A LOCAL TOPOLOGICAL INFORMATION AWARE BASED DEEP LEARNING METHOD FOR GROUND FILTERING FROM AIRBORNE LIDAR DATA

Zhipeng Luo[1], Ziyue Zhang[2], Wen Li[1], Haojia Lin[1], Yiping Chen[1], Cheng Wang[1], Jonathan Li[3]*

[1]Fujian Key Laboratory of Sensing and Computing for Smart Cities, School of Informatics
Xiamen University, Xiamen, FJ 361005, China
[2]School of Computer Science, University of Nottingham Ningbo China, Ningbo, ZJ 315100, China
[3]Department of Geography and Environmental Management and Department of Systems Design Engineering,
University of Waterloo, Waterloo, Ontario N2L 3G1, Canada
*Corresponding author: junli@uwaterloo.ca (J. Li)

IGARSS 2021 - 2021 IEEE International Geoscience and Remote Sensing Symposium | 978-1-6654-0369-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/IGARSS47720.2021.9554781

ABSTRACT

As a foundational preprocessing step for a lot of downstream tasks, ground filtering from airborne LiDAR data is designed to separate the ground points and preserve the off-ground points with complete shape information. However, because of the undulating terrain, it is still a challenge work to filter the ground under complex mountain regions. In this paper, we provide a deep learning based model to improve the ground filtering performance in abrupt slope using airborne LiDAR point clouds. Specifically, we first design a local topological information mining module to extract the local features. Then a modified graph convolutional networks (GCNs) is developed to fusion the local features and global features. Compared with most existing methods, our model not only enjoys the parameter-free advantage, which means it can be applied easily in various areas, but also obtains better ground filtering performance and can preserve more complete information contained in off-ground points. Experiments was implemented on seven forest areas. The proposed method obtains promising ground filtering results with mean total error of 6.46% and the mean kappa coefficient of 86.01%.

***Index Terms*—** Ground filtering, airborne LiDAR, point cloud, graph convolutional network

1. INTRODUCTION

Airborne Light Detection and Ranging (LiDAR) or airborne laser scanning (ALS) has been considered as one of the standard ways to acquire data in the large areas, especially the mountain regions. It has been used in several tasks, such as forest inventory [1], urban three-dimensional (3D) analysis [2], and object detection [3]. As a foundational preprocessing step for these tasks, ground filtering from airborne LiDAR data is designed to separate the ground points and preserve the off-ground points with complete shape information. It is critical in ALS point clouds analysis, especially under the abrupt slope environments. A suitable filtering algorithm can obtain clean point clouds of off-surface objects, thus providing good initial data for subsequent applications, such as tree extraction and parameter estimation.

Several researches have been undertaken to study the ground filtering problem, such as the morphological based methods [4], surface based methods [5], slope analysis based methods [6] and the statistical approach based method [7]. Although these methods can obtain promising results under relatively flat environments, the challenge remains in processing abrupt slope areas. Considering that the rich detail information is mainly contained the relationship between local points, the key problems of the above challenge mainly lies in two aspects: the effective mining of the local features and the descriptiveness of the global features. Therefore, in this paper, we first design a local topological information (LTI) module to effectively extract the local features. Then we attempt to apply the deep learning based approach to improve the descriptiveness of features generated by ALS point clouds in complex forests. Specially, a modified graph convolutional networks (GCNs) is developed to fusion the local features and global features.

The rest of this paper is structured as follows: Section 2 provides related works and Section 3 details the proposed method. Experimental results are presented in Section 4. Section 5 concludes our work.

2. RELATED WORK

Ground filtering is a critical preprocessing task in ALS point clouds analysis, especially under the abrupt slope environments. Existing methods can be categorized into two classes: point entities based methods and segment entities based methods [8]. In point based methods, ground and off-ground points are separated by their own geometric properties. Representative works include morphological based researches [4], surface based methods [5], slope based works [6]. Although these methods achieve good quality results under relatively flat environments, the challenge remains in processing non-flat areas. Most of the existing methods require different parameters to adapt for different kinds of environments (e.g. urban, mountains) and of different datasets, which results heavy manual editing costs and time.

Segment based methods consist of two steps: partitioning ALS point clouds into several segments and removing off-ground segments using the properties, such as shape, size and completeness [9,10]. Common segmentation methods include RANSAC, the slope, and the smoothness constraint. These approaches obtain promising performance under urban environments. However, it is still a difficult task for these methods to filter ground points from complex forest areas.

3. METHOD

Given N points set $P = \{p_1, p_2, \ldots, p_N$, where $p_i = (x_i, y_i, z_i)$, $i = 1, 2, \ldots, N\}$, the ground filtering task can be considered as a binary semantic segmentation problem. Our aim is to search a mapping, F, to map P into the corresponding label vector $V = \{v_1, v_2, \ldots, v_N, v_i = (v_i^1, v_i^2)$, i=1, 2, …, N, v_i^1, $v_i^2 = \{0,1\}\}$. In our work, the map F is designed as a modified GCN. As shown in Fig. 1, the proposed ground filtering method consists of three modules (represented by different colors). The first module is the local topological information (LTI) layer (see Fig. 2(a)), which provides local features from neighbor points. The second and third modules are the GCN based networks that designed to describe the global and local features, respectively.

The LTI layer is the core part. It computes two important features: the local covariance information and the dimensional features. Specifically, for each point, 20 neighbor points are searched using K-Nearest Neighbors (KNN) search algorithm and the covariance matric, denoted as $C = [c_{i,j}]_{3\times3}$, can be calculated. Because the covariance matric is a real symmetric matrix, the six upper triangle elements is selected as covariance features:

$$\{c_{11}, c_{12}, c_{13}, c_{22}, c_{23}, c_{33}\}. \tag{1}$$

In addition, dimensionality features [11,12], $\{a_{1D}, a_{2D}, a_{3D}\}$, are chosen as extra geometric information. Dimensionality features are defined as:

$$a_{1D}=\frac{\sqrt{\lambda_1}-\sqrt{\lambda_2}}{\sqrt{\lambda_1}},\quad a_{2D}=\frac{\sqrt{\lambda_2}-\sqrt{\lambda_3}}{\sqrt{\lambda_1}},\quad a_{3D}=\frac{\sqrt{\lambda_3}}{\sqrt{\lambda_1}} \tag{2}$$

where λ_1, λ_2, and λ_3 ($\lambda_1 \geqq \lambda_2 \geqq \lambda_3$) are eigenvalues generated by Principal Component Analysis (PCA) on local neighbor point set. Then, the output of LTI layer and raw point clouds are merged using a concatenation operation. Finally, an edge convolution with two layers is performed to obtain the final output, a tensor with shape of $N \times 64$.

The second module is designed to mine the global feature. An edge convolution with two layers is used to extract the detail features for each point to improve the overall descriptiveness of the filtering model. As a fuse stage, the third module is used to integrate the local and global features. Besides, considering the fact that the number of off-ground points is often greater than that of ground points, we modified the regular loss function in the local and global features. Besides, considering the fact that the number of off-ground points is often greater than that of ground points, we modified the regular loss function in GCN by applying the focal loss [13], to achieve better training performance.

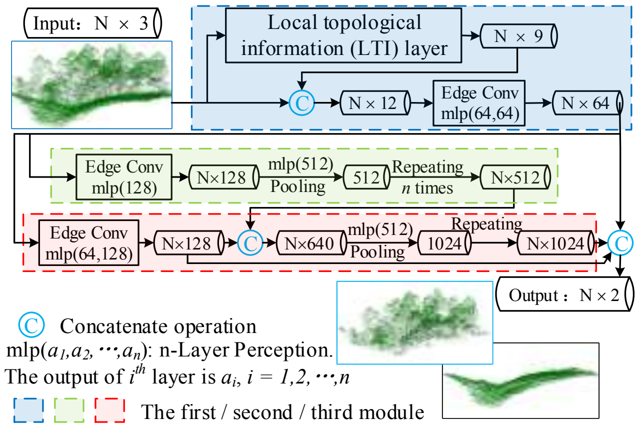

Fig. 1. Flowchart of ground filtering method. It is a modified GCN framework with three modules. The first module is the local topological information (LTI) layer, which provides local features from neighbor points. The second and third modules are the GCN based networks that designed to describe the global and local features, respectively.

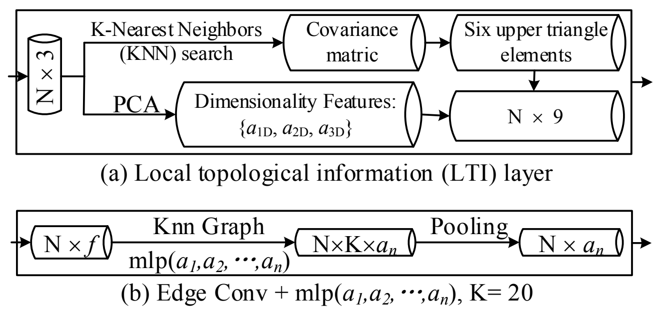

Fig. 2. The frameworks of LTI layer and Edge Conv layer.

In summary, compared with the original GCN, our modified model enjoys several advantages. Firstly, the design of LTI layer provides more detail information. Local spatial relationship between neighbor points is fully utilized and geometric properties are preserved in the input data. Secondly, the combination of local and global features enhances the descriptiveness of the modified model. Different from the original GCN, extra global features in our modified model compensate for the deficiencies in the overall characterization. Thirdly, the use of focal loss down-weights the loss assigned to well-classified samples (in our work, for example, a point with a very high or low z value will be a well-classified sample) and pay more attention on the hard samples (e.g. the bushes). Therefore, under the guidance of this loss function, the model can be trained more efficiently in the direction of convergence.

4. RESULTS AND DISCUSSION

4.1 Dataset and implementation

The dataset was acquired with a ALS system, which consists of a lightweight (3.5 kg) and compact (227×180×125 mm) laser scanner and an unmanned aerial vehicle (UAV). Seven areas with complex forests were selected for our experiments. Fig. 3 shows three samples. Obviously, all these samples contain mixed noise, especially unordered outliers

Table. 2. Comparing ground filtering results generated by different methods

Method	Metric (%)	Area1	Area2	Area3	Area4	Area5	Area6	Area7	Means	Max	Min
CSF	I error	2.49	2.04	**2.01**	**4.57**	3.31	6.53	**3.54**	**3.49**	6.53	2.01
	II error	9.34	9.11	8.66	11.85	14.42	16.6	12.25	11.74	16.6	8.66
	Total error	6.00	5.88	5.03	8.93	8.45	12.28	8.53	7.87	12.28	5.03
	k	87.52	87.75	89.12	81.86	82.3	74.75	81.46	83.53	89.12	74.75
PointNet	I error	4.28	2.58	4.32	9.18	2.97	8.98	8.83	5.88	9.18	2.58
	II error	**9.14**	6.22	4.06	8.51	15.41	11.26	5.95	**8.65**	15.41	4.06
	Total error	5.92	4.35	4.24	8.96	7.68	10.75	7.15	7.00	10.75	4.24
	k	87.21	90.76	90.32	81.64	82.48	77.73	84.29	84.91	90.76	77.73
DGCNN	I error	4.12	2.00	3.20	7.68	2.82	8.97	8.20	5.28	8.97	**2.00**
	II error	9.62	5.92	4.02	**8.34**	**13.31**	**11.25**	**5.81**	8.32	13.31	4.02
	Total error	6.01	3.91	**3.43**	8.15	6.82	10.73	6.78	6.54	10.73	3.43
	k	86.99	91.69	**92.13**	83.30	84.49	77.75	85.09	85.92	92.13	83.30
Ours	I error	**2.39**	**1.85**	4.23	6.86	**2.09**	**5.66**	7.05	4.30	7.05	1.85
	II error	10.49	**5.85**	**3.71**	9.09	14.76	15.15	7.03	9.44	15.15	**5.85**
	Total error	**5.41**	**3.77**	4.00	**7.91**	**6.79**	**10.59**	**6.80**	**6.46**	10.59	**3.77**
	k	**88.14**	**91.98**	90.85	**83.65**	**84.47**	**77.84**	**85.13**	**86.01**	**91.98**	83.65

that bring huge difficulty for detecting individual trees. Several key statistical information is presented in Table 1.

Our method was trained with TensorFlow on a NVIDIA Tesla P100 GPU. The batch size was set to 8 and the initial learning was 0.001. When training the model, we used adaptive moment estimation (Adam) with a momentum of 0.9. The number of epochs was 50.

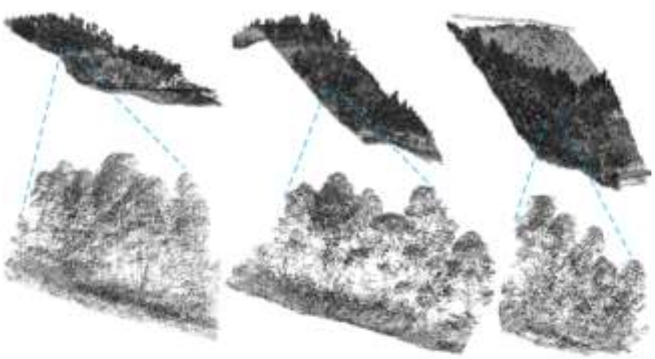

Fig. 3. Samples from selected areas. These forests are extremely noisy.

Table. 1. Key statistical information of seven study areas . P.N. and Alti. are Point Number and Altitude, respectively.

	A1	A2	A3	A4	A5	A6	A7
Size (m^2)	20,231	12,495	11,708	13,168	12,636	15,115	16,110
P. N.	4,498,425	4,268,649	3,655,193	3,898,510	3,426,171	3,976,401	2,595,690
Max (m)	212	212	210	208	201	181	159
Min (m)	106	107	124	115	123	95	95
Alti. (m)	106	105	86	93	78	86	64

4.2 Experimental results

We conducted experiments on the above datasets and compared our method with previous ground filtering algorithms, including the deep learning based methods, PointNet [14] and DGCNN [15], as well as the classic method, the cloth simulation filtering (CSF) proposed in [16]. We used four metrics for quantitative analysis [17]. Type I error measures the rate of ground points mislabeled as off-ground points, while Type II error represents the percentage of off-ground points mislabeled as ground points. Total error shows the rate of all mislabeled points. Besides, the Cohen's kappa coefficient (k) measures the inter-ratio agreement more robustly. It also needs to point out that, as our method is a supervised method, training data is required when it is used to test one area. Therefore, in our experiments, when an area is used as testing data, other six areas are considered as training data.

Table 2 presents the compared results. Obviously, our method achieves excellent performance. More specifically, our method obtains the best results on areas 2. Besides, in areas 1, 4, 5, 6 and 7, our method also achieves competitive results. Additionally, compared to CSF and PointNet, our method increases the performance by 2.47%and 1.08% in overall average k coefficient, respectively. These results mean that our method can filtering the ground points more precisely. On the overall average total error and Type II error, our method also has the obvious advantage, with a significant reduction comparing with other three methods. That means our method can reject object points more effectively and has a smaller proportion of all error points. However, compared with other methods, our method has a limitation that the Type I error cannot be greatly reduced. Our future work will focus on this issue.

The reason why our method achieves the better performance is that the local graph structure using in the proposed ground filtering framework can preserve the relationship among neighbor points. Compared with PointNet and DGCNN, the local features of ground points are captured by our method more effectively. Besides, compared with CSF, as a non-parametric method, our method does not need to find the optimal parameters, which means fewer mistakes caused by manual intervention.

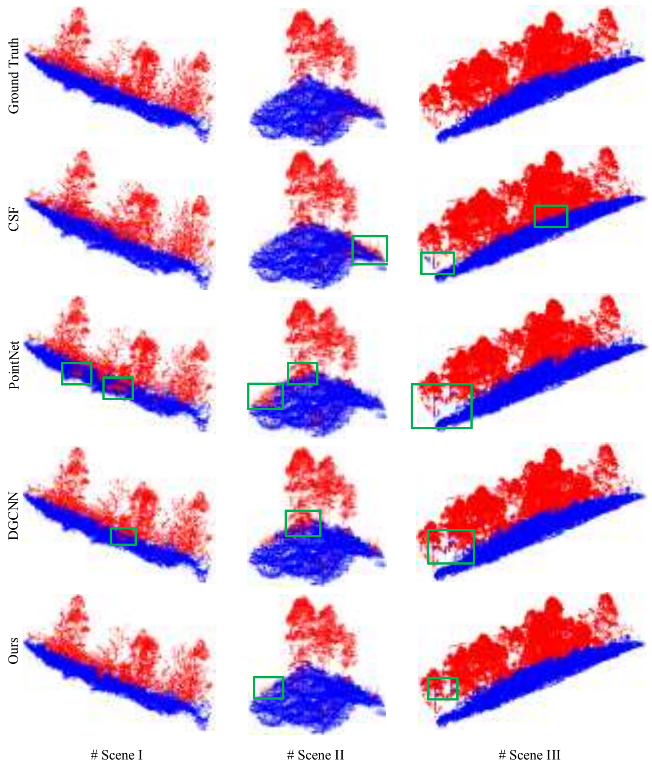

Fig. 4. Results of ground filtering generated by different methods on several scenes. The ground and off-ground points are marked by blue and red color, respectively. The areas marked by the green boxes are areas with mislabeling error points .

To further present the comparative results, we visualized the ground and off-ground points on some areas after filtering by using CSF and our method. As shown in Fig. 4, our method achieves better performance than CSF in preserving ground points. Especially, the improvements of our method are more significant in steep areas, as shown on the first two lines of Fig. 4. Note that filtering ground points in steep areas is another challenge [17]. Our method also labels the ground points as off-ground points in some areas, just as shown in the last line of Fig. 4. However, the number of mislabeling points is very low and acceptable in practice.

5. CONCLUDING REMARKS

In this paper, we provide a three-module based GCNs to improve the filtering performance in abrupt slope. As the key part of our method, a well-designed local topological information (LTI) module is presented to effectively extract the local features. Compared with other existing works, our ground filtering method can obtain better ground filtering performance and preserve more complete information contained in off-ground points.

ACKNOWLEDGEMENTS

This work was supported in part by the National Natural Science Foundation of China under Grant 41871380.

REFERENCES

[1] J. White, N. Coops, M. Wulder, M. Vastaranta, T. Hilker, and P. Tompalski, “Remote sensing technologies for enhancing forest inventories: A review,” *Can. J. Remote Sens.*, vol. 42, no. 5, pp. 619–641, 2016.

[2] Z. Luo, J. Li, Z. Xiao, G. Mou, X. Cai, and C. Wang, “Learning high-level features by fusing multi-view representation of MLS point clouds for 3d object recognition in road environments,” *ISPRS-J. Photogramm. Remote Sens.*, vol. 150, pp. 44–58, 2019.

[3] Z. Gong, H. Lin, D. Zhang, Z. Luo, J. Zelek, Y. Chen, A. N. Nurunnabi, C. Wang, J. Li, “A Frustum-based probabilistic framework for 3D object detection by fusion of LiDAR and camera data,” *ISPRS-J. Photogramm. Remote Sens.*, vol. 159, pp. 90-100, 2020.

[4] Z. Cui, K. Zhang, C. Zhang, S. Chen, “A cluster-based morphological filter for geospatial data analysis,” *in: ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data*, pp. 1–7, 2013.

[5] D. Mongus, N. Lukac, B. Zalik, 2014. “Ground and building extraction from lidar data based on differential morphological profiles and locally fitted surfaces,” *ISPRS-J. Photogramm. Remote Sens.*, vol. 93, pp. 145–156, 2014.

[6] G. Sithole, “Filtering of laser altimetry data using a slope adaptive filter,” *ISPRS Archives*, vol. 34, pp. 203–210, 2001.

[7] A. Nurunnabi, G. West, D. Belton, 2016. “Robust locally weighted regression techniques for ground surface points filtering in mobile laser scanning three dimensional point cloud data,” *IEEE Trans. Geosci. Remote Sens.*, vol. 54, pp. 2181–2193, 2016.

[8] S. Xu, G. Vosselman, S. O. Elberink, “Multiple-entity based classification of airborne laser scanning data in urban areas,” *ISPRS-J. Photogramm. Remote Sens.*, vol. 88, pp. 1–15, 2014.

[9] G. Sithole, G. Vosselman, “Filtering of airborne laser scanner data based on segmented point clouds,” *ISPRS Archives*, vol. 36(W19), pp. 66-71, 2005.

[10] B. Yang, R. Huang, Z. Dong, Y. Zang, J. Li, “Two-step adaptive extraction method for ground points and breaklines from lidar point clouds,” *ISPRS-J. Photogramm. Remote Sens.*, vol. 119, pp. 373–389, 2016.

[11] A. Gressin, C. Mallet, J. Demantke, N. David, “Towards 3D LiDAR point cloud registration improvement using optimal neighborhood knowledge,” *ISPRS-J. Photogramm. Remote Sens.*, vol. 79, pp. 240–251, 2013.

[12] H. Lin, J. Chen, P. Su, C. Chen, “Eigen-feature analysis of weighted covariance matrices for LiDAR point cloud classification,” *ISPRS-J. Photogramm. Remote Sens.*, vol. 94, pp. 70–79, 2014.

[13] T. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, “Focal loss for dense object detection,” *In: ICCV 2017*, pp, 318-327, (doi: 10.1109/ICCV.2017.324), 2017.

[14] C. R. Qi, H. Su, K. Mo, L. J. Guibas, “PointNet: Deep learning on point sets for 3D classification and segmentation,”. *In: IEEE CVPR2016*, pp. 77–85, 2016.

[15] Y. Wang, Y. Sun, Z. Liu, S. Sarma, M. Bronstein, J. Solomon, “Dynamic graph CNN for learning on point clouds,” *ACM Trans. Graph.*, vol, 38(5), pp. 146.1–146.12, 2019.

[16] W. Zhang, J. Qi, W. Peng, H. Wang, D. Xie, X. Wang, G. Yan, “An easy-to-use airborne lidar data filtering method based on cloth simulation,” *Remote Sens.,* vol. 8, pp. 501.1–501.22, 2016.

[17] G. Sithole, G. Vosselman, “Experimental comparison of filter algorithms for bare-earth extraction from airborne laser scanning point clouds,” *ISPRS-J. Photogramm. Remote Sens.*, vol. 59, pp. 85–101, 2004.