Attribute Filtering of Urban Point Clouds Using Max-Tree on Voxel Data

Florent Guiotte, Sébastien Lefèvre, Thomas Corpetti

Attribute filtering of urban point clouds using max-tree on voxel data*

Florent Guiotte[1,2], Sébastien Lefèvre[2], and Thomas Corpetti[1]

[1] Univ. Rennes 2, LETG, 35000 Rennes, France
{florent.guiotte,thomas.corpetti}@univ-rennes2.fr
[2] Univ. Bretagne Sud, IRISA, 56000 Vannes, France
sebastien.lefevre@irisa.fr

Abstract. This paper deals with morphological characterization of unstructured 3D point clouds issued from LiDAR data. A large majority of studies first rasterize 3D point clouds onto regular 2D grids and then use standard 2D image processing tools for characterizing data. In this paper, we suggest instead to keep the 3D structure as long as possible in the process. To this end, as raw LiDAR point clouds are unstructured, we first propose some voxelization strategies and then extract some morphological features on voxel data. The results obtained with attribute filtering show the ability of this process to efficiently extract useful information.

Keywords: Point clouds · Max-tree · Rasterization · Voxel · Attribute filtering.

1 Introduction

Thanks to the advances both in technologies such as laser scanning (LiDAR) and in methods from photogrammetry, digital point clouds form a popular object of study in many scientific fields such as geosciences (flow, erosion, rock deformations, ...), computer graphics (3D reconstruction), urban environments analysis from Earth Observation (detection of trees, roads, buildings, ...). In the context of urban scenes, they provide a rich 3D information w.r.t. digital 2D photographs.

Despite their growing interest, only limited studies have explored how to apply mathematical morphology on point clouds [2, 13]. The usual approach remains to first rasterize the point cloud to obtain a digital image (also called a raster of pixels) on which standard morphological operators are applied [18]. This strategy was also recently followed for morphological hierarchies on LiDAR point clouds [8].

In this paper, we claim that discretizing a point cloud into a 2D raster leads to an oversimplification of the image that greatly reduces the potential of morphological operators. Thus, we consider here a 3D discretization in a voxel grid

* This work was financially supported by Région Bretagne (CAMLOT doctoral project).

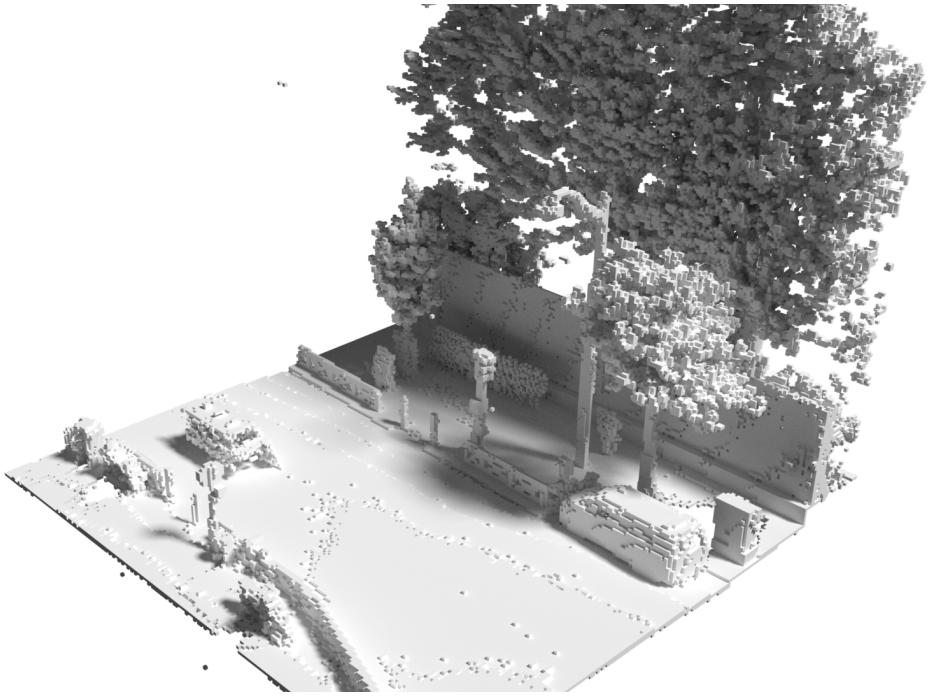


Fig. 1. LiDAR data projected into a voxel grid.

as illustrated in Figure 1. Such an approach allows us to benefit from efficient algorithms that have been introduced for morphological hierarchies, while still maintaining the 3D information. We illustrate our solution with a very popular processing, namely attribute filtering, that is applied on an urban point cloud. The reported results show the relevance and the potential of this approach.

The paper is organized as follows. In Section 2 we review morphological approaches (especially those based on hierarchies) for point clouds and 3D processing. We then explain the different steps of our method in Section 3, before illustrating it in the urban remote sensing context in Section 4. Section 5 concludes this manuscript and provides directions for future research.

2 Related Work

2.1 Mathematical morphology on point clouds

Conversely to digital images that are usually defined on a discrete 2D grid, a point cloud is characterized by a sparse set of points defined with continuous coordinates. So dealing with a 3D point cloud raises many issues including the lack of efficient processing algorithms. One of the most critical questions is the definition and fast computation of the neighborhood of a point, which is a fundamental concept in morphology.

As already stated, the usual approach is to project the 3D point cloud in a discrete grid (either 2D or 3D) where the neighborhood computation becomes straightforward. Thus, LiDAR point clouds were considered in [6] to characterize vegetation (trees). The authors use the point density to define voxel values, that are further processed with 3D adaptation of standard 2D morphological methods. In the astronomical context, the discretization proposed by [4] also relies on the density, this latter being estimated using adaptive kernel. A max-tree is then used to find local maxima that allow for identification of relevant subspaces for clustering the data.

More closely related to our study, a few attempts have been made to process urban point clouds. In [1], the segmentation and classification of an urban point cloud is achieved by means of super-voxels. They are created using a distance computed in the feature space between voxel properties such as distribution (by mean, variance) of spectral values (intensity or color information). Finally, a reprojection step is involved to obtain the resulting labeled point cloud. The same tasks are addressed by [18], with a different approach though. The point cloud is projected in a digital elevation model (DEM) before applying 2D morphology and classification. The results are then reprojected into a point cloud. More recently, the same authors have addressed segmentation of facades with attribute profiles [17]. Combination of DEM morphology and attribute filters (elongation) were considered on binary 3D images denoting an occupancy grid. Beyond these works on 2D and 3D rasters, a few works have been conducted directly on the continuous space of 3D points cloud, such as [2, 13]. However the morphological methods introduced in these papers are dedicated to point clouds describing surfaces, and as such cannot be used with LiDAR data since a surface can not be reconstructed in each situation (points are also likely to belong to the inside of objects, for example in vegetation areas).

2.2 Morphological hierarchies on 3D images

The extension of morphological hierarchies (e.g. max-tree) from a 2D image to a 3D volume is rather straightforward. Unsurprisingly, it led to several works attempting to use it on 3D voxels, especially in the medical domain. Early work in [22] has for example used the max-tree to filter 3D images with volume and inertia attributes. Later on, the max-tree was used to filter and visualize the medical images [21] with three new 3D moment-based attributes (elongation, flatness and sparseness). Another 3D attribute was proposed in [10] to estimate the sphericity of objects. It was based on the computation of surface and volume of connected components and aims to be more efficient than the previous measures. Roundness of objects was estimated through another 3D attribute for max-tree filtering in [11]. Filtering medical images has also received a lot of attention until very recently, e.g. [3, 7, 20, 12]. Finally, we can mention the work of [5] to compute the tree of shapes of nD images.

3 Method

In order to filter point clouds using hierarchical representations, we propose to rely on a prior discretization of the continuous domain into a regular 3D voxel grid (instead of a 2D raster). This intermediate representation allows us to use directly mathematical morphology with the 3D data. We then reproject the results of the morphological filtering into the continuous domain (i.e. as a new point cloud).

3.1 From point cloud to voxel grid

A raw dataset $\boldsymbol{X}$ issued from LiDAR acquisitions lives in $\mathbb{R}^3 \times \mathbb{R}$ where each data $\boldsymbol{x} = \{x, y, z, \mathcal{I}\} \in \boldsymbol{X}$ is such that the intensity taken in location (x, y, z) is $\mathcal{I}$. Though very interesting, the irregularity of available locations (x, y, z) prevents from the use of tools devoted to structured data with ordering relations as images or volumes. To cope this difficulty, we suggest to transform this dataset $\boldsymbol{X}$ into a structured volume. This "voxelization" step aims at defining the data on a regular 3D grid $\boldsymbol{E}_h \subset \mathbb{N}^3$ with a given spatial resolution h (for the sake of simplicity, we consider here isotropic resolutions but the method can be applied with anistropic ones) such that the value taken in any point $(i, j, k) \in \boldsymbol{E}_h$ represents an information about initial data. This information can either be a boolean (related to the presence/absence of points in the voxel), the number of LiDAR points into the voxel, the average/standard deviation of associated intensities, the average/standard deviation of associated elevations, the majority label (for 3D labels), etc.

More formally, we apply a transformation $\mathcal{PV}_{h,f}$ (for "points to voxels", associated with a discretization step h and an information function f) defined as:

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

with x_m (resp. (y_m, z_m)) the minimum value of all points x (resp. y, z) in the dataset $\boldsymbol{X}$. The rule of function f is to associate to each voxel location (i, j, k) an information related to the original data points. Let us denote $\mathcal{I}_{i,j,k}$ the set of intensities $\mathcal{I}$ of points (x, y, z) inside a voxel (i, j, k) (i.e. fulfilling the 3 first conditions of (1)). Its cardinal is noted $|\mathcal{I}_{i,j,k}|$.

Many functions f can be defined, as for example:

- **Boolean** (noted f_b):

$$f_b(i, j, k, x, y, z, \mathcal{I}) = \begin{cases} 1 & \text{if } |\mathcal{I}_{i,j,k}| \geq 1 \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

– **Density** (noted f_d, similar to [4]):

$$f_d(i,j,k,x,y,z,\mathcal{I}) = |\mathcal{I}_{i,j,k}| \tag{3}$$

– **Empirical average intensity** (noted f_a):

$$f_a(i,j,k,x,y,z,\mathcal{I}) = \frac{1}{|\mathcal{I}_{i,j,k}|}\sum \mathcal{I}_{i,j,k} \tag{4}$$

– **Empirical standard deviation of intensity** (noted f_s):

$$f_s(i,j,k,x,y,z,\mathcal{I}) = \sqrt{\frac{1}{|\mathcal{I}_{i,j,k}|}\sum \big(\mathcal{I}_{i,j,k} - f_a(i,j,k,x,y,z,\mathcal{I})\big)^2} \tag{5}$$

with f_a and f_s defined only if $f_b \neq 0$. Depending on the sought applications, many other functions can be used, for example related to the geometry (normal surface, main orientation, ...) or any other features of $\mathcal{I}_{i,j,k}$.

It should be outlined that empty cells can occur from two situations: 1) empty spaces into the scene or 2) missing data because of occlusions. Several approaches are possible to deal with such empty voxels (affecting a value 0, linear interpolation, ...). In this study, and without loss of genericity, we chose to assign them the null value.

Once the voxelization transformation $\mathcal{PV}$ is performed, our data $(i,j,k,\mathbf{I}) \in \boldsymbol{E}_h \times \mathbb{R}$ can be represented through a volume V such that:

$$\begin{aligned} V: \quad \boldsymbol{E}_h &\longrightarrow \mathbb{R} \\ (i,j,k) &\longmapsto \mathbf{I}\,. \end{aligned} \tag{6}$$

3.2 Attribute filtering with the max-tree of voxels

Max-tree Attribute filtering is a popular tool in mathematical morphology. It operates on connected components of an image (if binary) or of its level sets (if greyscale). As a connected filter, it does not shift object edges but proceeds by removing the components that do not fulfill a given criterion (related to the aforementioned attribute). It benefits from efficient implementation through the image representation as a max-tree.

As already stated, the usual definition of the max-tree for 2D images remains valid in case of 3D volumes. Only the connectivity needs to be updated, from 4- and 8-connectivity in 2D to 6-, 18- and 26-connectivity in 3D. The upper level sets of the volume V are obtained from successive thresholdings of the grey levels $l \in \mathbb{R}$ and noted

$$\mathcal{L}_l = \{(i,j,k) \in \boldsymbol{E}_h \mid V(i,j,k) \geq l\}\ . \tag{7}$$

We index by c the connected components within a level set, i.e. $\mathcal{L}_{l,c}$. These components are nested and form a hierarchy called the max-tree. The leaves of the tree correspond to the regional maxima while the root contains the whole volume.

Filtering The max-tree structure provides an efficient way to filter its nodes (i.e. the connected components of the level sets). Such a filtering relies on some predefined criteria called attributes, whose values are usually computed for each node during the tree construction step.

We distinguish here between two kinds of attributes that are scale-dependent and scale-invariant, respectively. In the former category, we can mention the volume and surface (i.e. 3D counterparts of the 2D area and perimeter, respectively), as well as dimensions of the bounding box or the convex hull. Examples of scale-invariant attributes are distributions of grey levels (e.g. standard deviation, entropy), measures computed with moments of inertia (e.g. compactness, sphericity), or moment invariants (e.g. elongation, flatness). The interested reader is referred to [16] for more details.

We provide below a formal definition of the three attributes that have been used in the experiments reported in this paper:

- **Height** (noted A_h):

$$A_h(\mathcal{L}_{l,c}) = \max_{i,j,k}(k) - \min_{i,j,k}(k) \tag{8}$$

- **Volume** (noted A_v):

$$A_v(\mathcal{L}_{l,c}) = \big|(i,j,k)\big| \tag{9}$$

- **Extent** (also named "fill ratio" [9], noted A_e):

$$A_e(\mathcal{L}_{l,c}) = \frac{A_v(\mathcal{L}_{l,c})}{A_v(\mathcal{B}(\mathcal{L}_{l,c}))} \tag{10}$$

with $\mathcal{B}(\cdot)$ the bounding box of a set. Let us note that for the sake of conciseness, the condition $(i,j,k) \in \mathcal{L}_{l,c}$ was systematically omitted in the right part of the previous equations.

The aforementioned attributes are either increasing or non-increasing, depending if their value is increasing from leaves to root or not. The filtering simply consists in assessing each connected component by comparing its attribute value to a given threshold T, and retaining only the filtered set $\mathcal{L}' \subseteq \mathcal{L}$ defined as

$$\mathcal{L}'_l = \{\mathcal{L}_{l,c} \mid A\left(\mathcal{L}_{l,c}\right) \geq T\} \ . \tag{11}$$

While the filtering with an increasing attribute is achieved through pruning the tree (i.e. removing all descendant nodes of $\mathcal{L}_{l,c}$ if $A(\mathcal{L}_{l,c}) < T$), considering a non-increasing attribute leads to pruning and non-pruning strategies [15, 19] that remove full branches or isolated nodes, respectively.

The final step of the filtering is to reconstruct the filtered volume F based on remaining nodes of the max-tree, i.e.:

$$\begin{aligned} F: \quad & \boldsymbol{E}_h \longrightarrow \mathbb{R} \\ & (i,j,k) \longmapsto \max_{l \in \mathbb{R}} \big((i,j,k) \in \mathcal{L}'_l\big) \ . \end{aligned} \tag{12}$$

3.3 Reprojection to the 3D point cloud

After having performed attribute filtering (or any other morphological processing) on the 3D volume, the filtered volume $F(i,j,k)$ needs to be reprojected in the original continuous set of coordinates to produce a dataset $\{x,y,z,\mathcal{F}\} \in \mathbb{R}^3 \times \mathbb{R}$. To this end, we assign to all initial points embedded in each voxel (i,j,k) the value $F(i,j,k)$. Let us note that more complex functions could have been considered here (e.g. interpolation taking into account the position of (x,y,z) in (i,j,k) and intensities of the neighboring voxels). Our choice mathematically reads as applying the inverse transformation function $\mathcal{VP}$ (for "voxels to points") defined as:

$$
\begin{aligned}
\mathcal{VP}: \quad & \boldsymbol{E}_h \times \mathbb{R} \longrightarrow \mathbb{R}^3 \times \mathbb{R} \\
& \{i,j,k,F(i,j,k)\} \longmapsto \{x,y,z,\mathcal{F}\} \text{ with:} \\
& \begin{cases}
(x,y,z) = (x_e,y_e,z_e) \in \boldsymbol{X} \text{ s.t.} \\
\qquad \begin{cases}
x_m + ih \leq x_e < x_m + (i+1)h \\
y_m + jh \leq y_e < y_m + (j+1)h \\
z_m + kh \leq z_e < z_m + (k+1)h
\end{cases} \\
\mathcal{F} = F(i,j,k)
\end{cases}
\end{aligned}
\tag{13}
$$

The proposed scheme allows us to consider the 3D information contained in the 3D point cloud by processing the associated volume. We will illustrate in the next section the relevance of such an approach.

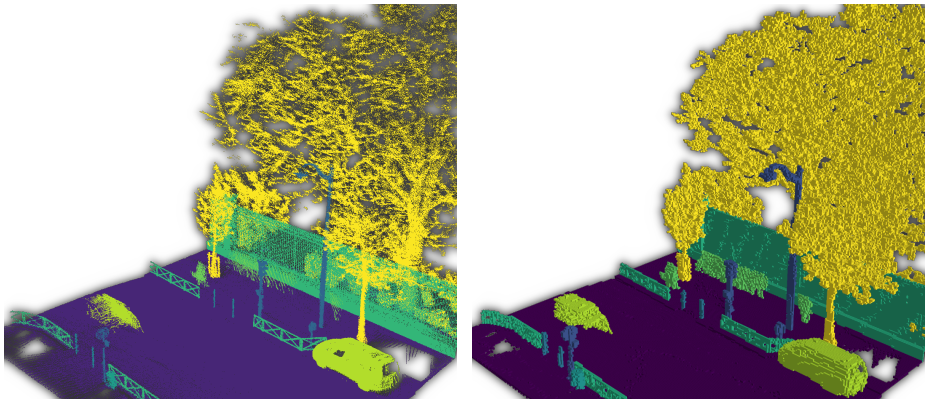
4 Experiments

4.1 Dataset

Experiments have been carried out on the Paris Lille 3D dataset [14]. The LiDAR tiles considered here have been acquired by a mobile laser scanning (MLS) on a street of Paris. As the acquisition source is close to the ground, the point density varies greatly according to the distance of the scanned object (as a consequence, density based max-tree will tend to remove distant objects). However the point cloud density of this dataset is significantly high (between $1,000$ to $2,000$ per square meters), which gives flexibility in the choice of the spatial resolution h in the voxelization process. Additional data is available, with LiDAR intensity return and label associated with each point in the cloud.

4.2 Experimental setup

As a first experiment, we have chosen to filter the labelled point cloud, which is illustrated in Figure 2a. For this point cloud we have fixed the step of the voxel grid to $h = 10$ cm. The labels are given with the dataset and ordered as follows: void (value 0), unclassified (1), ground (2), road sign and traffic light (4), bollard



(a) Point cloud (b) Voxel grid

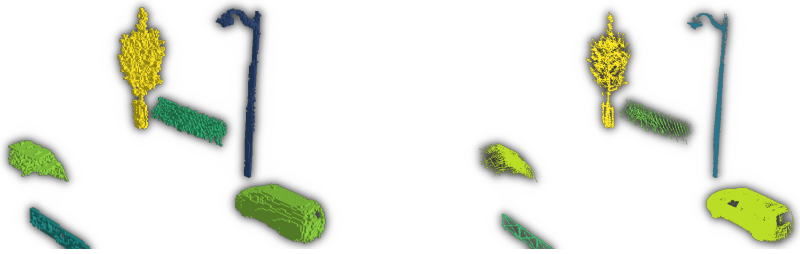
Fig. 2. Visualization of the valued point cloud and the corresponding voxel grid. The 5 classes are represented as follows: road in purple, cars in green, fences in teal, trees in yellow and urban furniture in blue.

(5), trash can (6), barrier (7), pedestrian (8), car (9) and vegetation (10). This order has been used to construct max-tree on the $[0, 10]$ value range. We have represented for each cell of the grid the majority-class of the points (Figure 2b).

We have built the max-tree considering 26-connectivity (i.e. two voxels (i, j, k) and (i', j', k') are neighbors if $\max(|i - i'|, |j - j'|, |k - k'|) = 1$). The tree is augmented with spectral features such as mean grey level and standard deviation and also with spatial features such as the volume, the bounding box and several geometric ratios. During the filtering process, we used the direct non-pruning strategy for the sake of simplicity and to retrieve all the objects corresponding to the required description on non-increasing criteria (extent in our case). Then the filtered max-tree is transformed back into a 3D volume and reprojected into a point cloud (therefore function F in (13) is the label value).

With this experiment we were able to interactively filter objects from the max-tree with geometric object attributes (e.g. volume, height, compactness) in order to choose the appropriate threshold values. The illustration given in Figure 3a is the voxel grid result of a filtering with the volume criterion set as $1,000 < A_v \leq 5,000$. We then transfer the filtered result back into the point cloud (Figure 3b).

An additional example of attribute filtering is given in Figure 4. We can observe the relevance of the height attribute to extract tall objects (e.g. the lamp post is the only object with a height comprised between 10 and 13 meters, Figure 4a). Considering the extent attribute with value between 0.14 and 0.16 allows us to highlight road signs, cars and a few branches, as shown in Figure 4b. Finally, it is possible to combine multiple attribute for a more precise filtering, e.g. objects with height between 1.5 and 3 meters, a volume greater than $1,000$ and an extent between 0.14 and 0.16 match cars in Figure 4c.

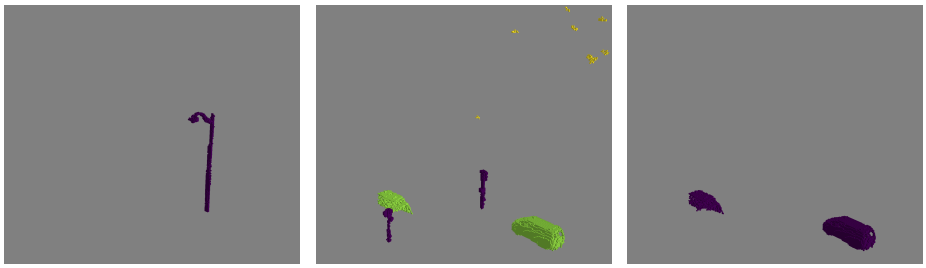


(a) Filtered voxel grid (b) Filtered point cloud

Fig. 3. Visualization of the filtered voxel grid from (Fig. 2b) with an area criterion (a) and the reprojection into the point cloud (b).

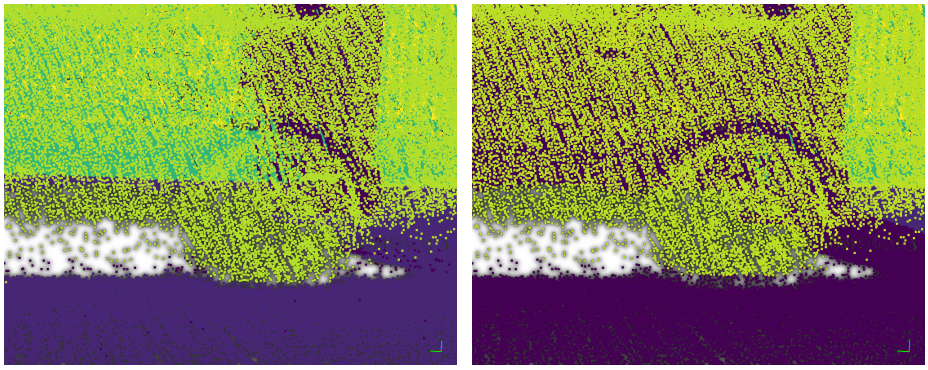
Among the current limitations of the proposed approach, we have noticed that some small blocking artifacts were appearing at the border between two objects in the point cloud (see close-up view in Figure 3). These artifacts are directly linked with the grid discretization method and depend on the voxel size. In our experiments, we observe that artifacts remain small with $h = 10$ cm.

The previous experiments showed that attribute filtering is useful to filter a labeled point cloud. The labels are either defined by visual expert analysis or by automatic classification of the raw LiDAR data. It is also possible to filter directly such raw data. We illustrate some results of preliminary experiments with LiDAR intensity in Figure 6. We can see here the relevance of attribute filters to remove the noise in the point cloud. Indeed, the noise can be easily



(a) Height filtering (b) Extent filtering (c) Multi-attribute filtering

Fig. 4. Different attribute filters on V: connected components characterized by (a) $A_h \in [100, 130]$, (b) $A_e \in [0.14, 0.16]$, (c) $A_h \in [15, 30]$, $A_v > 1,000$, and $A_e \in [0.14, 0.16]$.



(a) Initial point cloud (b) Filtered point cloud

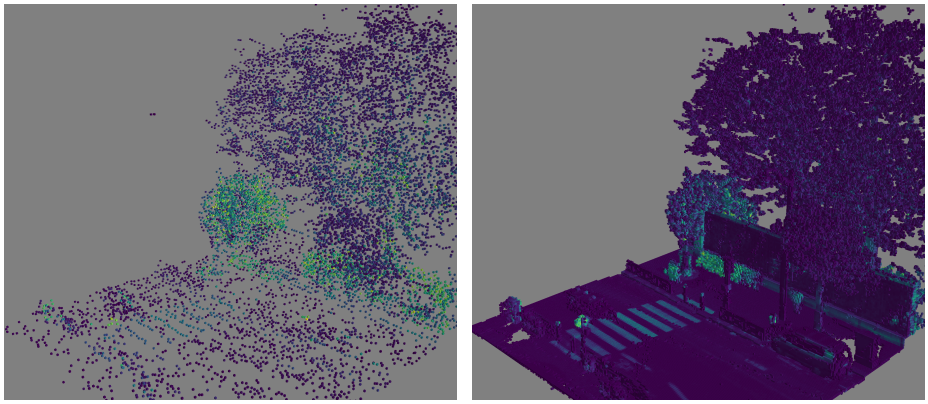
Fig. 5. Close-up of the area where an object is in contact with the ground. In the filtered point cloud (b), blocking artifacts may appear at the boundary between objects (in this image part of the wheel is "blocked" in the road.

characterized by geometric attributes (e.g. small volume nodes correspond to points disconnected from the rest of the scene, see Figure 6a). It is also relevant to remove extreme intensity values (i.e. outliers) with the max-tree.

5 Conclusion

While most approaches for applying mathematical morphology on point clouds are relying on a prior rasterization step into a 2D image, we explore here a different strategy. Indeed, we rather suggest a discretization of the space into voxels to build a 3D volume instead of a 2D image. This choice is motivated by the straightforward extension of morphological operators (including hierarchies) to nD data considering a connectivity of higher dimension. It allows us to benefit from a richer hierarchical representation where each node contains a set of voxels from which advanced features can be computed. We illustrate the relevance of such a framework with the popular attribute filtering, that is applied here on the voxel hierarchy before reconstructing a filtered point cloud. The results obtained on an urban point cloud show the performance of the proposed strategy, that goes far beyond the standard 2D processing usually considered in the literature.

Future work will include extending our proposal to other hierarchical models (e.g. min-tree, tree of shapes) as well as considering some other tree-based tools beyond the standard attribute filtering. Note than an alternative could be to a create nearest neighbour graph representation from the raw set of points and perform filtering using max-tree on this graph. Besides, multiscale description with attribute profiles and data segmentation with hierarchical cuts (to name a few) would be of great interest to deal with point clouds such as those considered in remote sensing. Finally, the blocking artifacts discussed in the paper call for some further studies.



(a) Noise (b) Denoised

Fig. 6. Denoising of raw LiDAR data: outliers and isolated points are removed from the scene using the volume attribute: (a) $A_v < 2$ identifies noise removed from the scene, (b) $A_v \geq 2$ performs the denoising.

References

1. Aijazi, A., Checchin, P., Trassoudaine, L.: Segmentation based classification of 3D urban point clouds: A super-voxel based approach with evaluation. Remote Sensing **5**(4), 1624–1650 (2013)
2. Calderon, S., Boubekeur, T.: Point Morphology. ACM Transactions on Graphics **33**(44) (2014)
3. Dufour, A., Tankyevych, O., Naegel, B., Talbot, H., Ronse, C., Baruthio, J., Dokládal, P., Passat, N.: Filtering and segmentation of 3D angiographic data: Advances based on mathematical morphology. Medical Image Analysis **17**(2), 147–164 (2013)
4. Ferdosi, B.J., Buddelmeijer, H., Trager, S., Wilkinson, M.H.F., Roerdink, J.B.T.M.: Finding and visualizing relevant subspaces for clustering high-dimensional astronomical data using connected morphological operators. In: IEEE Symposium on Visual Analytics Science and Technology. pp. 35–42 (2010)
5. Géraud, T., Carlinet, E., Crozet, S., Najman, L.: A quasi-linear algorithm to compute the tree of shapes of nD images. In: International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing. pp. 98–110. Springer (2013)
6. Gorte, B., Pfeifer, N.: Structuring laser-scanned trees using 3D mathematical morphology. International Archives of Photogrammetry and Remote Sensing **35**(B5), 929–933 (2004)
7. Grossiord, E., Talbot, H., Passat, N., Meignan, M., Terve, P., Najman, L.: Hierarchies and shape-space for PET image segmentation. In: IEEE International Symposium on Biomedical Imaging. pp. 1118–1121 (2015)
8. Guiotte, F., Lefevre, S., Corpetti, T.: Rasterization strategies for airborne LiDAR classification using attribute profiles. In: IEEE/ISPRS Joint Urban Remote Sensing Event (2019)

9. Hernández, J., Marcotegui, B.: Ultimate attribute opening segmentation with shape information. In: International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing. pp. 205–214. Springer (2009)
10. Kiwanuka, F.N., Ouzounis, G.K., Wilkinson, M.H.: Surface-area-based attribute filtering in 3D. In: International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing. pp. 70–81. Springer (2009)
11. Kiwanuka, F.N., Wilkinson, M.H.F.: Radial moment invariants for attribute filtering in 3D. In: Applications of Discrete Geometry and Mathematical Morphology. pp. 68–81. Springer (2012)
12. Padilla, F.J.A., Romaniuk, B., Naegel, B., Servagi-Vernat, S., Morland, D., Papathanassiou, D., Passat, N.: Hierarchical forest attributes for multimodal tumor segmentation on FDG-PET/contrast-enhanced CT. In: IEEE International Symposium on Biomedical Imaging. pp. 163–167 (2018)
13. Peternell, M., Steiner, T.: Minkowski sum boundary surfaces of 3D-objects. Graphical Models **69**(3-4), 180–190 (2007)
14. Roynard, X., Deschaud, J.E., Goulette, F.: Paris-Lille-3D: a large and high-quality ground truth urban point cloud dataset for automatic segmentation and classification. ArXiv e-prints (2017)
15. Salembier, P., Oliveras, A., Garrido, L.: Antiextensive connected operators for image and sequence processing. IEEE Transactions on Image Processing **7**(4), 555–570 (1998)
16. Salembier, P., Wilkinson, M.: Connected operators. IEEE Signal Processing Magazine **26**(6), 136–157 (2009)
17. Serna, A., Marcotegui, B., Hernández, J.: Segmentation of facades from urban 3D point clouds using geometrical and morphological attribute-based operators. ISPRS International Journal of Geo-Information **5**(1) (2016)
18. Serna, A., Marcotegui, B.: Detection, segmentation and classification of 3D urban objects using mathematical morphology and supervised learning. ISPRS Journal of Photogrammetry and Remote Sensing **93**, 243–255 (2014)
19. Urbach, E., Wilkinson, M.: Shape-only granulometries and grey-scale shape filters. In: International Symposium on Mathematical. Morphology. pp. 305–314 (2002)
20. Urien, H., Buvat, I., Rougon, N., Soussan, M., Bloch, I.: Brain lesion detection in 3D PET images using max-trees and a new spatial context criterion. In: Mathematical Morphology and Its Applications to Signal and Image Processing. pp. 455–466. Springer (2017)
21. Westenberg, M.A., Roerdink, J.B.T.M., Wilkinson, M.H.F.: Volumetric attribute filtering and interactive visualization using the max-tree representation. IEEE Transactions on Image Processing **16**(12), 2943–2952 (2007)
22. Wilkinson, M.H.F., Westenberg, M.A.: Shape preserving filament enhancement filtering. In: Medical Image Computing and Computer-Assisted Intervention. pp. 770–777. Springer (2001)