

Airborne LiDAR Point Cloud Filtering Algorithm Based on Supervoxel Ground Saliency

Weiwei Fan, Xinyi Liu*, Yongjun Zhang*, Dongdong Yue, Senyuan Wang, Jiachen Zhong

School of Remote Sensing and Information Engineering, Wuhan University, 430079 Wuhan Hubei Province, China –
(fanweiwei, liuxy0319, zhangyj, yueyisui, collenwang, zjc0813)@whu.edu.cn

Commission II/WG 2

KEY WORDS: Airborne LiDAR, Ground Filtering, Supervoxel Clustering, Ground Saliency, Semi-global Optimization

ABSTRACT:

Airborne laser scanning (ALS) is able to penetrate sparse vegetation to obtain highly accurate height information on the ground surface. LiDAR point cloud filtering is an important prerequisite for downstream tasks such as digital terrain model (DTM) extraction and point cloud classification. Aiming at the problem that existing LiDAR point cloud filtering algorithms are prone to errors in complex terrain environments, an ALS point cloud filtering method based on supervoxel ground saliency (SGSF) is proposed in this paper. Firstly, a boundary-preserving TBBP supervoxel algorithm is utilized to perform supervoxel segmentation of ALS point clouds, and multi-directional scanning strip delineation and ground saliency computation are carried out for the clusters of supervoxel point clouds. Subsequently, the energy function is constructed by introducing the ground saliency and the optimal filtering plane of the supervoxel is solved using the semi-global optimization idea to realize the effective distinction between ground and non-ground points. Experimental results on the ALS point cloud filtering dataset openGF indicate that, compared to state-of-the-art surface-based filtering methods, the SGSF algorithm achieves the highest average values across various terrain conditions for multiple evaluation metrics. It also addresses the issue of recessed structures in buildings being prone to misclassification as ground points.

1. INTRODUCTION

With the rapid development of UAV and LiDAR technologies, airborne laser scanning (ALS) is widely used for 3D spatial information acquisition (Xue et al., 2020). The original ALS point cloud contains 3D information of various features in the scene, such as ground surface, vegetation, buildings, etc. The point cloud filtering can filter out the ground point cloud from the massive point cloud, which provides a foundation for the subsequent downstream tasks such as DTM extraction and point cloud classification. However, due to the highly similar geometrical structure of ground points and non-ground points in ALS point clouds, the problem of realizing accurate extraction of ground points from ALS data of large-scale and complicated scenes is still not completely solved.

Over the past several decades, numerous ground filtering algorithms for LiDAR point clouds have been proposed. Existing methods can be broadly categorized into slope-based, mathematical morphology-based, surface-based, and machine-learning-based methods (Sithole and Vosselman, 2004; Qin et al., 2023a). Methods based on slope assume that terrain slope changes are gradual, while non-ground points cause significant variations in slope. According to this assumption, ground points can be filtered out by applying a slope threshold between adjacent points (Vosselman, 2000). In order to improve the applicability of this method in scenarios such as steep slopes and terraces, adaptive slope-based filtering algorithms (Sithole and Vosselman, 2001; Susaki, 2012) have been proposed to accommodate the selection of filtering thresholds for different terrain features. Methods based on morphology typically use morphological opening and closing operations to construct filters for ground point filtering (Kilian et al., 1996). However, morphological methods with fixed-size windows are usually unable to filter large-size features and preserve fine ground surface at the same time, so progressive morphological filters were proposed (Zhang et al., 2003), where the filter window size corresponds to different elevation thresholds.

Surface-based methods determine a topographic reference surface by constructing a discriminant function and then using

the reference surface to separate ground points from non-ground points. Earlier studies (Kraus and Pfeifer, 2001) utilized a linear prediction-based filter to compute the mean elevation to construct the initial surface. In order to improve the accuracy of the reference surface, a representative algorithm based on multi-scale curvature classification, MCC, was proposed (Evans and Hudak, 2007), which utilizes Thin Plate Spline (TPS) interpolation to the seed points to generate the reference ground surface, and achieves good results in forested environments. Subsequent research has improved upon the iterative updating approach based on TPS surface generation (Hu et al., 2014; Liu et al., 2020). The triangular irregular net (TIN)-based method is another representative method for reference surface generation, which constructs a coarse TIN and iteratively encrypts it using the local minimum as a reference point (Axelsson, 2000). The method is widely used by commercial software, but the iterative computational process has a high computational cost.

Recently, several innovative methods for calculating terrain reference surfaces have been proposed. Zhang et al. (2016) introduced Cloth Simulation Filtering (CSF), which determines the reference surface by simulating the interaction between cloth nodes and the inverted point cloud. Hu et al. (2015) introduced the concept of ground saliency, which is calculated by segmenting the point cloud in a grid and following a prescribed sweep direction. They incorporated ground saliency into an energy function, employing semi-global optimization to determine the optimal reference surface for each grid. The approach of partitioning the point cloud into a regular grid demonstrates high computational efficiency but overlooks the boundary structure of the point cloud. It may face challenges in calculating the optimal reference surface in grids with relatively small height differences, thereby limiting its potential application in complex and mixed terrains.

Furthermore, point cloud filtering, as a form of classification task, can be addressed through machine learning methods. Earlier methods achieved classification of ground and non-ground points in point clouds by manually constructing features based on traditional classifiers (Lu et al., 2009; Jahromi et al., 2011), and the performance of such methods is limited by the descriptive

power of manually constructed features. Deep learning-based methods, depending on the type of input data into the network, can be classified into pipelines based on 2D images (Tatarchenko et al., 2018; Wen et al., 2020), 3D voxels (Graham et al., 2018; Yotsumata et al., 2020), and directly on 3D points (Qi et al., 2017; Zhang et al., 2021). Generally deep neural networks learn implicit features that better describe the contextual information of 3D point clouds, but learning-based methods usually require labeling samples to train models based on specific terrain conditions. In practice, the density variation of the point cloud and the quality of the labeled samples can affect the generalization ability of the filtering, which may produce poor filtering results in challenging localized areas. This paper proposes an ALS point cloud filtering method based on supervoxel ground saliency. In comparison to existing surface-based filtering methods, proposed SGSF algorithm comprehensively considers the geometric structural information of LiDAR point clouds. Through supervoxel segmentation, SGSF achieves initial differentiation of point clouds in the vertical direction, mitigating issues related to boundary intersections in filtering results. By incorporating semi-global optimization principles, the method aims to enhance the universality of filtering results while minimizing the input of filtering parameters.

2. METHODOLOGY

As illustrated in Figure 1, the SGSF filtering algorithm proposed in this paper primarily involves the following three steps: 1) Generation of point cloud supervoxel units based on boundary preservation. 2) Scanning strip delineation and ground saliency computation based on the supervoxel units. 3) Optimal filter plane solution based on the idea of semi-global optimization. Finally, we tested SGSF against representative surface-based filtering algorithms based on the latest airborne LiDAR point cloud filtering dataset.

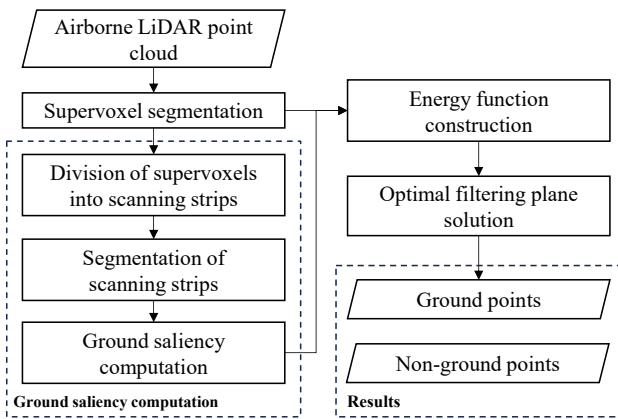
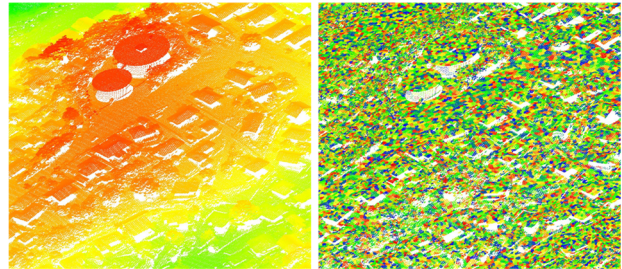


Figure 1. Flowchart of the SGSF algorithms.

2.1 Supervoxel unit segmentation

Methods that employ clustering for point cloud filtering frequently partition the point cloud into regular grids (Evans and Hudak, 2007; Hu et al., 2015). However, this approach may disrupt the original boundaries of objects in the point cloud, making it less conducive to further segmentation and clustering based on elevation. Therefore, this paper employs the supervoxel segmentation algorithm TBBP (Lin et al., 2018) to partition the point cloud into boundary-preserving supervoxel clusters. Unlike methods that require initializing seed points for super-voxels, this approach generates super-voxel units with adaptive size. This characteristic allows it to effectively preserve the boundaries of

fine structures such as trees and vehicles in ALS point clouds. Figure 2 shows an example of ALS point cloud supervoxel segmentation. The fundamental resolution r of supervoxels is provided as external input. After completing the supervoxel segmentation, the centroid coordinates of each super-voxel point cloud cluster are computed as the center point of the respective super-voxel. This information is utilized in subsequent supervoxel-based scan strip division.

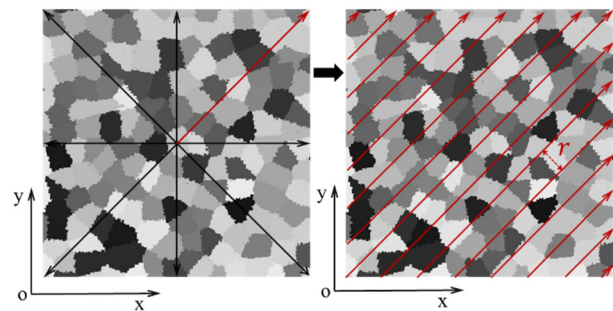


(a) ALS point cloud (b) supervoxel segmentation
 Figure 2. ALS point cloud supervoxel segmentation

2.2 Ground saliency calculation based on supervoxels

2.2.1 Scanning strip division

Ground saliency is determined by cumulatively computing the number of height difference step changes in multiple directions. It represents the initial probability of the supervoxel unit belonging to the ground category. Firstly, the supervoxels are divided into scan strips along N directions based on their centroid coordinates, as illustrated in Figure 3. Specifically, in the 2D XY plane, starting from the y-axis and rotating clockwise around the origin, each scan direction is separated by $360/N$ degrees. This process generates N evenly distributed scan directions in the 2D plane. The scanning direction here refers to the direction in which the supervoxel unit performs elevation difference calculations, and is independent of the scanning direction during LiDAR point cloud acquisition. The value of N is a constant, and a larger N enhances the smoothness of the filtering but also increases computational complexity. In this paper, N is set to 8, as utilizing 8 directions strikes a balance between filtering accuracy and computational efficiency.



(a) Division into 8 scan directions. (b) Scanning strip divisions in a single direction.

Figure 3. Schematic diagram of scanning strip divisions

2.2.2 Calculation of ground saliency

After dividing the scanning directions and obtaining the scanning strips in each direction, the scanning strips in each direction were sorted, segmented and the ground significance values were calculated. The specific steps include the following:

(1) Within each scanning strip, supervoxel units are sorted in ascending order based on the x-axis of their centroids. Specifically, for strips parallel to the y-axis scanning direction, supervoxel units are sorted in ascending order based on the y-axis of their centroids. The elevation value of each supervoxel is set

to the elevation of the lowest point within that supervoxel, and the initial saliency value λ_i of supervoxel S_i is uniformly set to 1. (2) Compute the absolute value of the elevation difference between two adjacent supervoxels in the sorting direction. If this absolute value is less than the segmented elevation difference threshold ΔH , the supervoxel is assigned to the current segment; otherwise, it is assigned to the next segment. Segmentation of all supervoxels in scanning strips along the scanning direction.

(3) After the segmentation is complete, calculate the elevation difference between the last supervoxel in the current segment and the first supervoxel in the adjacent next segment along the scanning direction. If the elevation difference is greater than twice the segmented elevation difference threshold ΔH , subtract $1/N$ from the saliency value λ_i of all supervoxel units within that segment.

(4) Perform the operations of steps (2) and (3) separately for each of the N scanning directions to obtain the saliency value λ_i for each supervoxel unit S_i .

The elevation difference threshold ΔH described above is externally inputted, and this parameter determines the lowest feature that can be distinguished during the segmentation process, which is typically set to around 1.0m. After the significance calculation, supervoxels with elevations higher than the adjacent units have low saliency (close to 0), while supervoxels with elevations lower than the adjacent units have high saliency (close to 1). However, the elevation distribution of the actual terrain is more complex. For instance, the central depressed part of a building is prone to be incorrectly segmented as the ground. Therefore, for point clouds with structures featuring central depressions, such as buildings, an iterative calculation of saliency in step (3) can be performed for segments where saliency remains unchanged during a single scanning process. The iterative calculation process is illustrated in Figure 4, and typically, iterating three times to calculate ground saliency during scanning effectively filters out abnormal ground saliency values caused by central depressions in buildings. For point clouds without central depression structures in buildings, a single scanning process is sufficient for accurate saliency calculation.

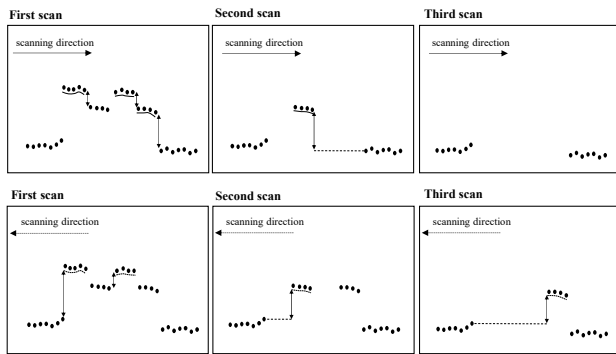


Figure 4. Iterative calculation of ground saliency. The figure shows the process of calculating the ground saliency for three iterative scans in each of the two opposite scanning directions. The underlined segments in the figure represent that the height difference of the segment is greater than the threshold ΔH , then the saliency of all the supervoxel units in the current segment is subtracted by $1/N$, which is hidden in the next iteration and does not need to participate in the calculation.

2.3 Optimal filter plane solution

In order to achieve accurate classification of ground and non-ground points, we utilize the ground saliency calculated at the supervoxel unit as a coordinating factor. For each supervoxel, we construct an energy function for the optimal filtering plane and

solve for the optimal segmentation plane to accomplish point cloud filtering. Firstly, the minimum elevation value H_{S_i} within the point cloud of supervoxel unit S_i is considered as the elevation value for the current supervoxel. Here, H_{min} represents the overall minimum elevation value of the input point cloud. The point cloud of the supervoxel is then discretized in the elevation direction.

$$N_z = \frac{H_{S_i} - H_{min}}{d} + m \quad (1)$$

$$L_{S_i} = \{l_i^n \mid l_i^n = H_{min} + n * d\}_{n=0}^{N_z} \quad (2)$$

where d is the unit distance of elevation discretization, which is set as 0.2 times of the segmentation threshold ΔH . The elevation compensation coefficient m is typically set to 5. N_z is the maximum value that the filtering plane of supervoxel S_i can take within the discretized range; The variable n is any integer within the range of $[0, N_z]$. l_i^n denotes the elevation value of the candidate filtering plane for supervoxel S_i , and L_{S_i} is the set of l_i^n . The energy function of the supervoxel S_i on the t -th scanning strip is expressed as:

$$E_t(S_i) = E_{data}(S_i) + E_{smooth}(S_{i-1}, S_i) \quad (3)$$

where the data term $E_{data}(S_i)$ consists of the difference H_d between the elevation value l_i^n of the candidate filter plane and the elevation value H_{S_i} of the current supervoxel, and the penalty term for l_i^n crossing the supervoxel. Simultaneously, the ground saliency value λ_i calculated based on supervoxels in Section 2.2 is introduced as a coordinating factor. This ensures that the energy function is dominated by the data term when the ground saliency is high, and by the smoothing term when the ground saliency is low. The expression for E_{data} is given by:

$$E_{data}(S_i) = \begin{cases} \lambda_i * (1.0 - e^{-H_d^2}) + (1 - \lambda_i) * H_d, & H_d \geq 0 \\ \lambda_i * (1.0 - e^{-H_d^2}), & H_d < 0 \end{cases} \quad (4)$$

where $H_d = l_i^n - H_{S_i}$. The smoothing term E_{smooth} consists of the absolute value of the difference between the elevation of the optimal filter plane of the current supervoxel S_i and that of the adjacent previous supervoxel S_{i-1} in the scanning direction, and is used to smooth out the elevation difference of the optimal filter planes between the neighboring segments to make the overall filter plane smoother. E_{smooth} is expressed as:

$$E_{smooth}(S_{i-1}, S_i) = |l_{i-1}^n - l_i^n| \quad (5)$$

The optimal filter plane is determined by accumulating and minimizing the energy function for each supervoxel unit in N directions. Following the concept of semi-global optimization (Hirschmuller, 2005), the result of this filter plane solution approximates the global optimal filter plane:

$$l_i = \underset{l_i}{\operatorname{argmin}} \left(\sum_{t=1}^N E_t(S_i) \right) \quad (6)$$

Based on the optimal filtering plane l_i , each point in the supervoxel point cloud clusters S_i is individually assessed using the following calculation method: If the elevation value of a point is less than l_i or if the absolute difference between the elevation value of the point and l_i is less than the discretization unit length d , then the point is classified as a ground point. Otherwise, the point is classified as a non-ground point.

3. EXPERIMENTAL RESULTS

To validate the effectiveness of the proposed method, a comprehensive comparison was conducted with three advanced surface-based airborne LiDAR point cloud filtering methods: CSF, MCC, and SGF. In the experimental process, the supervoxel resolution r was set to 2m, the elevation difference threshold ΔH was set to 1m, and the number of scanning directions N was set to 8. Among the compared methods, the CSF algorithm was implemented using CloudCompare software,

the MCC algorithm was implemented using MCC-LiDAR software, and both SGF and the proposed SGSF algorithm were implemented on the Visual Studio 2019 platform. The experimental computer was configured with an AMD Ryzen 9 5900HX CPU, 32GB-RAM, and Windows 11 x64 operating system. The hyperparameters of the compared algorithms were also appropriately adjusted to their optimal settings.

3.1 Dataset

Area	Topographic features	Main objects
S1	smooth	complicated buildings, vegetation, viaduct, vehicles
S2	smooth	dense buildings, vegetation, waterbody, vehicles, boats
S3	smooth	dense buildings, vegetation
S4	smooth	dense buildings, vegetation
S5	rugged	dense buildings, vegetation, slope
S6	rugged	buildings, vegetation, slope
S7	rugged	dense vegetation, roads
S8	sharp	sparse vegetation, steep
S9	sharp	dense vegetation, steep

Table 1. Main topographic features of the test area

For an objective and comprehensive comparison with other methods, we selected the latest known airborne LiDAR point cloud filtering dataset, OpenGF (Qin et al., 2023b). The OpenGF dataset comprises 9 representative scenes, including metropolis with large roofs, metropolis with dense roofs, small city with flat ground, small city with local undulating ground, small city with rugged ground, village with scattered roofs, mountain with gentle slopes and dense vegetation, mountain with steep slopes and sparse vegetation, and mountain with steep slopes and dense vegetation. For each scene, multiple point cloud blocks with semantic ground and non-ground labels are provided for the development of deep learning-based filtering pipelines. As our method is not learning-based, we conducted comparative experiments by selecting point cloud of size 500×500 m² from each of the 9 representative scenes in the OpenGF dataset, for comparative experiments to verify the filtering performance across various types of scenes. Table 1 provides a detailed list of

the topographic features and main land objects types in the test area. Before conducting the experiments, we filtered out unnecessary noise in the point cloud based on label values.

3.2 Quantitative results

In order to objectively evaluate the accuracy of our method, we refer to the evaluation metrics of Qin et al. (2023b), including the intersection-over-union (IoU) of ground and non-ground points and the overall accuracy (OA). Where IoU_{ng} and IoU_g represent the classification accuracies of non-ground and ground points, respectively, and OA represents the overall accuracy of the filtering algorithm. The specific expressions are as follows:

$$OA = \frac{TP + TN}{TP + FN + FP + TN} \quad (7)$$

$$IoU_{ng} = \frac{TP_{ng}}{TP_{ng} + FP_{ng} + FP_g} \quad (8)$$

$$IoU_g = \frac{TP_g}{TP_g + FP_g + FP_{ng}} \quad (9)$$

where, TP_{ng} represents the number of correctly recognized non-ground points, FP_{ng} represents the number of non-ground points incorrectly recognized as ground points, TP_g represents the number of correctly recognized ground points, and FP_g represents the number of ground points incorrectly recognized as non-ground points. Additionally, we calculated the Kappa coefficient (Cohen, 1960) as a measure of consistency for the filtering classifier. A higher Kappa coefficient indicates that the classification results are unbiased.

The experimental results are shown in Table 1. From Table 1, it can be observed that SGSF filtering algorithm achieved the highest average OA and IoU_g compared to CSF, MCC, and SGF, and also demonstrated sub-optimal results in terms of the average Kappa coefficient and IoU_{ng} . Overall, SGSF achieved competitive results in 7 out of the 9 tested scenarios, demonstrating the robustness of our proposed algorithm across various environments.

Area	CSF				MCC				SGF				SGSF			
	OA (%)	Kp (%)	IoU_{ug} (%)	IoU_g (%)	OA (%)	Kp (%)	IoU_{ug} (%)	IoU_g (%)	OA (%)	Kp (%)	IoU_{ug} (%)	IoU_g (%)	OA (%)	Kp (%)	IoU_{ug} (%)	IoU_g (%)
S1	97.37	94.71	95.19	94.49	88.10	76.40	87.77	78.21	94.43	88.78	90.22	88.54	98.19	96.37	96.63	96.23
S2	96.29	91.75	94.53	89.67	94.31	87.28	91.77	84.45	95.26	89.08	93.29	86.13	96.41	92.03	94.68	90.02
S3	95.96	91.79	91.30	92.91	92.96	85.85	85.98	87.60	94.35	88.66	88.58	89.95	95.93	91.81	91.33	92.88
S4	96.65	93.29	93.31	93.71	95.94	91.88	92.05	92.33	96.89	93.78	93.77	94.06	96.98	93.96	93.87	94.39
S5	87.79	73.77	82.36	71.59	89.10	76.68	84.03	74.45	88.62	75.08	83.85	72.18	88.02	75.61	81.39	74.85
S6	86.95	74.10	86.81	76.69	77.14	71.47	75.09	74.41	91.90	83.77	84.20	85.74	96.74	93.39	92.85	94.36
S7	84.22	57.16	81.33	49.52	86.74	61.21	84.45	52.70	87.53	60.26	85.64	51.23	88.18	61.42	86.43	52.12
S8	77.89	54.54	68.57	57.30	83.11	66.12	72.61	69.41	71.76	40.52	64.30	42.54	80.45	61.86	65.17	69.17
S9	88.93	62.81	87.39	52.41	92.53	64.39	91.87	52.23	91.08	50.95	90.56	38.55	90.06	57.66	89.13	46.36
mean	90.23	77.10	86.75	75.37	88.88	75.70	85.07	73.98	90.20	74.54	86.05	72.10	92.33	80.46	87.94	78.93

Table 2. Quantitative results of four algorithms. The table highlights the optimal and suboptimal results for each metric.

3.3 Qualitative results

SGSF demonstrated significant advantages in the filtering results for scenes S1-S3 of the OpenGF dataset, with Figure 5 providing

qualitative filtering results for part of the S1 scene. In the figure, we labeled Type I errors (ground points mistakenly classified as non-ground points) and Type II errors (non-ground points mistakenly classified as ground points). The features in these

scenes often have relatively regular structures, and supervoxel segmentation can effectively address the issue of boundary crossings between ground and non-ground points. This is beneficial for subsequent calculations of the optimal filtering plane on a supervoxel point cloud cluster basis. For the CSF algorithm, the flat ground in the urban scene enables it to obtain more accurate filtering results under harder cloth conditions, but such parameter settings also cause the CSF to lose some details of the ground on gentle slopes, while setting softer cloth parameters will prompt the cloth particles to stick to the middle part of large buildings, increasing the Type II error. The SGF and MCC algorithms do not account for the top depressions in building structures, making them prone to misclassifying the depressed parts of large buildings as ground. Additionally, the point cloud partitioning method using a regular grid unit is not conducive to accurately calculating filtering planes at object boundaries (such as curbstones with elevation changes), which is prone to Type I errors.

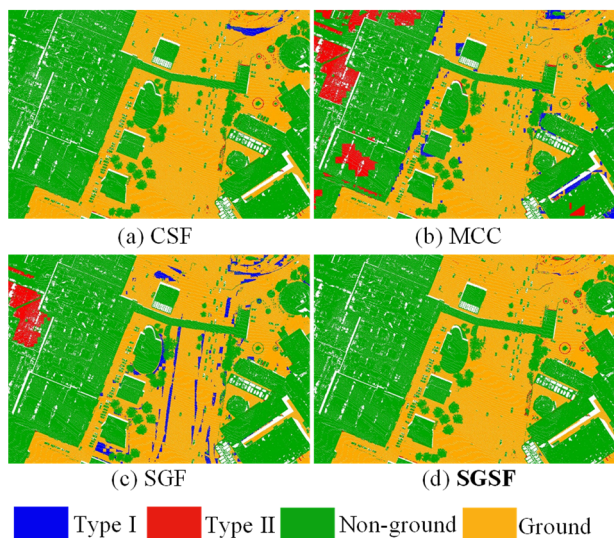


Figure 5. Qualitative results of the four algorithms in the S1 area.

In addition, for the small urban area and forest scenes with undulating ground in the OpenGF dataset (S4-S7), SGSF demonstrated advantages compared to the other three contrast algorithms. For steep mountainous forest terrain (Area S8, S9), SGSF shows a significant advantage over SGF and CSF algorithms, but its overall performance is not as effective as the MCC filtering algorithm designed specifically for forest environments. The reason may be that the supervoxel segmentation of SGSF is not sensitive to the sparse ground in forest areas with large height differences, making it difficult to detect sparse ground points with significant topographic height differences. Overall, SGSF outperforms SGF, CSF, and MCC in terms of the average values of various evaluation metrics under multiple terrain conditions, demonstrating the robust application potential of our method in various terrain scenarios.

3.4 Comparisons with Baseline Methods

We specifically compared our baseline method, SGF (Hu et al., 2015), which calculates ground saliency based on regular grid partitioning. Figure 6 visualizes the ground saliency calculated based on supervoxel segmentation (SGSF) and the ground saliency calculated based on regular grid partitioning (SGF) for the S6 area. The visualization also presents the final filtering results of both algorithms. It can be observed that due to the penetrability of LiDAR point clouds, there are sparse ground

points beneath the trees. As shown in Figure 6(a), the ground saliency computation based on a regular grid only considers the lowest point in the two-dimensional plane of the grid, neglecting the terrain boundaries in the vertical direction. Consequently, it is challenging to distinguish between ground and non-ground points in regions with coherent topography. However, the ground saliency computed based on supervoxels considers effective segmentation along the vertical direction of terrain boundaries, significantly alleviating the impact of cross-boundary effects. As depicted in the results shown in Figure 6(b), the ground saliency of ground points in the scene approaches 1, while the ground saliency of non-ground points approaches 0, achieving an effective preliminary distinction between ground and non-ground points. This further enhances the accuracy of the subsequent calculation of the optimal filtering plane.

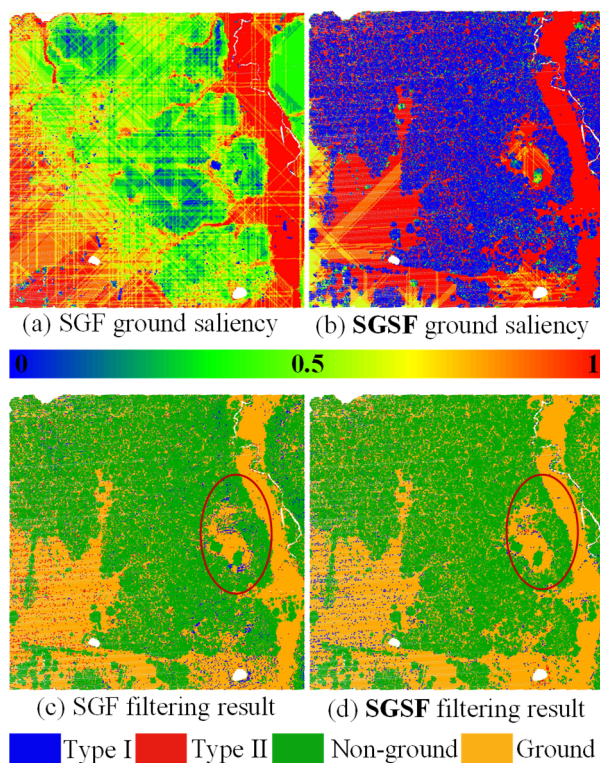
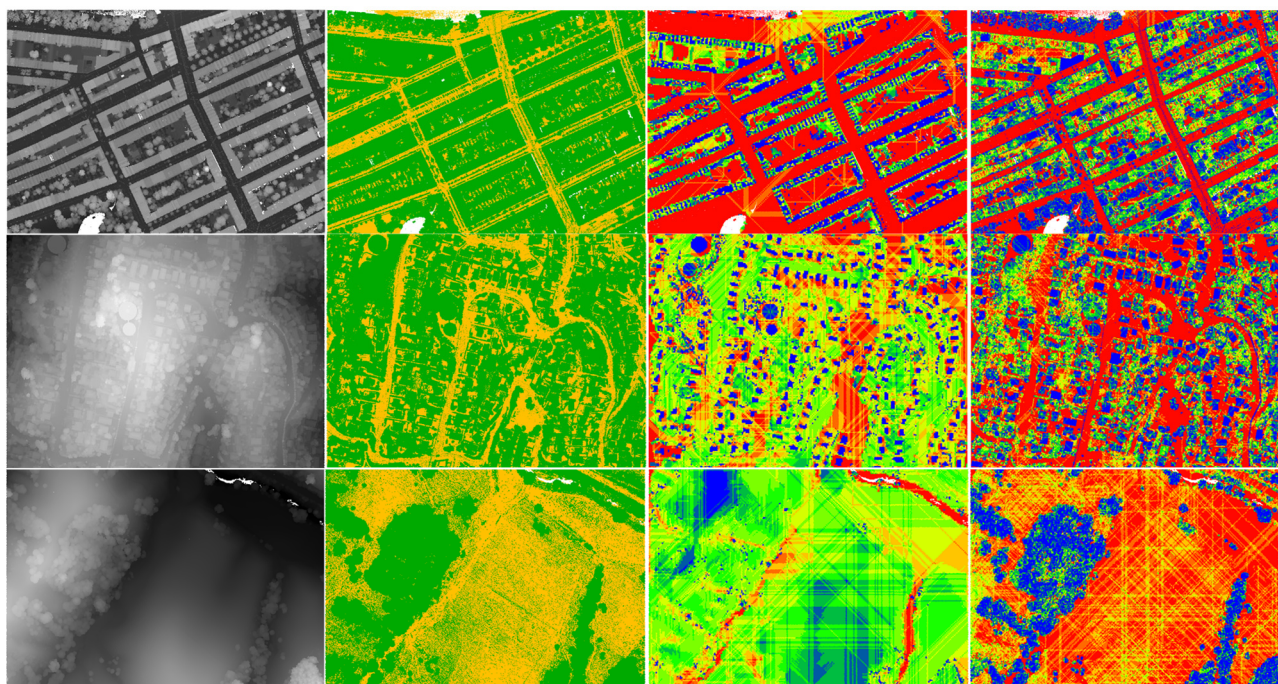


Figure 6. Comparison results of SGSF and SGF algorithms in the S6 area.

We also visualized the comparison results of ground saliency calculation under different terrain conditions in Figure 7. From the results, it can be observed that the ground saliency calculated based on supervoxel segmentation (Figure 7d) effectively separates small structures such as trees, vehicles, and power lines. These structures are often overlooked by the ground saliency calculated by SGF (Figure 7c). This enables SGSF to obtain more distinguishable ground saliency values in various terrain environments, enhancing the robustness of the SGSF algorithm. We conducted a statistical analysis of the distribution histograms of ground saliency values in S6 area calculated by the two methods (Figure 8). From the graph, it can be observed that the ground saliency values calculated by SGSF are mainly concentrated around 0 or 1, allowing for a good preliminary separation of ground and non-ground points. However, the ground saliency values calculated by SGF are uniformly distributed around 0.5, failing to effectively differentiate between ground and non-ground points. This can also have a negative impact on the subsequent calculation of the optimal filtering plane based on ground saliency.



(a) Point cloud depth map (b) ground truth (c) SGF ground saliency (d) SGSF ground saliency
Figure 7. Comparison of the results of ground saliency calculations in different terrains

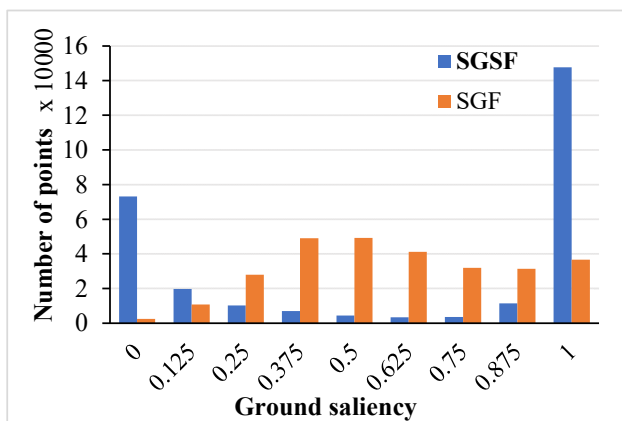


Figure 8. Histogram of the number distribution of ground saliency for the SGSF and SGF algorithms.

3.5 Algorithm efficiency comparison

In terms of algorithm efficiency, the number of point clouds in a single block of the openGF validation set used in the experiments ranges from 200,000 to 6 million points, and the computation time of SGSF in the S2 plot with the largest amount of data is about 24s, which is lower than the efficiency of SGF (about 5s) and CSF (about 4s) under the same parameters, but better than the computation efficiency of MCC, whose computation time is usually more than 4min. The increased computational effort of SGSF compared to SGF is mainly reflected in the segmentation of supervoxels. Since the TBBP algorithm can adaptively preserve the fine structure of features when performing supervoxel segmentation, the number of segmented supervoxels is generally larger than the number of regular grid divisions at the same resolution, which inevitably increases the computational amount.

4. CONCLUSIONS

We propose an airborne LiDAR point cloud filtering algorithm, SGSF, based on supervoxels to calculate ground saliency. This

method takes full consideration of the geometric structure of LiDAR point clouds, computes ground saliency using supervoxels as units, and subsequently solves for the optimal filtering plane, achieving precise classification between ground and non-ground points. The experimental results on the ALS filtering dataset OpenGF demonstrate that, across various terrain environments in ALS point clouds, the averages of various evaluation metrics for SGSF are superior to those of surface-based filtering algorithms such as CSF, MCC, and SGF, indicating its promising application potential across diverse scenarios.

Compared with the filtering algorithm based on regular grid division, SGSF is able to correctly recognize the depressed structures of large buildings in urban areas and the fine structural boundaries of terrain features, which is a significant advantage in the scenarios of urban areas with gentle terrain and small cities with rugged ground. However, the filtering performance of SGSF in steep mountainous terrain needs further improvement. Our future work will primarily focus on adaptively distinguishing and enhancing the filtering effectiveness in mountainous terrain. Additionally, the computational speed of SGSF is currently slow due to supervoxel segmentation. We plan to address this by introducing parallel computing techniques to improve the algorithm efficiency.

ACKNOWLEDGEMENTS

This work was partially supported by the National Natural Science Foundation of China (Grant No. 42201474) and the Science and Technology Major Project of Hubei Province (Grant No. 2021AAA010).

REFERENCES

Axelsson, P., 2000. DEM generation from laser scanner data using adaptive TIN models. *International archives of photogrammetry and remote sensing*, 33(4), 110–117.

- Cohen, J., 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1), 37–46.
- Evans, J.S., Hudak, A.T., 2007. A Multiscale Curvature Algorithm for Classifying Discrete Return LiDAR in Forested Environments. *IEEE Transactions on Geoscience and Remote Sensing*, 45(4), 1029–1038.
- Graham, B., Engelcke, M., Van Der Maaten, L., 2018. 3d semantic segmentation with submanifold sparse convolutional networks, in: IEEE Conference on Computer Vision and Pattern Recognition. pp. 9224–9232.
- Hirschmuller, H., 2005. Accurate and efficient stereo processing by semi-global matching and mutual information, in: IEEE Conference on Computer Vision and Pattern Recognition. pp. 807–814.
- Hu, H., Ding, Y., Zhu, Q., Wu, B., Lin, H., Du, Z., Zhang, Yeting, Zhang, Yunsheng, 2014. An adaptive surface filter for airborne laser scanning point clouds by means of regularization and bending energy. *ISPRS Journal of Photogrammetry and Remote Sensing*, 92, 98–111.
- Hu, X., Ye, L., Pang, S., Shan, J., 2015. Semi-Global Filtering of Airborne LiDAR Data for Fast Extraction of Digital Terrain Models. *Remote Sensing*, 7(8), 10996–11015.
- Jahromi, A.B., Zojj, M.J.V., Mohammadzadeh, A., Sadeghian, S., 2011. 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(4), 836–843.
- Kilian, J., Haala, N., Englich, M., 1996. Capture and evaluation of airborne laser scanner data. *International Archives of Photogrammetry and Remote Sensing*, 31, 383–388.
- Kraus, K., Pfeifer, N., 2001. Advanced DTM generation from LIDAR data. *International Archives Of Photogrammetry Remote Sensing And Spatial Information Sciences*, 34(3/W4), 23–30.
- Lin, Y., Wang, C., Zhai, D., Li, W., Li, J., 2018. Toward better boundary preserved supervoxel segmentation for 3D point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 143, 39–47.
- Liu, X., Zhang, Y., Huang, X., Wan, Y., Wang, S., 2020. Terrain-Adaptive Ground Filtering of Airborne LIDAR Data Based on Saliency-Aware Thin Plate Spline. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 43, 279–285.
- Lu, W.-L., Murphy, K.P., Little, J.J., Sheffer, A., Fu, H., 2009. A hybrid conditional random field for estimating the underlying ground surface from airborne LiDAR data. *IEEE Transactions on Geoscience and Remote Sensing*, 47(8), 2913–2922.
- Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space, in: Neural Information Processing Systems. pp. 5105–5114.
- Qin, N., Tan, W., Guan, H., Wang, L., Ma, L., Tao, P., Fatholahi, S., Hu, X., Li, J., 2023a. Towards intelligent ground filtering of large-scale topographic point clouds: A comprehensive survey. *International Journal of Applied Earth Observation and Geoinformation*, 125, 103566.
- Qin, N., Tan, W., Ma, L., Zhang, D., Guan, H., Li, J., 2023b. Deep learning for filtering the ground from ALS point clouds: A dataset, evaluations and issues. *ISPRS Journal of Photogrammetry and Remote Sensing*, 202, 246–261.
- Sithole, G., Vosselman, G., 2004. Experimental comparison of filter algorithms for bare-Earth extraction from airborne laser scanning point clouds. *ISPRS journal of photogrammetry and remote sensing*, 59(1–2), 85–101.
- Sithole, G., Vosselman, G., 2001. Filtering of laser altimetry data using a slope adaptive filter. *International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences*, 34(3/W4), 203–210.
- Susaki, J., 2012. Adaptive slope filtering of airborne LiDAR data in urban areas for digital terrain model (DTM) generation. *Remote Sensing*, 4(6), 1804–1819.
- Tatarchenko, M., Park, J., Koltun, V., Zhou, Q.-Y., 2018. Tangent convolutions for dense prediction in 3d, in: IEEE Conference on Computer Vision and Pattern Recognition. pp. 3887–3896.
- Vosselman, G., 2000. Slope based filtering of laser altimetry data. *International archives of photogrammetry and remote sensing*, 33(B3/2; PART 3), 935–942.
- Wen, C., Yang, L., Li, X., Peng, L., Chi, T., 2020. Directionally constrained fully convolutional neural network for airborne LiDAR point cloud classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 162, 50–62.
- Xue, F., Lu, W., Chen, Z., Webster, C.J., 2020. From LiDAR point cloud towards digital twin city: Clustering city objects based on Gestalt principles. *ISPRS Journal of Photogrammetry and Remote Sensing*, 167, 418–431.
- Yotsumata, T., Sakamoto, M., Satoh, T., 2020. Quality improvement for airborne lidar data filtering based on deep learning method. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 43, 355–360.
- Zhang, K., Chen, S.-C., Whitman, D., Shyu, M.-L., Yan, J., Zhang, C., 2003. A progressive morphological filter for removing nonground measurements from airborne LIDAR data. *IEEE transactions on geoscience and remote sensing*, 41(4), 872–882.
- Zhang, W., Qi, J., Wan, P., Wang, H., Xie, D., Wang, X., Yan, G., 2016. An Easy-to-Use Airborne LiDAR Data Filtering Method Based on Cloth Simulation. *Remote Sensing*, 8(6), 501.
- Zhang, Z., Sun, L., Zhong, R., Chen, D., Zhang, L., Li, X., Wang, Q., Chen, S., 2021. Hierarchical Aggregated Deep Features for ALS Point Cloud Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(2), 1686–1699.