ is calculated as follows:

$$\begin{cases} x_m = 2x_v - x_u \\ y_m = 2y_v - y_u \\ z_m = z_u \end{cases} \quad (4)$$

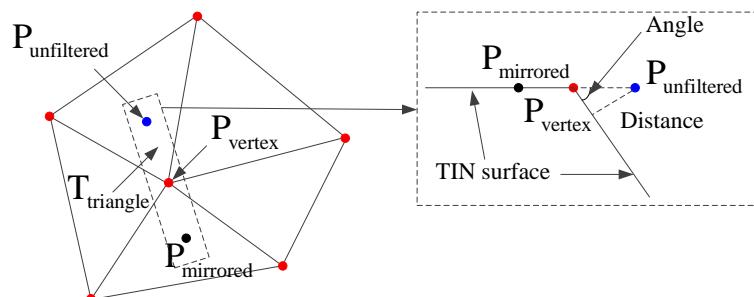


Figure 12. Mirrored operation.

Step4: Calculate the distance to the TIN facet and the angle to the vertex. If both computed values are lower than the maximum distance (d) and the maximum angle (θ), then the unfiltered point is labeled as a ground point.

Step5: Calculate edge length of the TIN facet in the X-Y plane. If the ratio between the maximum edge length and the minimum edge length is smaller than the length ratio threshold (it is set to four), then the ground point is added to update the initial provisional DTM. Note that the introduction of length ratio threshold helps to avoid long narrow triangles, which is low credibility in terrain expression [40].

Step6: Repeat Steps 1–5 until all unfiltered points have been judged. Figure 13 illustrates the final DTM.

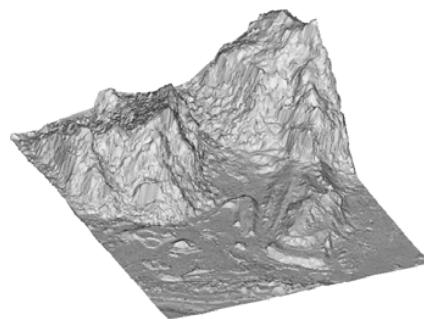


Figure 13. Final DTM.

3.4. Accuracy Indexes

Three accuracy indexes (i.e., type I, type II, and total errors) were used to assess the accuracy of the proposed algorithm quantitatively. The type I error is the percentage of ground points incorrectly classified as non-ground points. The type II error is the percentage of non-ground points incorrectly classified as ground points. The total error is the percentage of all mistakenly classified points. These indexes are calculated as follows [2]:

$$\left\{ \begin{array}{l} \text{Type I} = \frac{a}{c} \\ \text{Type II} = \frac{b}{d} \\ \text{Total} = \frac{a+b}{c+d} \end{array} \right. \quad (5)$$

where a is the number of reference ground points incorrectly classified as non-ground points, b is the number of reference non-ground points incorrectly classified as ground points, and c and d are the numbers of reference ground and non-ground points, respectively.

4. Experiments

4.1. Testing with ISPRS Dataset

The distance between neighboring cloth particles, which is defined as cloth resolution (CR), has a strong relationship with the number of ground seed points. Generally, the smaller the CR value, the more ground seed points are obtained, and vice versa. Meanwhile, the parameter affects the time of cloth simulation because it decides how many cloth particles are generated and modified for the dataset. In CSF [44], CR is set to 0.5 m based on the principle of optimal accuracy. To objectively compare the proposed algorithm and CSF, the proposed algorithm under CR = 0.5 m was first tested. Then, the proposed algorithm under CR = 1 m was tested based on the balance between processing speed and filtering accuracy, which is discussed in Section 5.2. In addition, optimally tuned thresholds were used to compare our results with optimally tuned results. Specifically, we used a back-calibration optimization process based on the ground reference data provided by ISPRS to find the lowest total error of each sample. We tested the maximum angle threshold with 5° increments from 5° to 30°. The result

with the lowest total error identifies the provisional optimized threshold (POT). POT was further refined with 1° increments from POT- 2° to POT+ 2° for obtaining the final optimized threshold (FOT). For example, if POT equals to 15° , FOT will be selected from 13° , 14° , 15° , 16° and 17° , and the result with the lowest total error identifies the FOT. Similarly, we tested the maximum distance threshold with 0.5 m increments from 0.5 m to 1.5 m. POT was further refined with 0.1 m increments from POT-0.2 m to POT + 0.2 m. We also tested the maximal terrain slope threshold with 10° increments from 60° to 90° . POT was further refined with 1° increments from POT- 2° to POT+ 2° .

Figure 14 shows the accuracies of the proposed algorithm under CR = 0.5 m against other 31 algorithms [2,24–27,31–34,37,38,41–44,47,54–61]. The average total error of the proposed algorithm under CR = 0.5 m provides competitive performance, below four filtering algorithms, but above 27 other algorithms. Four filtering algorithms with higher accuracy are discussed in Section 5.3.

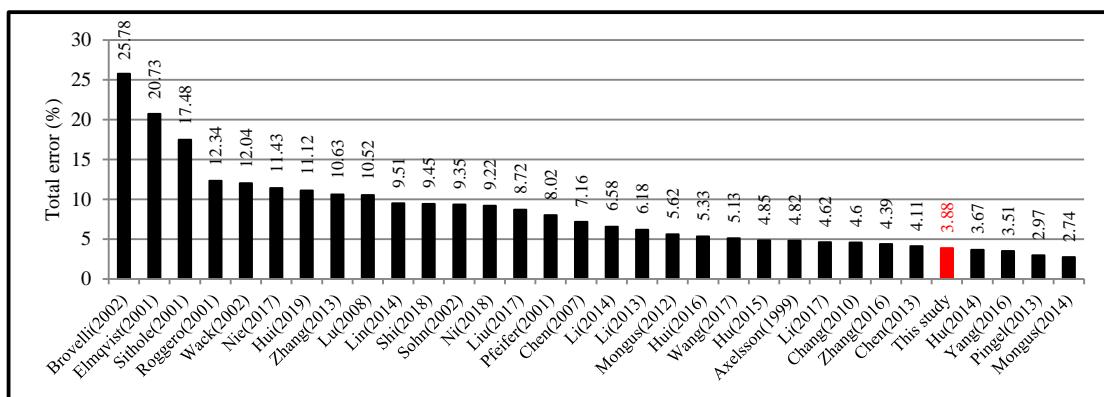


Figure 14. Total error average on 15 samples for 32 filters, in which CR value in the proposed algorithm is set to 0.5 m. The proposed algorithm is marked in red.

As shown in Figure 15, the average total error of the proposed algorithm under CR = 1 m is in the middle of all algorithms. However, the time consumption of the proposed algorithm under CR = 1 m is only about 0.25 times against CR = 0.5 m. We compared the proposed algorithm under CR = 1 m with all publicized improved PTDF algorithms, which completely used the 15 reference samples of the ISPRS dataset to assess their performances in terms of average total error. All parameters in the first three publicized improved PTDF algorithms were determined by the developers' experienced judgment for a particular terrain type, which is somewhat subjective. However, in the proposed algorithm, these parameters were automatically extracted from the initial provisional DTM. The detailed values of parameters in these algorithms are listed in Table 2. Note that the two algorithms proposed by Zhang and Lin [37], Lin and Zhang [38] share the same parameter values.

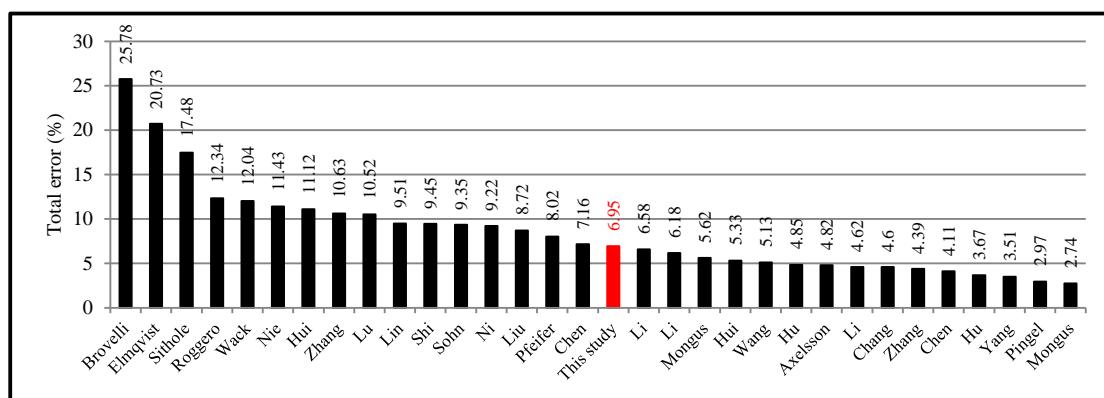


Figure 15. Total error average on 15 samples for 32 filters, in which CR value in the proposed algorithm is set to 1 m. The proposed algorithm is marked in red.

Table 2. The values of parameters in all filters for each sample. Note that “c”, “θ”, “d” and “s” represent “grid cell size”, “maximum angle”, “maximum distance” and “maximal terrain slope”, respectively.

Samples	Zhang and Lin				Nie et al.				The Proposed Algorithm			
	c(m)	θ(°)	d(m)	s(°)	c(m)	θ(°)	d(m)	s(°)	c(m)	θ(°)	d(m)	s(°)
S11	20	6	1.4	80	20	15	1.4	80	1	23.37	>15	88.96
S12	20	6	1.4	80	20	15	1.4	80	1	32.68	>15	85.04
S21	60	6	1.4	88	60	15	1.4	88	1	23.12	>15	62.71
S22	60	6	1.4	88	60	15	1.4	88	1	18.21	>15	88.26
S23	60	6	1.4	88	60	15	1.4	88	1	28.21	>15	85.63
S24	60	6	1.4	88	60	15	1.4	88	1	25.6	>15	63.37
S31	35	6	1.4	88	35	15	1.4	88	1	9.33	>15	84.71
S41	60	6	1.4	88	60	15	1.4	88	1	15.62	>15	62.23
S42	60	6	1.4	88	60	15	1.4	88	1	16.37	>15	70.76
S51	10	6	1	70	10	15	1	70	1	18.75	>15	82.42
S52	10	6	1	70	10	15	1	70	1	22.02	>15	86.37
S53	10	6	1	70	10	15	1	70	1	27.96	>15	85.92
S54	10	6	1	70	10	15	1	70	1	16.06	>15	59.33
S61	40	6	1.4	70	40	15	1.4	70	1	20.64	>15	80.97
S71	20	6	1.4	70	20	15	1.4	70	1	16.38	>15	62.32

Note: In the proposed algorithm, d does not work and is set to a rather large value.

Table 3a shows the total error of the proposed algorithm against four other improved PTDF algorithms. The average total error of the proposed algorithm is 6.95%. Compared to the results of other improved PTDF algorithms, the proposed algorithm achieves the highest precision for 7/15 samples in terms of the total error and is also the best for average values. Seven samples achieving the highest precision are S22, S23, S52, S53, S54, S61, and S71. These samples contain a variety of complex terrain features, including steep slopes, terraced terrain, and discontinuity. The statistical results in Table 3b suggest the average type I and II errors of the proposed algorithm are 4.6% and 11.42%. Compared to the results of other improved PTDF algorithms, the proposed algorithm achieves the highest precision for 11/15 samples in terms of the type I error and is also the best for average values. However, the proposed algorithm is poor in avoiding the type II error. The main reason is more ground seed points will increase the risk of the type II error, while they are controlled to an acceptable level, because the accuracy of ground seed points obtained from CS is extremely high (the average overall precision is 98.39%). Moreover, the inclination to type II error may not be a fatal flaw for the filter strategy, taking into consideration that type II error can be more easily handled by human editing than type I error [2].

Table 3a. Performance comparison between the proposed algorithm and the publicized improved PTDF algorithms in terms of the total error (%). The best results across all the algorithms are marked in red.

Samples	Zhang and Lin	Lin and Zhang	Nie et al.	Shi et al.	Ours
S11	18.49	19.5	18.79	11.12	16.24
S12	5.92	4.78	6.62	7.17	8.85
S21	4.95	6.08	5.60	6.58	14.18
S22	14.18	9.24	14.89	14.02	4.25
S23	12.06	14.43	18.08	17.43	8.52
S24	20.26	5.28	24.57	13.06	15.59
S31	2.32	1.61	2.14	3.13	7.28
S41	20.44	32	27.13	10.06	13.04
S42	3.94	5.95	2.42	1.91	4.75
S51	5.31	4.09	2.85	12.69	3.51
S52	12.98	7.56	14.43	16.67	4.65
S53	5.58	9.9	19.37	9.77	3.95
S54	6.4	10.72	4	4.99	2.58
S61	16.13	6.27	6.89	7.51	0.86
S71	10.44	5.22	3.68	5.68	2.03
Avg.	10.63	9.51	11.43	9.45	6.95

Table 3b. Performance comparison between the proposed algorithm and the publicized improved PTDF algorithms in terms of the type I and II errors (%). The best results across all the algorithms are marked in red.

Samples	Zhang and Lin		Lin and Zhang		Nie et al.		Shi et al.		Ours	
	I	II	I	II	I	II	I	II	I	II
S11	25.67	8.84	26.28	10.4	37.24	1.35	14.74	6.07	7.51	27.98
S12	8.13	3.61	6.56	3.31	11.86	1.05	11.86	1.93	4.68	13.21
S21	1.17	18.23	0.85	24.45	6.2	4.49	7.54	3.18	16.27	6.87
S22	19.05	3.44	6.43	15.44	20.82	3.6	19.44	2.15	2.22	8.73
S23	19.25	4.05	23.21	4.64	35.63	1.6	29.69	3.89	3.48	14.13
S24	22.86	13.41	3.99	8.7	32.58	15.42	15.86	5.63	3.13	48.49
S31	2.1	2.59	0.54	2.59	2.02	2.41	4.61	1.43	12.74	0.91
S41	39.54	1.44	62.22	1.92	52.03	0.32	17.92	2.33	25.56	0.35
S42	9.72	1.55	19.02	0.54	6.69	1.26	3.88	1.1	9.71	2.71
S51	2.05	16.97	2.22	10.81	2.9	2.77	15.14	3.79	0.07	15.81
S52	12.53	16.77	6.46	16.89	16.14	2.96	17.52	9.31	0.98	35.9
S53	4.25	37.22	9.62	16.41	20.22	0.72	10.11	1.77	2.72	33.05
S54	3.59	8.82	3.16	17.23	6.76	1.78	8.65	1.87	1.16	3.81
S61	16.62	2.49	6.26	6.55	8.17	2.07	7.75	0.84	0.39	13.93
S71	10.07	13.39	2.62	25.65	5.24	0.79	6	3.18	0.28	15.71
Avg.	13.11	10.19	11.96	11.06	17.63	2.84	12.71	3.23	4.6	11.42

Figure 16 shows our results and optimally tuned results under CR = 0.5 m and CR = 1 m in terms of average total error. We can observe two characteristics. First, it appears that our results are relatively close to the optimally tuned results. The fact that the proposed algorithm is effective to parameter estimation demonstrates the wide applicability of this algorithm. Second, the difference between our result and optimally tuned result under CR = 0.5 m is lower than the difference under CR = 1 m, and the difference between optimally tuned results under CR = 0.5 m and CR = 1 m is small. It indicates that the detail degree of the initial terrain and the accuracy of parameter estimation have a significant positive correlation relationship.

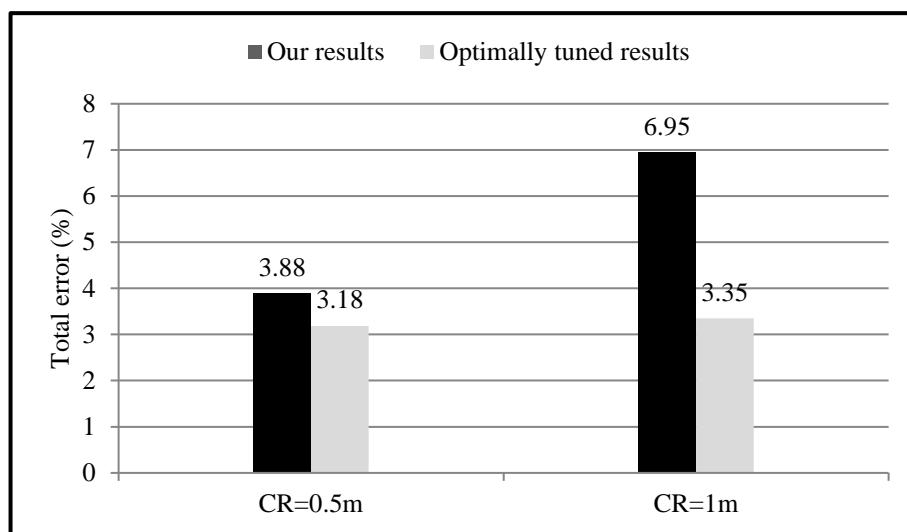


Figure 16. Accuracy comparison of our results and optimally tuned results under different CR values.

Figure 17 shows the DTMs and the space distribution of type I and II errors (SDEs) generated by the proposed algorithm under CR = 0.5 m and CR = 1 m in several representative samples, which are named as DTM_{CR=0.5}, DTM_{CR=1}, SDE_{CR=0.5} and SDE_{CR=1}. S11 has a considerable slope change, and many mixtures of buildings and vegetation are on the hillside. The visualizations of DTM_{CR=0.5} and DTM_{CR=1} exhibit similar appearances. Exploration of further details for some recognizable differences between

them shows $DTM_{CR=0.5}$ is closer to the reference DTM than $DTM_{CR=1}$ in the terrain discontinuities, as shown in the rectangle regions in Figure 17b–d. However, compared to $DTM_{CR=1}$, there are more non-ground objects in $DTM_{CR=0.5}$, as shown in the circle regions in Figure 17b–d. Compared to $SDE_{CR=1}$, $SDE_{CR=0.5}$ shows lower type I error but higher type II error. The reason can be explained by exploring the procedure of CS. When cloth gradually approaches terrain with more cloth particles, a more detailed initial terrain is constructed, thereby assisting PTD to correctly classify the ground point in complex terrain. However, the ground seed points are not 100% correctly obtained by CS. Thus more seed points will increase the risk of type II error. For S52, the ground surface has many abrupt changes. In comparison with the reference DTM, $DTM_{CR=0.5}$ and $DTM_{CR=1}$ can effectively preserve the steep and terraced slopes.

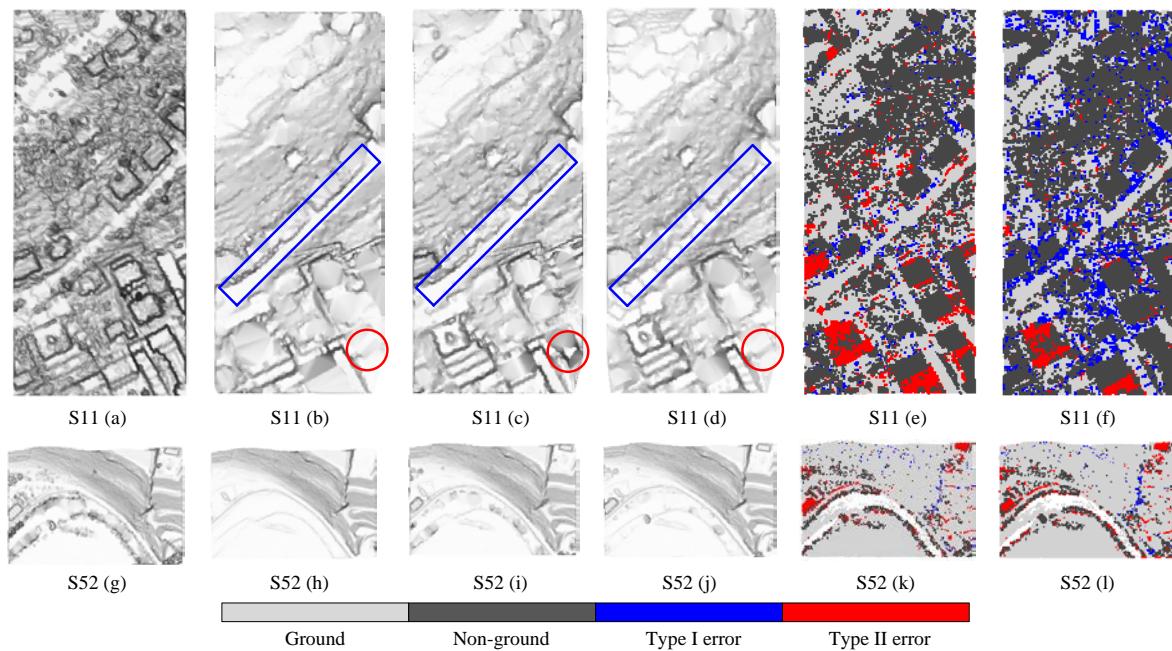


Figure 17. Results of each group (S11 and S52 are selected as representatives). First column: DSMs, second column: DTMs generated from the references, third column: DTMs produced by the proposed algorithm under $CR = 0.5$ m, fourth column: DTMs produced by the proposed algorithm under $CR = 1$ m, fifth column: spatial distributions of the type I and type II errors produced by the proposed algorithm under $CR = 0.5$ m, sixth column: spatial distributions of the type I and type II errors produced by the proposed algorithm under $CR = 1$ m.

4.2. Testing with Dense Point Cloud

To quantitatively assess the advantages of the proposed algorithm under $CR = 1$ m, which achieves the complementary advantages of CSF and PTDF, we compared the three types of errors of the proposed algorithm with CSF and PTDF, where CR value in CSF was set to 1 m. The results are shown in Figure 18. We can observe three characteristics. First, the type I error is obviously larger than the type II error for PTDF, CSF, and the proposed algorithm. The main reason is that the number of non-ground points in this data is more than the ground points [34]. Specifically, non-ground points comprise 83.97% of all the points. Thus, only a few ground points being misclassified as non-ground points will yield a large type I error. Second, in terms of the type I and total errors, the proposed algorithm is better than PTDF. It occurs because CS is capable of providing enough ground seed points evenly distributed in general, which can cover terrain characteristics and retains finer details, especially for hilltops and steep slopes. The initial provisional DTM based on the ground seed points is close to the actual terrain, which is important because the subsequent PTD is based on the initial provisional DTM [40]. Third, compared with CSF, the proposed algorithm performs better in terms of three types

of errors. It indicates that, when refining the initial provisional DTM, PTD outperforms the method that uses elevation information only. This finding is also shared by previous studies, which have illustrated that additional contexts can improve filtering results [2,47].

The generated DTM and the cross-section of point clouds for quality evaluation are shown in Figure 19. In general, the DTM generated from the proposed algorithm shows the closer appearances with the reference DTM, i.e., most of non-ground points are removed and ground points remain. Details of the comparison show that the proposed algorithm can preserve the ground points on hill tops and steep slopes, whereas PTDF and CSF hardly preserve terrain characteristics in areas with large terrain variations.

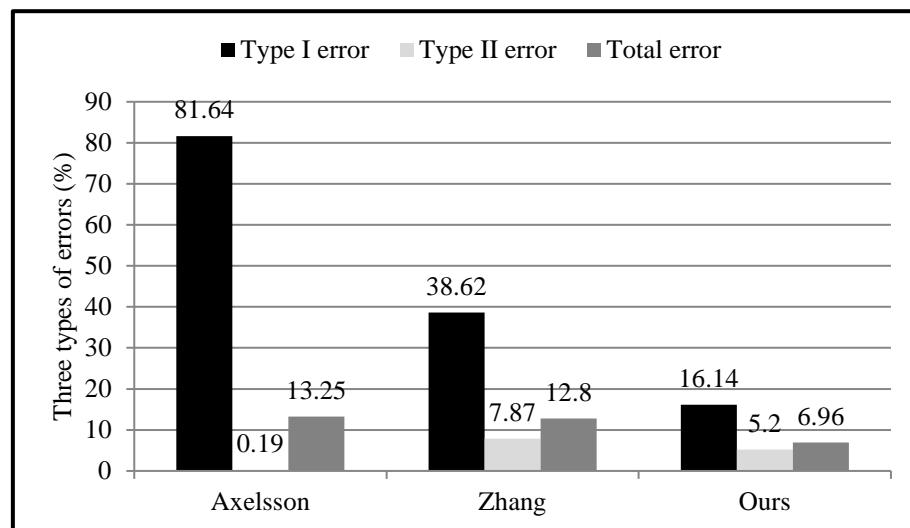


Figure 18. Three types of errors on data with high point density for PTDF, CSF, and our algorithm.

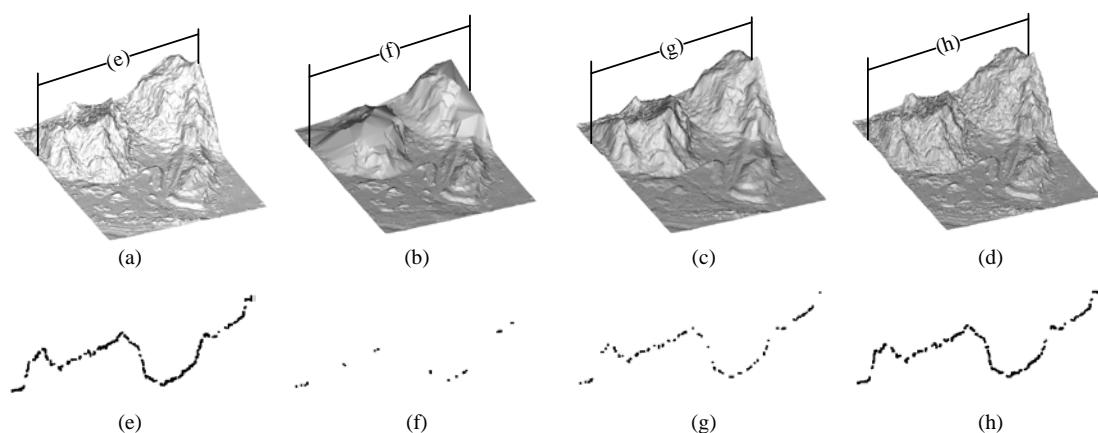


Figure 19. Comparison of DTM generated by PTDF, CSF and our algorithm. (a) Reference DTM, (b) DTM generated from PTDF, (c) DTM generated from CSF, (d) DTM generated from our algorithm. Filtering results for the cross-section for (e) reference data, (f) PTDF, (g) CSF, and (h) our algorithm.

An awareness of the relationship between filtering algorithms and point density helps to balance the processing speed and filtering accuracy [53]. To analyze the effect of point density on the proposed algorithm, the low-density versions of original LiDAR data were generated using a simple point thinning algorithm, keeping only every nth (for example, the 10th) point along the time line from the original LiDAR data [62]. In this study, we used every 2nd, 5th, 10th and 15th points from original LiDAR data and generated a series of low-density versions with average point density of 15.56, 6.22, 3.11 and 2.07 points/m² (named as LD 2nd, LD 5th, LD 10th and LD 15th). As shown in Figure 20, it can

be seen that point density has relatively less effect on the filtering results. Specifically, the root mean square errors (RMSEs) of the type I, type II and total errors are 2.02%, 0.81%, and 0.45%, respectively. Additionally, there is a general tendency for the type II error to be slightly negative. The possible reason is that the effect of the type II error from high-density data is limited because non-ground points are surrounded closely by ground points [63]. In terms of the type I error, the proposed algorithm does not show a recognizable tendency.

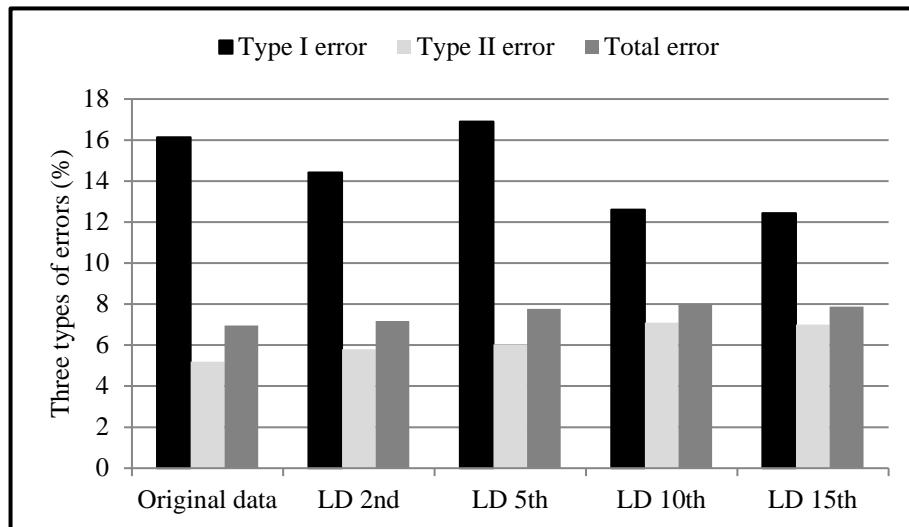


Figure 20. Accuracy comparison of the proposed algorithm under different point density.

5. Discussion

5.1. Accuracy of Ground Seed Points

To explore the effects of ground seed point extraction, the numbers of ground seed points for 15 samples obtained by PTDF and the proposed algorithm under CR = 1 m are listed in Table 4. The results show that, on average, approximately 370 ground seed points for 15 samples are extracted by PTDF. However, the proposed algorithm can provide approximately 10368 ground seed points, which indicates that it can significantly increase the number of ground seed points. In addition, overall precision (OP) is employed to evaluate the quality of ground seed points. OP is the rate of correctly classified ground points in all extracted ground seed points and is calculated as:

$$OP = \frac{a}{a + c} \quad (6)$$

where a is the number of ground points correctly classified, and c is the number of non-ground points incorrectly classified as ground points. The statistics in Table 4 indicate that the accuracy of ground seed points acquired by the proposed algorithm is extremely high, and the average OP is 98.39%. To assess the quality of ground seed point extraction intuitively, S11 is selected as the representative case for demonstration. As shown in Figure 21a,b, the ground seed points provided by the proposed algorithm can cover the terrain features and retain fine details, especially for steep slopes. The detailed changes in the terrain can be presented relatively well, especially for the regions of significant variations in the terrain. Figure 21d shows that the initial provisional DTM constructed from the proposed algorithm is rather near the actual terrain, which is important because the subsequent PTD is based on the initial terrain.

Table 4. Number and OP (overall precision) of ground seed points.

Samples	PTDF (Points)	Ours (Points)	OP (%)
S11	116	12,863	94.75
S12	158	12,135	98.95
S21	10	4600	99.7
S22	20	9973	99.38
S23	16	6391	98.18
S24	10	2554	98.08
S31	29	7687	99.32
S41	10	2310	99.65
S42	20	6377	94.65
S51	1008	10,247	97.3
S52	1329	14,942	98.02
S53	1935	23,544	99.57
S54	517	2727	99.12
S61	146	28,112	99.8
S71	224	11,062	99.39
Avg.	369.87	10,368.27	98.39

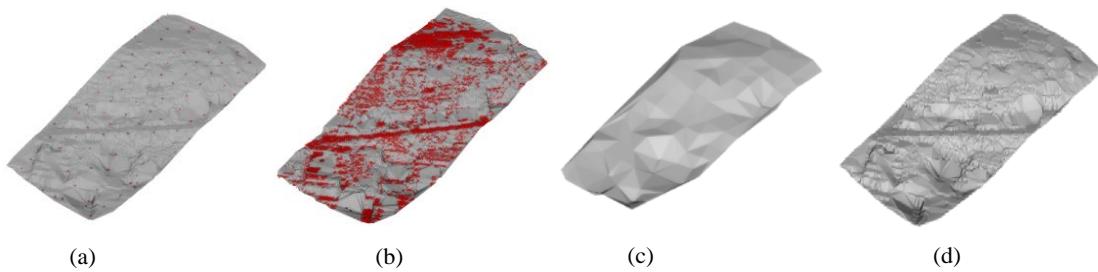


Figure 21. Results of representative sample: (a) Overlay result of reference DTM and ground seed points (red points) produced by PTDF, (b) overlay result of reference DTM and ground seed points (red points) produced by the proposed algorithm, (c) initial provisional DTM generated from ground seed points of (a), (d) initial provisional DTM generated from ground seed points of (b).

5.2. Parameter Analysis

For ground seed points of the 15 reference samples, the simulation time and the total error at different cloth resolution (CR) values (ranging from 0.3 m to 1.5 m) are shown in Figure 22. The results show that a small CR value usually has large time consumption, and the time cost is relatively stable after the CR value exceeds 0.8 m. The accuracies are relatively stable, and the average total error is less than 10% when CR ranges from 0.3 m to 1.1 m, and the accuracies are the highest around 0.6 m. In view of the balance of efficiency and accuracy, the value of this parameter is usually set from 0.8 m to 1.1 m. In terms of the principle of optimal accuracy, the value of this parameter is usually set from 0.5 m to 0.7 m.

After ground seed points are obtained, the remaining ground points are extracted using PTD. To analyze the sensitivity of the parameters (θ and d) on the improvement of filtering accuracy, S11 was selected as a representative in this section. We adjusted the two parameters as follows: $\theta \in \{10^\circ, 15^\circ, 20^\circ, 25^\circ, 30^\circ, 35^\circ, 40^\circ, 45^\circ, 50^\circ, 55^\circ, 60^\circ\}$, $\text{interval}_\theta = 5^\circ$ and $d \in \{0.5\text{m}, 1\text{m}, 1.5\text{m}\}$, $\text{interval}_d = 0.5\text{m}$. For each parameter configuration, the total error was calculated and illustrated in Figure 23. It shows that the total error of the proposed algorithm is lower than PTDF under all parameter configurations, which indicates obtaining more ground seed points is an important way to improve the filtering accuracy of PTDF. This is in agreement with the previous studies that almost all improved PTDF focus on the densification of ground seed points. For PTDF and the proposed algorithm, the RMSE of the total error is 4.78% and 1.51% with the change of θ and d , respectively.

It illustrates that the proposed algorithm is insensitive to the parameter selection even though the algorithm is parameter-dependent.

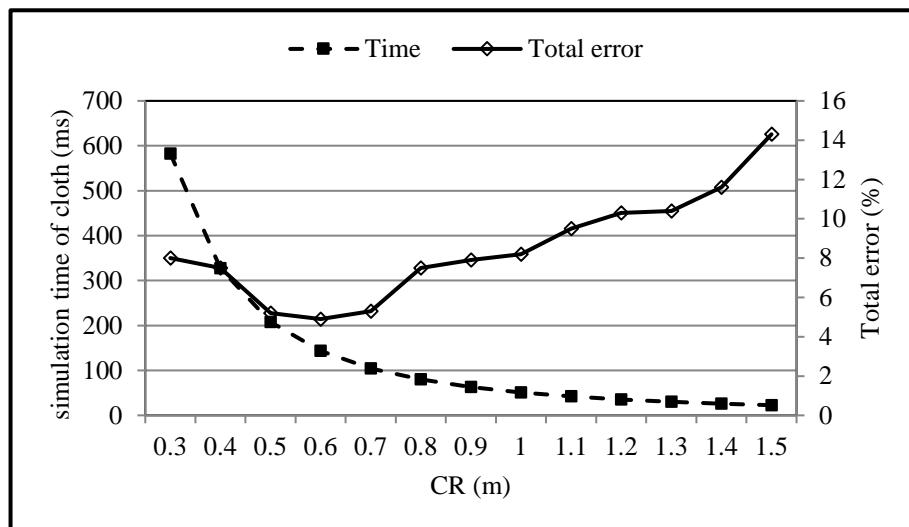


Figure 22. Cloth simulation time and total error at different CR values.

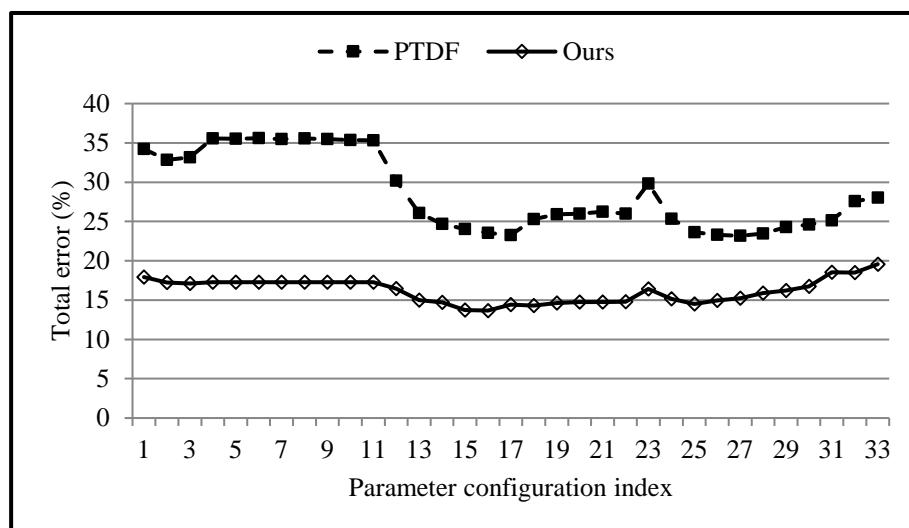


Figure 23. The graph of the relationship between the total error and parameter configuration index.

5.3. Four Filters with Higher Accuracy

Among the four filters with higher accuracy, the two filters developed by Mongus et al. [56] and Pingel et al. [24] used the optimal parameter values for each sample, but the two filters developed by Yang et al. [47] and Hu et al. [34] used the same parameter values or adaptive parameter values for each sample. Although the accuracy of the latter two filters is slightly lower than the accuracy of the first two filters, the latter two filters are more practical. Comprehensively considering practicality and accuracy, the performance of the proposed algorithm is only poorer than the two filters developed by Yang et al. [47] and Hu et al. [34]. Specifically, Yang et al. [47] fused on two filters based on different entities (i.e., points and segments) to extract ground points from LiDAR data. The algorithm is composed of two parts: (1) Smoothness-constrained segmentation is used to separate point clouds into point entities (e.g., trees) and segment entities (e.g., buildings) based on the area of each segment. (2) Filters based on point and segment entities are used in corresponding entities, respectively. Filter based

on point entities calculates the geometric properties (i.e., height difference) of each point and its neighboring points to judge each point property (ground or non-ground). Filter based on point entities usually fails to detect ground points around break lines. Filter based on segment entities calculates the height difference of each segment and its neighboring segments to judge each segment property (ground or non-ground). Filter based on segment entities is capable of retaining the break lines, but it has a disadvantage in removing non-ground points from the segments mixed ground and non-ground points. The area of the mixed segment is usually small, and they can be filtered better by filter based on point entities. The fusion of filters based on different entities makes use of their advantages. However, the performance of the fused filter significantly depends on segmentation results, and parameters in the segmentation process are difficult to set. Compared to the filter proposed by Yang et al. [47], the possible reason that the proposed algorithm yields lower accuracy is that we only used the filter based on point entities. Hu et al. [34] used a step factor to control the pyramidal hierarchical structure to obtain more ground points in a single iteration. In addition, the terrain surface was progressively refined using thin plate spline (TPS) instead of TIN, and the bending energy function that explicitly depicts the surface smoothness was used to calculate the adaptive threshold automatically. In contrast to TIN, TPS is constructed to generate smooth and oscillation-free trend surface, and it can produce local maxima in the neighborhood of the control points [3,31]. Above features reduce the influence of terrain gradient on the accuracy of the filtering. However, at the top level of the pyramid, cell resolution is difficult to determine. A low resolution may degrade rugged terrain features, whereas a high resolution may fail to remove large objects. Compared to the filter proposed by Hu et al. [34], the possible reason that the proposed algorithm yields lower accuracy is that TPS is more suitable for terrain expression than TIN. To summarize, compared to the proposed algorithm, the aforementioned filters have slightly higher accuracy but lower practicality, due to some parameters that are difficult to set.

6. Conclusions

Automatic filtering is usually necessary prior to LiDAR application for both DTM generation and feature extraction. In this study, we present a high accuracy and easy-to-use filtering algorithm. CS is used to obtain high-quality initial provisional DTM. Based on the initial provisional DTM, statistical analysis is exploited to adaptively adjust the PTD parameter thresholds under different terrain scenes. Finally, the algorithm utilizes PTD to refine the initial provisional DTM. The proposed algorithm makes the best use of the advantages of CSF and PTDF, that is, (1) CS can provide high-quality initial provisional DTM for PTD with only an easy-to-set integer parameter RT, and the high-quality initial provisional DTM can provide a good foundation for parameter threshold estimation of PTD, (2) PTD can yield promising results during refinement of the initial terrain, especially for hilltops and steep slopes. Benchmark datasets with low and high point density are used to compare the performance of the proposed algorithm with those of other publicized filtering algorithms. Overall, the proposed algorithm produces a promising filtering result and is highly automatic. The fact may help the users to use LiDAR data in their own applications. In the future, we will aim to develop a more sophisticated filter by combining optical images and LiDAR data.

Author Contributions: Conceptualization, S.C.; data curation, S.C., P.W. and J.S.; funding acquisition, W.Z. and G.Y.; methodology, S.C. and W.Z.; software, S.C. and J.Q.; project administration, G.Y.; supervision, W.Z.; writing—original draft, S.C.; writing—review and editing, W.Z., X.L., and S.Y.

Funding: This work was supported by the National Natural Science Foundation of China Grant Nos 41671414, 41331171 and 41171265. The National Key Research and Development Program of China (NO. 2016YFB0501404).

Acknowledgments: The authors would like to thank the editors and the anonymous reviewers for their valuable comments and suggestions, which have helped immensely in improving the quality of this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

- Meng, X.; Currit, N.; Zhao, K. Ground filtering algorithms for airborne LiDAR data: A review of critical issues. *Remote Sens.* **2010**, *2*, 833–860. [[CrossRef](#)]
- Sithole, G.; Vosselman, G. Experimental comparison of filter algorithms for bare-Earth extraction from airborne laser scanning point clouds. *ISPRS J. Photogramm. Remote Sens.* **2004**, *59*, 85–101. [[CrossRef](#)]
- Chen, Z.; Gao, B.; Devereux, B. State-of-the-art: DTM generation using airborne LIDAR data. *Sensors* **2017**, *17*, 150. [[CrossRef](#)] [[PubMed](#)]
- Liu, X. Airborne LiDAR for DEM generation: Some critical issues. *Prog. Phys. Geog.* **2008**, *32*, 31–49.
- Guo, Q.; Li, W.; Yu, H.; Alvarez, O. Effects of topographic variability and Lidar sampling density on several DEM interpolation methods. *Photogramm. Eng. Remote Sens.* **2010**, *76*, 701–712. [[CrossRef](#)]
- Hyppä, J.; Hyppä, H.; Leckie, D.; Gougeon, F.; Yu, X. Review of methods of small—Footprint airborne laser scanning for extracting forest inventory data in boreal forests. *Int. J. Remote Sens.* **2008**, *29*, 37–41. [[CrossRef](#)]
- Wang, Y.; Hyppä, J.; Liang, X.; Kaartinen, H.; Yu, X.; Lindberg, E.; Holmgren, J.; Qin, Y.; Mallet, C.; Ferraz, A.; et al. International benchmarking of the individual tree detection methods for modeling 3-D canopy structure for silviculture and forest ecology using airborne laser scanning. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 5011–5027. [[CrossRef](#)]
- Hyppä, J.; Yu, X.; Hyppä, H.; Vastaranta, M.; Holopainen, M. Advances in forest inventory using airborne laser scanning. *Remote Sens.* **2012**, *4*, 1190–1207. [[CrossRef](#)]
- Wulder, M.A.; White, J.C.; Nelson, R.F.; Næsset, E.; Ole, H.; Coops, N.C.; Hilker, T.; Bater, C.W.; Gobakken, T. Lidar sampling for large-area forest characterization: A review. *Remote Sens. Environ.* **2012**, *121*, 196–209. [[CrossRef](#)]
- Lim, K.; Treitz, P.; Wulder, M.; St-onge, B.; Flood, M. LiDAR remote sensing of forest structure. *Prog. Phys. Geog.* **2003**, *27*, 88–106. [[CrossRef](#)]
- Drake, J.B.; Dubayah, R.O.; Clark, D.B.; Knox, R.G.; Blair, J.B.; Hofton, M.A.; Chazdon, R.L.; Weishampel, J.F.; Prince, S. Estimation of tropical forest structural characteristics, using large-footprint lidar. *Remote Sens. Environ.* **2002**, *79*, 305–319. [[CrossRef](#)]
- Means, J.E.; Acker, S.A.; Fitt, B.J.; Renslow, M.; Emerson, L.; Hendrix, C.J. Predicting forest stand characteristics with airborne scanning lidar. *Photogramm. Eng. Remote Sens.* **2000**, *66*, 1367–1371.
- Rottensteiner, F.; Sohn, G.; Gerke, M.; Dirk, J.; Breitkopf, U.; Jung, J. Results of the ISPRS benchmark on urban object detection and 3D building reconstruction. *ISPRS J. Photogramm. Remote Sens.* **2014**, *93*, 256–271. [[CrossRef](#)]
- Wang, R. 3D building modeling using images and LiDAR: A review. *Int. J. Image Data Fusion.* **2013**, *4*, 273–292. [[CrossRef](#)]
- Wang, R.; Peethambaran, J.; Chen, D. LiDAR point clouds to 3-D urban models: A review. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2018**, *11*, 606–627. [[CrossRef](#)]
- Sampath, A.; Shan, J. Segmentation and reconstruction of polyhedral building roofs from aerial Lidar point clouds. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 1554–1567. [[CrossRef](#)]
- Vosselman, G. Slope based filtering of laser altimetry data. *Int. Arch. Photogramm. Remote Sens.* **2000**, *33*, 935–942.
- Sithole, G. Filtering of laser altimetry data using a slope adaptive filter. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2001**, *34*, 203–210.
- Shan, J.; Sampath, A. Urban DEM generation from raw lidar data. *Photogramm. Eng. Remote Sens.* **2005**, *71*, 217–226. [[CrossRef](#)]
- Meng, X.; Wang, L.; Silván-Cárdenas, J.L.; Currit, N. A multi-directional ground filtering algorithm for airborne LIDAR. *ISPRS J. Photogramm. Remote Sens.* **2009**, *64*, 117–124. [[CrossRef](#)]
- Susaki, J. Adaptive slope filtering of airborne lidar data in urban areas for Digital Terrain Model (DTM) generation. *Remote Sens.* **2012**, *4*, 1804–1819. [[CrossRef](#)]
- Keqi, Z.; Shu-Ching, C.; Whitman, D.; Mei-Ling, S.; Jianhua, Y.; Chengcui, Z.; Zhang, K. A progressive morphological filter for removing nonground measurements from airborne LIDAR data. *IEEE Trans. Geosci. Remote Sens.* **2003**, *41*, 872–882. [[CrossRef](#)]
- Chen, Q.; Gong, P.; Baldocchi, D.; Xie, G. Filtering airborne laser scanning data with morphological methods. *Photogramm. Eng. Remote Sens.* **2007**, *73*, 175–185. [[CrossRef](#)]

24. Pingel, T.J.; Clarke, K.C.; McBride, W.A. An improved simple morphological filter for the terrain classification of airborne LiDAR data. *ISPRS J. Photogramm. Remote Sens.* **2013**, *77*, 21–30. [[CrossRef](#)]
25. Li, Y.; Wu, H.; Xu, H.; An, R.; Xu, J.; He, Q. A gradient-constrained morphological filtering algorithm for airborne LiDAR. *Opt. Laser Technol.* **2013**, *54*, 288–296. [[CrossRef](#)]
26. Li, Y.; Yong, B.; Wu, H.; An, R.; Xu, H. An improved Top-Hat filter with sloped brim for extracting ground points from airborne Lidar point clouds. *Remote Sens.* **2014**, *6*, 12885–12908. [[CrossRef](#)]
27. Hui, Z.; Hu, Y.; Evenyo, Y.Z.; Yu, X. An improved morphological algorithm for filtering airborne LiDAR point cloud based on multi-level kriging interpolation. *Remote Sens.* **2016**, *8*, 35. [[CrossRef](#)]
28. Kraus, K.; Pfeifer, N. Determination of terrain models in wooded areas with aerial laser scanner data. *ISPRS J. Photogramm. Remote Sens.* **1998**, *53*, 193–203. [[CrossRef](#)]
29. Pfeifer, N.; Stadler, P.; Briese, C. Derivation of digital terrain models in the SCOP++ environment. In Proceedings of the OEEPE Workshop on Airborne Laser Scanning and Interferometric SAR for Digital Elevation Models, Stockholm, Sweden, 1–3 March 2001.
30. Elmquist, M. Ground surface estimation from airborne laser scanner data using active shape models. In Proceedings of the ISPRS Commission III Symposium, Photogrammetric and Computer Vision, Graz, Austria, 9–13 September 2002; pp. 114–118.
31. Mongus, D.; Žalík, B. Parameter-free ground filtering of LiDAR data for automatic DTM generation. *ISPRS J. Photogramm. Remote Sens.* **2012**, *67*, 1–12. [[CrossRef](#)]
32. Maguya, A.S.; Junttila, V.; Kauranne, T. Adaptive algorithm for large scale dtm interpolation from lidar data for forestry applications in steep forested terrain. *ISPRS J. Photogramm. Remote Sens.* **2013**, *85*, 74–83. [[CrossRef](#)]
33. Chen, C.; Li, Y.; Li, W.; Dai, H. A multiresolution hierarchical classification algorithm for filtering airborne LiDAR data. *ISPRS J. Photogramm. Remote Sens.* **2013**, *82*, 1–9. [[CrossRef](#)]
34. Hu, H.; Ding, Y.; Zhu, Q.; Wu, B.; Lin, H.; Du, Z.; Zhang, Y.; Zhang, Y. An adaptive surface filter for airborne laser scanning point clouds by means of regularization and bending energy. *ISPRS J. Photogramm. Remote Sens.* **2014**, *92*, 98–111. [[CrossRef](#)]
35. Qin, L.; Wu, W.; Tian, Y.; Xu, W. LiDAR filtering of urban areas with region growing based on moving-window weighted iterative least-squares fitting. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 841–845. [[CrossRef](#)]
36. Axelsson, P. DEM generation from laser scanner data using adaptive TIN models. *Int. Arch. Photogramm. Remote Sens.* **2000**, *33*, 111–118.
37. Zhang, J.; Lin, X. Filtering airborne LiDAR data by embedding smoothness-constrained segmentation in progressive TIN densification. *ISPRS J. Photogramm. Remote Sens.* **2013**, *81*, 44–59. [[CrossRef](#)]
38. Lin, X.; Zhang, J. Segmentation-based filtering of airborne LiDAR point clouds by progressive densification of terrain segments. *Remote Sens.* **2014**, *6*, 1294–1326. [[CrossRef](#)]
39. Chen, Q.; Wang, H.; Zhang, H.; Sun, M.; Liu, X. A point cloud filtering approach to generating DTMs for steep mountainous areas and adjacent residential areas. *Remote Sens.* **2016**, *8*, 71. [[CrossRef](#)]
40. Zhao, X.; Guo, Q.; Su, Y.; Xue, B. Improved progressive TIN densification filtering algorithm for airborne LiDAR data in forested areas. *ISPRS J. Photogramm. Remote Sens.* **2016**, *117*, 79–91. [[CrossRef](#)]
41. Liu, X.; Chen, Y.; Cheng, L.; Yao, M.; Deng, S.; Li, M.; Cai, D. Airborne laser scanning point clouds filtering method based on the construction of virtual ground seed points. *Appl. Remote Sens.* **2017**, *11*, 016032. [[CrossRef](#)]
42. Nie, S.; Wang, C.; Dong, P.; Xi, X.; Luo, S.; Qin, H. A revised progressive TIN densification for filtering airborne LiDAR data. *Measurement* **2017**, *104*, 70–77. [[CrossRef](#)]
43. Shi, X.; Ma, H.; Chen, Y.; Zhang, L.; Zhou, W. A parameter-free progressive TIN densification filtering algorithm for lidar point clouds. *Int. J. Remote Sens.* **2018**, *39*, 6969–6982. [[CrossRef](#)]
44. Zhang, W.; Qi, J.; Wan, P.; Wang, H.; Xie, D.; Wang, X. An Easy-to-Use airborne LiDAR data filtering method based on cloth simulation. *Remote Sens.* **2016**, *8*, 501. [[CrossRef](#)]
45. Podobnikar, T.; Vrečko, A. Digital elevation model from the best results of different filtering of a LiDAR point cloud. *Trans. GIS.* **2012**, *16*, 603–617. [[CrossRef](#)]
46. Montealegre, A.L.; Lamelas, M.T.; De La Riva, J. A Comparison of open-source LiDAR filtering algorithms in a mediterranean forest environment. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2015**, *8*, 4072–4085. [[CrossRef](#)]

47. Yang, B.; Huang, R.; Dong, Z.; Zang, Y.; Li, J. Two-step adaptive extraction method for ground points and breaklines from lidar point clouds. *ISPRS J. Photogramm. Remote Sens.* **2016**, *119*, 373–389. [[CrossRef](#)]
48. Serifoglu Yilmaz, C.; Yilmaz, V.; Güngör, O. Investigating the performances of commercial and non-commercial software for ground filtering of UAV-based point clouds. *Int. J. Remote Sens.* **2018**, *39*, 5016–5042. [[CrossRef](#)]
49. Wan, P.; Zhang, W.; Skidmore, A.K.; Qi, J.; Jin, X.; Yan, G.; Wang, T. A simple terrain relief index for tuning slope-related parameters of LiDAR ground filtering algorithms. *ISPRS J. Photogramm. Remote Sens.* **2018**, *143*, 181–190. [[CrossRef](#)]
50. Kobler, A.; Pfeifer, N.; Ogrinc, P.; Todorovski, L.; Oštir, K.; Džeroski, S. Repetitive interpolation: A robust algorithm for DTM generation from Aerial Laser Scanner Data in forested terrain. *Remote Sens. Environ.* **2007**, *108*, 9–23. [[CrossRef](#)]
51. Guan, H.; Li, J.; Yu, Y.; Zhong, L.; Ji, Z. DEM generation from lidar data in wooded mountain areas by cross-section-plane analysis. *Int. J. Remote Sens.* **2014**, *35*, 927–948. [[CrossRef](#)]
52. Hodgson, M.E.; Bresnahan, P. Accuracy of Airborne Lidar-Derived Elevation: Empirical Assessment and Error Budget. *Photogramm. Eng. Remote Sens.* **2004**, *70*, 331–339. [[CrossRef](#)]
53. Zhao, X.; Su, Y.; Li, W.; Hu, T.; Liu, J.; Guo, Q. A comparison of LiDAR filtering algorithms in vegetated mountain areas. *Can. J. Remote Sens.* **2018**, *44*. [[CrossRef](#)]
54. Lu, W.L.; Little, J.J.; Sheffer, A.; Fu, H. Deforestation: Extracting 3D bare-earth surface from airborne LiDAR data. In Proceedings of the 2008 Canadian Conference on Computer and Robot Vision, Windsor, ON, Canada, 28–30 May 2008; pp. 203–210.
55. Chang, L.; Slatton, K.; Krekeler, C. Bare-earth extraction from airborne LiDAR data based on segmentation modeling and iterative surface corrections. *Appl. Remote Sens.* **2010**, *4*, 1–30. [[CrossRef](#)]
56. Mongus, D.; Lukac, N.; Žalik, B. Ground and building extraction from LiDAR data based on differential morphological profiles and locally fitted surfaces. *ISPRS J. Photogramm. Remote Sens.* **2014**, *93*, 145–156. [[CrossRef](#)]
57. Hu, X.; Ye, L.; Pang, S.; Shan, J. Semi-global filtering of airborne LiDAR data for fast extraction of digital terrain models. *Remote Sens.* **2015**, *7*, 10996–11015. [[CrossRef](#)]
58. Li, Y.; Yong, B.; van Oosterom, P.; Lemmens, M.; Wu, H.; Ren, L.; Zheng, M.; Zhou, J. Airborne LiDAR data filtering based on geodesic transformations of mathematical morphology. *Remote Sens.* **2017**, *9*, 1104. [[CrossRef](#)]
59. Wang, L.; Xu, Y.; Li, Y. Aerial Lidar point cloud voxelization with its 3D ground filtering application. *Photogramm. Eng. Remote Sens.* **2017**, *83*, 95–107. [[CrossRef](#)]
60. Ni, H.; Lin, X.; Zhang, J.; Chen, D.; Peethambaran, J. Joint clusters and iterative graph cuts for ALS point cloud filtering. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2018**, *11*, 990–1004. [[CrossRef](#)]
61. Hui, Z.; Li, D.; Jin, S.; Ziggah, Y.; Wang, L.; Hu, Y. Automatic DTM extraction from airborne LiDAR based on expectationmaximization. *Opt. Laser Technol.* **2019**, *112*, 43–55. [[CrossRef](#)]
62. Khosravipour, A.; Skidmore, A.K.; Isenburg, M.; Wang, T.; Hussin, Y.A. Generating pit-free canopy height models from airborne lidar. *Photogramm. Eng. Remote Sens.* **2014**, *80*, 863–872. [[CrossRef](#)]
63. Zhang, K.; Whitman, D. Comparison of three algorithms for filtering airborne LiDAR data. *Photogramm. Eng. Remote Sens.* **2005**, *71*, 313–324. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).