Semantic segmentation of LiDAR points clouds: Rasterisation beyond Digital Elevation Models

Florent Guiotte, Minh-Tan Pham, Romain Dambreville, Thomas Corpetti, Sébastien Lefèvre

Semantic segmentation of LiDAR points clouds: Rasterization beyond Digital Elevation Models

Florent Guiotte, Minh-Tan Pham, Romain Dambreville, Thomas Corpetti, Sébastien Lefèvre

***Abstract*—LiDAR point clouds are receiving a growing interest in remote sensing as they provide rich information to be used independently or together with optical data sources such as aerial imagery. However, their non-structured and sparse nature make them difficult to handle, conversely to raw imagery for which many efficient tools are available. To overcome this specific nature of LiDAR point clouds, standard approaches often rely in converting the point cloud into a digital elevation model, represented as a 2D raster. Such a raster can then be used similarly as optical images, e.g. with 2D convolutional neural networks for semantic segmentation. In this letter, we show that LiDAR point clouds provide more information than only the DEM, and that considering alternative rasterization strategies helps to achieve better semantic segmentation results. We illustrate our findings on the IEEE DFC 2018 dataset.**

***Index Terms*—LiDAR, DEM, rasterization, Deep Learning, Semantic Segmentation.**

I. Introduction

Thanks to their very high resolution, LiDAR point clouds are known to be of very high interest to identify complex structures, especially in urban environments, such as trees, road, cars... However, as related point clouds are voluminous and irregularly distributed, land cover mappings are in many studies often simplified to a digital elevation model (DEM) used as additional information for fusion with multispectral or hyperspectral images.

In this study, we start from the idea that this rasterization step related to the production of a single DEM is not optimal as many additional information embed in the LiDAR point cloud is lost. We then rather prefer to focus on the extraction of more advanced rasterized maps. In a first attempt with the same dataset and the same evaluation protocol [3], we computed a series of attribute profiles on 2D grids containing various information extracted during the mapping from 3D to 2D (number of points in a cell, first echo, last echo, ...); these features have fed a simple Random Forest classifier with efficient results. In this paper, we extend this work by considering a deep learning network.

More than designing the most adapted network, our aim is rather to show that LiDAR point clouds provide more information than only the DEM, and that considering alternative rasterization strategies helps to achieve better semantic segmentation results. We illustrate our findings on the IEEE DFC 2018 dataset.

The paper is organized as follows. We review related works in Sec. II. We then present in Sec. III various strategies that can be employed to map a 3D point cloud into a 2D raster. After recalling the deep network architecture used in this paper (Sec. IV), we report in Sec. V the outcomes of our experiments conducted on the IEEE DFC 2018 dataset. We finally conclude the paper in Sec. VI.

F. Guiotte and T. Corpetti are with LETG, Université Rennes 2.

M.-T. Pham, R. Dambreville and S. Lefèvre are with IRISA, Université Bretagne Sud.

II. Related work

LiDAR point clouds have become a popular remote sensing data source for land cover mapping. Recent developments have allowed precise point cloud segmentation, especially using deep networks [5], [6]. However large point clouds like those provided by airbone LiDAR are more challenging for direct end to end learning because of the large amount of data and their unstructured nature, as opposed to regular 2D grids in images [4]. Therefore, to address this problem, many authors either suggest to reorganise the point cloud into regular 2D grids and/or to exploit the multispectral information. These directions are detailed in the following.

As a matter of fact, recent LiDAR sensors now provide multispectral signals, through the generation of a point cloud specific to a given wavelength. A few studies have been reported with such data. We can mention [9] which showed that using dual wavelength led to substantial improvements in land cover mapping w.r.t. a single wavelength. In [10], a multi-spectral LiDAR system was used to classify ground with pattern matching classifier applied pointwise on intensities and NDVI.

Another possibility consists in using various rasters computed on a LiDAR point cloud. Let us note that this idea is not totally new, and a few recent attempts have been made in this direction, as for example [8] were the DFC 2018 dataset is classified with LiDAR only or together with other optical data. Several features are extracted based solely on LiDAR information (e.g. median to altitude, intensity and number of echoes) or combined with other information (composition of spectral intensities, intensity ratio, brightness, difference between DSM, etc), and/or with local features computed solely on intensity. Experiments were conducted using several classifiers: random forest (RF), gradient boosting machine and CNN on 20 classes. While this study had shown the relevance of combining multiple data sources to process the DFC 2018 dataset, it did not allow to derive any conclusion regarding the relevance of alternative LiDAR rasters and their specific performance with deep semantic segmentation networks.

In a very recent study [3], we have shown that such alternative LiDAR rasterizations actually provide additional information source that can help to describe the contents of

a remotely-sensed scene, and improve its classification. This was demonstrated using well-established multilevel features (attribute profiles) and classifier (RF). Nevertheless, given the widely-recognized performance of deep learning for semantic segmentation, the interest of such rasters as inputs to a deep network has still to be demonstrated. This is the goal of this letter and the next section discusses about rasterization strategies.

III. Rasterization of unstructured point clouds

The main benefits of LiDAR rasterization are:

- To **reduce the complexity** since data are represented on a regular grid;
- To provide a **regular sampling** (easier to manipulate neighbours) instead of dealing with irregular point clouds;
- To have a prior **known** number of data unlike unknown number of point clouds;
- To **reduce the radiometric and altimetric artefacts** thanks to the aggregation of values.

In general the use of a DEM only is not optimal since LiDAR systems enable to capture complex patterns in the three dimensions and a DEM aggregates the vertical information. This loss in the vertical direction is prejudicial since many urban objects are characterised by their vertical structure. For example in vegetated areas, the information in the vertical direction enables to capture the whole trees structure (and not only their surface). In addition, the vertical information detects ground below vegetation or objects below trees such as residential buildings, roads and cars. Therefore, we suggest here to provide rasters where such information in the vertical direction is kept. In comparison with other works [8], we tried to summarise this vertical component by creating several feature maps based on the vertical distribution, in addition to usual DEMs. The general rasterization process can be defined in three steps detailed below:

1- The reorganisation of the point clouds by binning them into a regular grid. More formally, we apply a transformation $\mathcal{PR}_{h,f}$ (for "points to raster", associated with a discretization step h and an information function f) defined on the dataset $\boldsymbol{X}$ as:

$$\begin{aligned} \mathcal{PR}_{h,f} : \mathbb{R}^3 \times \mathbb{R} &\longrightarrow \boldsymbol{E}_h \times \mathbb{R} \\ \{x,y,z,\mathcal{I}\} &\longmapsto \{i,j,\mathbf{I}(\boldsymbol{X}_{i,j})\} \text{ with:} \\ &\begin{cases} \boldsymbol{X}_{i,j} \text{ the set of points s.t.} \\ i \text{ s.t. } x_m + ih \leq x < x_m + (i+1)h \\ j \text{ s.t. } y_m + jh \leq y < y_m + (j+1)h \\ \mathbf{I}(\boldsymbol{X}_{i,j}) = f(i,j,\boldsymbol{X}_{i,j},\mathcal{I}) \end{cases} \end{aligned} \tag{1}$$

with $\boldsymbol{E}_h$ the raster grid, x_m and y_m the minimum values of all points x and y in the dataset $\boldsymbol{X}$. The rule of function f is to associate to each cell location (i,j) an information related to the data point $\boldsymbol{X}_{i,j}$ included in the cell. Its value is discussed below.

2- The extraction of LiDAR feature maps. Many functions f can be defined to provide rasters. For example a DEM high and a DEM low uses respectively the positions of the maximum and the minimum z coordinates i.e., first and last returns inside a cell $\boldsymbol{X}_{i,j}$:

$$\begin{aligned} f_{Dh}(i,j,x,y,z,\mathcal{I}) &= \max(z) \text{ s.t } (x,y,z) \in \boldsymbol{X}_{i,j} \\ f_{Dl}(i,j,x,y,z,\mathcal{I}) &= \min(z) \text{ s.t } (x,y,z) \in \boldsymbol{X}_{i,j} \end{aligned} \tag{2}$$

To enable more flexibility, other functions can be used such as the intensity of the highest and lowest points:

$$\begin{aligned} f_{Ih}(i,j,x,y,z,\mathcal{I}) = I(x^p,y^p,z^p),\ p \text{ being the point s.t.} \\ z^p = \max(z)_{(x,y,z)}\boldsymbol{X}_{i,j} \end{aligned} \tag{3}$$

$$\begin{aligned} f_{Il}(i,j,x,y,z,\mathcal{I}) = I(x^p,y^p,z^p),\ p \text{ being the point s.t.} \\ z^p = \min(z)_{(x,y,z)}\boldsymbol{X}_{i,j} \end{aligned} \tag{4}$$

or the number of echoes per cell:

$$f_N(i,j,x,y,z,\mathcal{I}) = |\boldsymbol{X}_{i,j}| \tag{5}$$

3- The interpolation of empty cells. If the discretization step h is small, empty bins are likely to appear. In this work, we fill them using a linear interpolation.

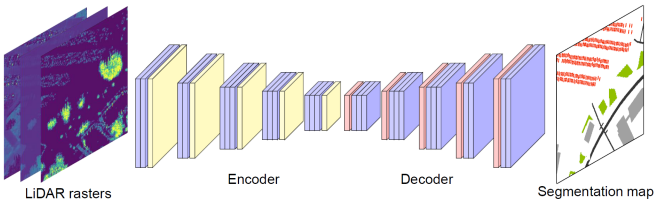


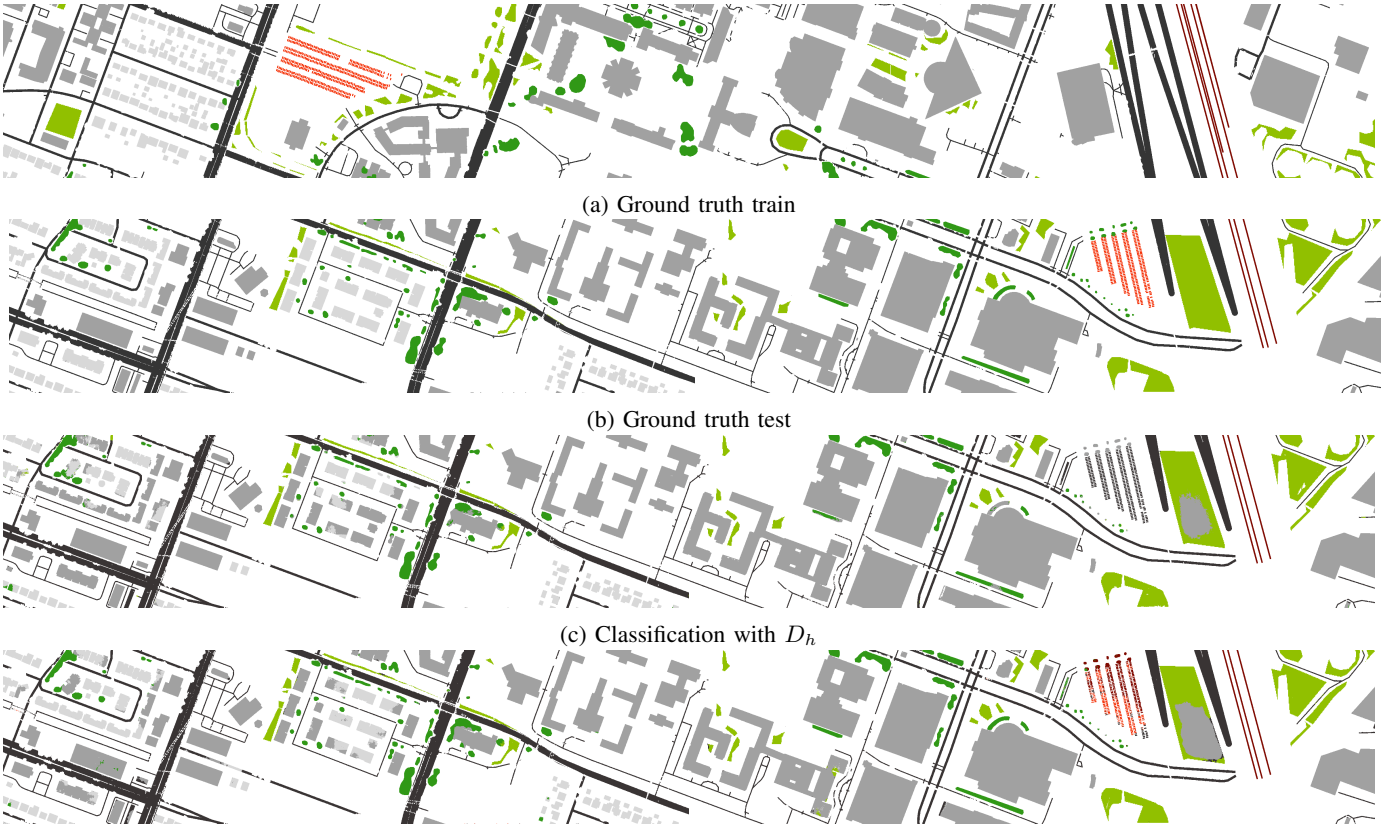
Fig. 1: Overview of the SegNet architecture with LiDAR rasters as input.

IV. Neural network

Deep learning approaches and particularly deep convolutional neural networks are currently unrivalled at the top of the state of the art for semantic segmentation applications. To evaluate the pertinence of our different rasters, we exploited the SegNet model [2] which has been widely used for semantic segmentation in computer vision domain. In remote sensing, this network has also proved its effectiveness on multispectral images with visible (RGB) and infrared bands in [1]. The SegNet model relies on an encoder-decoder architecture based on convolutional layers of the VGG-16 network [7], followed by batch normalization, rectified linear unit (ReLU) and then pooling and unpooling layers (w.r.t the encoder and decoder, respectively) [2]. The input of SegNet has usually three channels by default. In our work, not only each of LiDAR rasters but also their combinations will be used as input of the network, which allows us to exploit complementary information from those rasters for better segmentation results.

V. Experiments

We first introduce the dataset and the LiDAR rasters we used before discussing the efficiency of the proposed rasters.



(a) Ground truth train

(b) Ground truth test

(c) Classification with D_h

(d) Classification with I_h, I_l, D_h

Fig. 2: Illustration: Horizontal disjoint split between training set (a) and test set (b); (c) Classification result using DEM (D_h) only; (d) Classification result using the combination of $\{I_h, I_l, D_h\}$.

A. Dataset

We chose the multi-spectral LiDAR acquisition of the University of Houston issued from 2018 IEEE GRSS Data Fusion Contest dataset [11] to support our experiments. The associated ground truth map has a spatial resolution of $0.5\,\mathrm{m}$. The original 20 classes have been reduced to 7 generic urban classes (roads, grass, trees, residential buildings, non-residential buildings, cars and trains) to evaluate the overall accuracy.

B. LiDAR feature maps

To generate our rasters, the grid step was set to $h = 0.5\,\mathrm{m}$ to fit with the ground truth. We removed the first and last 0.1 percentiles of the point cloud based on elevation distribution since they are more likely to be outliers.

We chose to gather the geometric information contained in the 3 wavelengths to get a dense point cloud. With this composite point cloud we created several elevation rasters:

- Highest and lowest point in the cell (i.e. DEM and "reversed" DEM), noted D_h, D_l and corresponding to the use of function f_{Dh} (2));

Then, with each point cloud separately we created intensity and echo rasters:

- Intensity of the highest and lowest points in the cell (noted I_h, I_l and corresponding to function (3));
- Number of echoes per beam in the cell (noted N and corresponding to function (5)).

C. Classification

Training phase: The dataset has been divided into a training set and a test set with an horizontal split from the original data (*disjoint split*) as shown on Fig.2-(a)-(b). We used the code from [1][1] to perform all the experiments with parameter setting as default in [1] (learning rate 0.01 with momentum 0.9 and weight decay 5×10^{-4}) for a fair comparison. As the input size of our SegNet model varies w.r.t. the rasters or different feature combinations, the network was trained from scratch. During the training, we randomly extracted 256×256 image patches from the training set. Batch size was set to 16 and all experiments were stopped after 20 epochs.

D. Experimental results

In table I, we present the accuracy per-class, the Average Accuracy (AA), Overall Accuracy (OA) and Cohen's Kappa coefficient (κ) for each feature (see section V-B) and their combination. As one can observe, the use of a DEM only is globally far from optimal (except for grass and roads where this value is really meaningful while other features slightly disturb the identification) despite the fact that most studies

[1] https://github.com/nshaud/DeepNetsForEO

		Per-class accuracy (%)							Evaluation metrics		
		Grass	Trees	Residential	Non-res building	Roads	Cars	Trains	AA(%)	OA(%)	$\kappa(\times 100)$
1	D_h	**95.62**	77.66	44.18	98.72	**99.83**	0.09	100.00	73.72	90.13	85.24
2	N	16.08	92.48	46.50	97.63	95.39	6.50	100.00	64.94	87.27	78.37
3	I_l	56.94	89.35	41.96	**99.39**	97.16	98.81	99.94	83.36	86.98	80.10
4	D_l	90.93	89.67	22.09	99.12	99.57	0.00	99.94	71.57	87.67	81.31
5	I_h	59.82	95.15	47.78	98.27	96.43	**99.98**	99.96	85.34	87.54	81.14
6	$\{N, I_h, D_h\}$	93.15	**97.72**	39.47	98.99	99.66	13.42	100.00	77.49	90.66	86.01
7	$\{I_h, I_l, D_h\}$	81.64	94.12	76.95	98.36	99.38	57.10	100.00	**86.87**	**93.88**	**91.00**
8	$\{I_h, I_l, D_h, D_l\}$	87.63	92.84	60.20	93.32	96.98	68.16	100.00	86.44	92.31	88.57
9	$\{N, I_h, I_l, D_h, D_l\}$	79.13	95.35	**82.70**	96.28	99.19	10.05	100.00	80.38	92.42	88.97

TABLE I: Per-class accuracy, average accuracy (AA), overall accuracy (OA) and Cohen's Kappa coefficient (κ) for each feature and combination of them using Segnet.

exploit only this property when rasterizing LiDAR point clouds. The use of the last echos (position D_l and intensity I_l) enable to greatly improve the classification. These echos are related to structures behind vegetated areas and provide very relevant information, as noticed on the classification results. As for the number of echos N, its value combined with other features enables to discriminate more properly only trees (where many echos are included) and residential areas (where only one echo is present) but it does not improve the overall classification in our experiments. Finally, the combination of the DEM, first and last intensities enables to provide the best classification results. This demonstrates the fact the LiDAR data are very rich and are currently not optimally exploited when they are rasterized in a DEM only.

VI. Conclusion

In this letter, we explored the use of alternative rasters (beyond the standard DEM) to classify LiDAR point clouds. We measured the performance of a well-established deep neural network for multispectral semantic segmentation with different rasters extracted from the multispectral LiDAR point cloud provided with the IEEE DFC 2018.

Our results show that the DEM is not the most discriminative feature, and that alternative features can be more helpful for land cover mapping.

Furthermore, an advantage of our map-based method is to allow us to rely on image (raster) segmentation networks with no or very small adaptation effort, instead of requiring to design specific networks dedicated to point clouds. Since semantic segmentation of images is a very active topic in computer vision, our approach will allows LiDAR processing tasks to benefit from future developments in the field.

Among future works, we would like to see if combining the different features in a same network leads to better results. Indeed, it is a promising direction given our preliminary results with non-deep learning techniques [3]. Furthermore, we plan to investigate deep architectures among those well-established for semantic segmentation.

Acknowledgment

The authors would like to thank the National Center for Airborne Laser Mapping and the Hyperspectral Image Analysis Laboratory at the University of Houston for acquiring and providing the data used in this study, and the IEEE GRSS Image Analysis and Data Fusion Technical Committee.

The authors would like to thank the helpful comments of the anonymous reviewers of this article.

References

[1] Nicolas Audebert, Bertrand Le Saux, and Sébastien Lefèvre. Beyond rgb: Very high resolution urban remote sensing with multimodal deep networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 140:20–32, 2018.

[2] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

[3] Florent Guiotte, Sébastien Lefèvre, and Thomas Corpetti. Rasterization strategies for airborne LiDAR classification using attribute profiles. In *2019 Joint Urban Remote Sensing Event (JURSE)*, pages 1–4. IEEE, 2019.

[4] Loïc Landrieu and Martin Simonovsky. Large-scale Point Cloud Semantic Segmentation with Superpoint Graphs. *CoRR*, abs/1711.09869, 2017.

[5] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017.

[6] Gernot Riegler, Ali Osman Ulusoy, and Andreas Geiger. OctNet: Learning Deep 3d Representations at High Resolutions. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6620–6629, Honolulu, HI, 2017. IEEE.

[7] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[8] Sergey Sukhanov, Dmitrii Budylskii, Ivan Tankoyeu, Roel Heremans, and Christian Debes. Fusion of Lidar, Hyperspectral and RGB Data for Urban Land Use and Land Cover Classification. In *IEEE International Geoscience and Remote Sensing Symposium*, pages 3864–3867, 2018.

[9] Cheng-Kai Wang, Yi-Hsing Tseng, and Hone-Jay Chu. Airborne Dual-Wavelength LiDAR Data for Classifying Land Cover. *Remote Sensing*, 6(1):700–715, 2014.

[10] V. Wichmann, M. Bremer, J. Lindenberger, M. Rutzinger, C. Georges, and F. Petrini-Monteferri. Evaluating the potential of multispectral airborne lidar for topographic mapping and land cover classification. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3/W5:113–119, 2015.

[11] Y. Xu, B. Du, L. Zhang, D. Cerra, M. Pato, E. Carmona, S. Prasad, N. Yokoya, R. Hänsch, and B. Le Saux. Advanced Multi-Sensor Optical Remote Sensing for Urban Land Use and Land Cover Classification: Outcome of the 2018 IEEE GRSS Data Fusion Contest. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(6):1709–1724, 2019.