Automatic classification of urban ground elements from Mobile Laser Scanning Data

J. Balado\textsuperscript{a}, L. Díaz-Vilariño\textsuperscript{a,b}, P. Arias\textsuperscript{a}, H. González-Jorge\textsuperscript{a}

\textsuperscript{a}Applied Geotechnologies Group, Dept. Natural Resources and Environmental Engineering, University of Vigo, Campus Lagoas-Marquesende, CP 36310 Vigo, Spain
\textsuperscript{b}GIS Technology, OTB Research Institute for the Built Environment, Delft University of Technology, Julianalaan 134, 2628 BL Delft, The Netherlands

Abstract

Accessibility diagnosis of as-built urban environments is essential for path planning, especially in case of people with reduced mobility and it requires an in-depth knowledge of ground elements. In this paper, we present a new approach for automatically detect and classify urban ground elements from 3D point clouds. The methodology enables a high level of detail classification from the combination of geometric and topological information. The method starts by a planar segmentation followed by a refinement based on split and merge operations. Next, a feature analysis and a geometric decision tree are followed to classify regions in preliminary classes. Finally, adjacency is studied to verify and correct the preliminary classification based on a comparison with a topological graph library. The methodology is tested in four real complex case studies acquired with a Mobile Laser Scanner Device. In total, five classes are considered (roads, sidewalks, treads, risers and curbs). Results show a success rate of 97\% in point classification, enough to analyse extensive urban areas from an accessibility point of view. The combination of topology and geometry improves a 10\% to 20\% the success rate obtained with only the use of geometry.

Keywords: urban environment, as-built 3D, graph library, accessibility, smart cities, point cloud, topology, adjacency;

1. Introduction

The United Nations estimates that over 70\% of the world’s population will be living in towns and cities by 2050 [1]. A population increase implies a greater number of urban trips, and pedestrian displacements that make necessary a smart city use [2]. The smart city is a recent concept that integrates multiple information related to the city such as energy, construction, transport, services and resources. The first step to create a smart city is modelling the as-built environment [3]. 3D models are becoming essential to represent cities and the basis for storing and using the information of the as-built environment.

Laser scanning is a consolidated technology for the collection and analysis of three-dimensional data on the as-built status of large-scale civil infrastructures. Urban acquisitions with a mobile laser scanning (MLS) are faster than with terrestrial laser scanner (TLS) and resulting point clouds are obtained with good quality, much higher than with aerial laser scanning (ALS). Laser scanner also can be equipped in UAV providing more detailed information than common ALS [4] and giving a different point of view than MLS, but with less autonomy than both. However, point clouds are composed of massive and raw information that should be processed to extract the information that is useful for the applications they are intended to serve.

Although, intense efforts have been made to facilitate the automatic processing of 3D laser scanning data, the productive modelling of the complete as-built environment is still an unsolved issue. With regard to city modelling, most works focus in façade modelling from Terrestrial Laser Scanner [5,6] or roof reconstruction from Airborne Laser Scanner ([7–9]). Other works are focused on modelling the city ground because it is the element that communicates all components in the city: metro station, bus stop, parking with buildings or gardens. Many times, it is supposed that every building is

\textsuperscript{*} Corresponding author.

\textit{E-mail address:} jbalado@uvigo.es (J. Balado), lucia@uvigo.es L.diaz-Vilarino@tudelft.nl (L. Díaz-Vilariño), parias@uvigo.es (P. Arias), higiniog@uvigo.es (H. González-Jorge).
connected to each other without paying attention to the way itself. However, modelling ground urban elements is essential to understand a city and more specifically, it is of great interest for those people that present mobility impairments.

City accessibility is treated indirectly in works aiming to detect curbs for autonomous vehicles [10–14]. Curbs, and others small steps, are also important from an accessibility point of view. For instance, wheelchairs cannot navigate through curbs, acting as an architectural barrier. In this context, Serna and Marcotegui [15] contextualise their work on curbstones detection as a barrier identification from an accessibility point of view.

Actually, architectural barriers are an important problem in urban environments. Many places can only be accessible by stairs or steps, and mobility-impaired and elderly people have problems to access them. In this context, different initiatives are being developed to mark obstacles in urban zones [16]. In terms of regulations, the European Union aims to guarantee disabled people rights [17]. Furthermore, International organizations promote standards, such as the ISO-25142, to eliminate architectural barriers and preserve measures in stairs, hallways and ramps and in their proximity.

The aim of this work is to develop an automatic methodology to detect and classify urban ground elements such as stairs, curbs, sidewalks and ramps from point clouds acquired with Mobile Laser Scanner. The methodology is based on segmentation followed by feature analysis and geometric classification in preliminary classes that are verified according to their topology from comparison with a topological graph dictionary. The methodology is tested in four real case studies.

This paper is organized as follows. Section 2 collects related work about urban element classification, topology and graphs. Section 3 deals with the methodology while Section 4 presents the results. Finally, Section 5 is devoted to conclude this work.

2. Related Work

This section deals with the review of the recent literature on urban element classification, followed by a discussion about the use of topology and graphs to identify and reconstruct models.

2.1. Urban element classification

Urban reconstruction is mainly composed of three consecutive processes: segmentation, classification and reconstruction [19–22].

Segmentation is the process of dividing a point cloud or range image into a number of disjoint subsets [23,24]. In a classification, these subsets are sorted into classes. And in the reconstruction, the segmented and classified subsets are used to create a new representation of the scene. Segmentation and classification usually appear together and different approaches have been presented in the recent years due to the increasing availability of 3D point cloud data and respective acquisition systems [25].

Methodologies based on rasterizing point clouds are conventionally used to extract façades of urban environments. Hernández and Marcotegui [26] and Serna et al. [27] project points in 2D creating intensity images based on their height. Pixels with high intensity values correspond with vertical elements such as façades and trees. Morphological operations can be applied to improve the images and the position of the elements is finally extracted.

Aijazi et al. [28] present a method based on segmenting a point cloud in voxels. Then, the voxels are joined into super-voxels to form objects according to their attributes such as the geometrical centre, the mean of the RGB value of the constituting 3D points, etc. Finally, they are classified into buildings, roads, poles, cars and trees based on their local descriptors (surface normals, geometrical shape, barycentre, etc.). Another octree-based voxelized methodology to segment urban elements is introduced by Vo [29]. In this case, point clouds are converted to voxels and segmented according to their orientation.

Currently, machine learning is a very popular instrument to classify elements. Weinmann et al. [25] compare different number of neighbourhoods, features, values and classifiers searching an optimal façades, ground and vegetation detection. This methodology gets good results in extensive point clouds, but it also needs big samples to train the machine learning and it is not exempt of some false positives.

More specifically related to urban floor classification, most of the existing literature is focused on curbs. Using vehicle trajectory, Wang et al. [13] detect salient points near to MLS trajectory and consider them as curbs. Others authors [10,12] design methods to work in real-time. They implement more sensors than a LIDAR, such as high-resolution cameras to use density, geometry information towards the detected curbs or other lateral obstacles such as barriers and walls.

Related to raster methods, Liu et al. [11] transform the point cloud in a local Digital Elevation Map (DEM) and search elevation gradient variation that correspond with curbs. Then, they extract curbs pixel to a new image where they apply
Hough transform and RANSAC to extract lines of the curbs. In a similar study, Serna and Marcotegui [15] also use a raster image, but they interpolate it to create a new image without occlusions to enable the detection of curbs. They can be partially occluded by parked cars so it is necessary to reconnect them. This approach is very important because they study accessibility of an extensive area, with independence of occlusions during the acquisition.

Other ground elements very important for accessibility diagnosis and navigation planning are stairs and ramps. Most literature involving addressing stairs and ramps detection is related to indoor environments. Oßwald et al. [30] and Luo et al. [31] detect vertical and horizontal planes that forming stairs and extract their geometric parameters. Both papers use humanoid robots climbing and scanning stairs. This method is applied to point clouds in which stairs are isolated from the rest of the environment. Sanchez and Zakhor [32] detect stairs using RANSAC from inclined planes segmented with PCA. After, they extract six parameters (number of steps, reference point, tread depth, riser height, step width and azimuth) to create a model from the stair.

Schnabel et al. [33] search in point clouds graphs with geometry and topology information from their previous designed models to recognize basic elements like stairways, dormers and columns. This study illustrates the potential of the graphs and their negative part: isomorphism problem. This is the most restrictive limitation in comparing graphs, which consists in determining which graphs are equivalent or if a graph A can be included into a larger graph B. It is a problem called hard-NP [34] and their unique solution is an exhaustive search, with their cost in computing time and resources.

2.2. Topology and graphs

Topology studies proximity and consistence relations between different elements [35,36]. These relations can be represented through graphs. Although topology relations and graphs have also been used in few applications, like Schnabel et al. [33], who employ graphs to classify different elements independently of indoor or outdoor environments, for example, columns are connected at both ends, and dormers on roofs or stairs have specific query graphs. Most of applications deal with roof reconstruction, modelling and network analysis, but not classifying [37,38].

Sampath and Shan [39] extract roof topology from isolated roofs applying Voronoi diagram and define Voronoi neighbourhood to set adjacency relation between points. With these relations, they cluster the points to segment the cloud into roof planes. Elberink and Vosselman [7] add a 2D building map to cluster roof buildings and extract relations of roof planes. Bizjak [40] uses K-neighbourhood with its normal to detect roofs and bounding boxes of roof sides to calculate adjacencies.

Taillandier [41] combines cadastral maps and aerial images to create 3D graphs with normal vectors of roof shapes and he uses them to model building. These basic graphs, called roof-topology graphs will be used and extended for many authors: Elberink and Vosselman [7], Perera et al. [8], Xiong et al. [42] and Verma et al. [9] among others. Out of roof applications, Dörschlag et al. [43] complete adjacency graph with geometric information (parallel and right-angles) between parts of a polygon building to coarse level detailed models.

Some authors use graphs after classification or reconstruction phase. They extract topologic information (adjacency and inclusion) to different applications such as energy or navigation purposes. Cao [44] extracts room topology in indoor environments, previously reconstructed from a grammar-based methodology [45], to create a graph which relates rooms as nodes with the doors and the walls between them as edges. Zhou and Neumann [46] simplify geometric reconstructed models using topology information. Topologic models are proved to be more coherent than geometric models. The graphs also can be used to store different data types. In Becker [47] a six level hierarchical graph stores urban information, since district organization of the city to any window for each building façade, but this approach is not dealing with ground elements or complex floors.

It is possible to reconstruct buildings with a group of simple models that they were previously defined in a library. In the approach of Lafarge et al. [48], final models are constructed comparing roof forms with their library 3D forms and selecting the most probable. Xiong et al. [42] join roof-topology graphs with a library of graphs to generate their own reconstruction methodology using graph isomorphism. Also, they extend the library use to correct remaining error models of their methodology, they train the library-dictionary with the graphs errors and associate them for each solution, similar to a spelling corrector.

With regard to previous approaches, we design a methodology to detect and classify ground elements in urban scenes from Mobile Laser Scanning data. The approach enables a detailed classification from the combination of a geometric preliminary classification followed by a topologic verification and correction from a graph library, which makes it more robust and less dependent on data quality.
3. Methodology

The initial step of the methodology consists on a planar segmentation and refinement to divide point cloud into planar regions (Section 3.1). Next, planar regions are classified, firstly according to their geometric features (Section 3.2) and finally to their adjacency by comparing their topological graphs with a graph library (Section 3.3).

3.1. Segmentation

This section includes a planar segmentation followed by a refinement process (Fig. 1). Most elements in urban ground are planar surfaces. However, the quality of the input data affects to the segmentation results causing under and over-segmentation. Split and merge operations are applied to handle with global errors. A method is implemented to refine coplanar and continuous elements of different sizes such as risers and walls. And finally, a refinement operation is developed to segment road from sidewalks because imperfections on road surface affects to the previous results.

![Fig. 1. Segmentation workflow](image)

3.1.1. Planar segmentation

A planar segmentation similar to Bueno et al. [49] and based on the comparison of normal point and distance point-to-plane is implemented. Normal vectors \( \mathbf{N} = (\mathbf{N}_x, \mathbf{N}_y, \mathbf{N}_z) \) associated to points \( \mathbf{P} = (x, y, z) \) are calculated according to the Principal Component Analysis (PCA), taking into account a \( k \) neighbourhood points obtained by applying a K Nearest Neighbours (KNN) procedure [50]. The point with the lowest normal variance, that is the point with the minimum residual of the planar fitting, is selected as the planar seed \( s \). Following, the algorithm includes in the planar region \( \mathbf{R} = \{ \mathbf{R}_1, \mathbf{R}_2 \ldots \mathbf{R}_n \} \) all points satisfying two geometric conditions: normal fitting \( \mathbf{nt} \) and distance between candidate points and normal seed point \( d \). If the number of points selected exceed a minimum of points \( np \), they are saved into a new region \( \mathbf{R}_1 = \{ \mathbf{P}: \mathbf{N}_i \sim \mathbf{N}_1 \land |\mathbf{R}_x \perp \mathbf{P}_i| < d \} \).

After this process, point cloud is organised in regions \( \mathbf{P} \sim \mathbf{R}_1 \cup \mathbf{R}_2 \cup \ldots \mathbf{R}_n \). There is no unambiguous correspondence between planar regions and elements for characterization. Planar segmentation can result in zones with under-segmentation or over-segmentation [51,52], which can be caused by imperfections, deformations, noise, occlusions and ornaments on surfaces; even by the own urban scene geometry. To improve the quality of segmentation, a sequence of refinement processes is implemented after planar segmentation.

3.1.2. Split refinement

Split (Algorithm 1) consists in analysing point continuity inside each planar region. They are organized into a kd-tree and relations are searched between points less than \( rds \) distance. Points (nodes) and relations (edges) are represented into
a graph and applied connected graph component. After, different disconnected sub-graphs $dsg$ are classified, if they have more than $np$ number of points, into new planar regions $Rs$. Split operation solves under-segmentation problem. Fig. 2 shows the results of split in a sidewalk with benches.

**Algorithm 1: Split**

**Inputs:** Regions Points $\{R\}$, radio_search $rds$, number_points $np$

**Outputs:** Splitted Regions Points $\{Rs\}$

**For** $i = 1$ to length$\{R\}$ **do**

Ordered points in kdtree $\{kd-points\} \leftarrow$ kdtree $\{R_i\}$
Point relations $\{rel\} \leftarrow$ search kdtree ($\{kd-points\}, rds$)
Graph $\{G\} \leftarrow \{kd-points\} & \{rel\}$
Disconnected sub-graphs $\{dsg\} \leftarrow$ connected graph component $\{G\}$

**For** $j = 1$ to length$\{dsg\}$ **do**

If $\{dsg_j\}$ points $> np$

$\{Rs_{save}\} \leftarrow$ Add $\{dsg_j\}$ points
save++

**End_if**

**End_for**

**End_for**

Return $\{Rs\}$

---

Fig. 2. Benches and baskets before split operation (up), they are in the same planar region (same colour: benches in violet and baskets in yellow). After split operation (down), they are in different regions (different colours).

3.1.3. **Merge refinement**

Merge (Algorithm 2) consists in combining adjacent planar regions $Rs$ with planar region normals $Nr$ below a certain normal threshold $nth$ into a unique region $Rm_i = \{Rs_i, Rs_j; Nr_i \sim Nr_j \land Adx(R_i, R_j)\}$. Merge operation solves over-segmentation problem, caused by noise and deformations on element surfaces. Fig. 3 shows the results of merging a sidewalk.

**Algorithm 2: Merge**

**Inputs:** Splitted Regions Points $\{Rs\}$, Adjacent Regions $\{Adx\}$, Regions Normals $\{Nr\}$, normal_threshold $nth$

**Outputs:** Merged Regions Points $\{Rm\}$

**For** $i = 1$ to length$\{Rs\}$ **do**

If $\{Adx(\{N\} \text{ into } nth)\} \in \{Adx_i\}$

Region_Index $\{Idx\} \leftarrow i$

Region_Merged_Relation $\{rel\} \leftarrow \{Adx_i\}$

**End_if**

**End_for**

Graph $\{G\} \leftarrow \{Idx\} & \{rel\}$
Disconnected sub-graphs $\{dsg\} \leftarrow$ connected analysis $\{G\}$

**For** $j = 1$ to length$\{dsg\}$ **do**
{Rm\_save} \leftarrow \text{Add \{dsg\}_j\} \text{ points}
\text{save++}
\text{End}_\text{for}
\text{Return \{Rm\}}

Fig. 3. Floor before merge operation (up), it is segmented in three regions (colours green, blue and yellow) and floor after merge operation (down), in a unique region (colour violet).

Adjacency information between planar regions is necessary to carry out the merge operation. Adjacency information is extracted from point cloud as follows (Algorithm 3). For each point in the cloud $C = Rs_1 \cup Rs_2 \cup \ldots Rs_n$. KNN is used to construct a mesh relating all points. With these relations, we know which points are adjacent to each other $\text{Adx}(p_i, p_j)$. If a pair of adjacent points belongs to different planar regions, these regions are considered as adjacent regions $\text{Adx}(R_i, R_j)$.

**Algorithm 3**: Adjacent regions

- **Inputs**: Regions\_Points \{R\}, number\_nearest\_points \(k\)
- **Outputs**: Adjacent\_Regions \{Adx\}

```
Global cloud \{C\} \leftarrow \text{Add \{R\}}
Ordered points in kdtree \{kd\text{-}points\} \leftarrow \text{kdtree \{C\}}
Point relations \{rel\} \leftarrow \text{knn search (\{kd\text{-}points\},}\(k\))
\text{For } j = 1 \text{ to length\{rel\} do}
\quad \text{If } \text{kd-point}_j \neq \text{rel}_j
\quad \quad \quad \text{\{Adx}_j\} \leftarrow \text{Add rel}_j
\quad \text{End_if}
\text{End_for}
\text{Return \{Adx\}}
```

3.1.4. Coplanar refinement

In urban environments, it is common the existence of two coplanar and continuous ground elements such as a riser of a set of stairs and a wall (Fig. 4). After applying the previous phases, it results in an under-segmentation case. To solve this, a specific operation (Algorithm 4) is implemented. It is based on rasterizing the high vertical regions (i.e. walls), with an image resolution of 0.1 x 0.1 meters/pixel, and dividing in zones according to a height threshold $h_s$ (25cm, explained in Section 3.2.2) $\text{Rm}_i = \{\text{Rm}_i; I_{xy} \geq h_s\} \cup \{\text{Rm}_i; I_{xy} < h_s\}$. If the new zone with less height jump than $h_s$, candidate to be a step, has more points than $np$, it is separated and saved in a new region $Rs_i = \{\text{Rm}_i; I_{xy} < h_s\}$.
Fig. 4. Third riser (in red) is coplanar with region wall (grey). Colour code of the rest of elements: sidewalk in olive, risers in green and treads in blue.

**Algorithm 4:** Coplanar refinement

Inputs: Vertical_Regions_Points \{R_m\}, number_poits \(np\), height_step \(hs\), pixel_size \(ps\)

Outputs: Segmented_Regions_Points \{Rs\}

\[ \{Rs\} \leftarrow \{R_m\} \]
\[ \text{save} \leftarrow \text{length}\{R_m\} \]

For \(i = 1\) to length\{R\} do

\[ \text{image heights} \{I\} \leftarrow \text{raster} (\{R_m\} \text{ in } Z \text{ coordinate}, \text{ps}) \]
\[ \text{points} \{p\} \leftarrow \{I_{x,y}\} < \text{hs} \]

If length\{p\} > \(np\)

\[ \text{save}++ \]
\[ \{Rs_{\text{save}}\} \leftarrow \text{Add} \{\text{points}\} \]
\[ \{R_i\} \leftarrow \text{Remove} \{\text{points}\} \]

End_if

End_for

Return \{Rs\}

3.1.5. Road-Sidewalk refinement

Due to the high size of roads and sidewalks and to the presence of small deformations on their surface, these elements are usually not correctly segmented [53]. Roads tend to be over-segmented and sometimes they are hardly distinguishable from sidewalks.

In this approach (Algorithm 5), MLS trajectory \(\mathbf{T} = (\mathbf{T}_x, \mathbf{T}_y, \mathbf{T}_z)\) is used to locate the regions that include some road (Fig. 5.a), the trajectory points are over a region (road) where the MLS circulated. Then, normal variance is calculated with KNN search with \(k\) neighbours bigger than it was used in planar segmentation with the objective of detect small deformations on the floor. Next, points with variance \(v_{th} = 0.0005\) (value obtained experimentally and considered as characteristic of irregular zones: curbs, deformations, union zones between sloped elements) are removed from the region \(R_{\text{road}} \cap \text{sidewalk} = \{R_{\text{road}} \cap \text{sidewalk}: \mathbf{T}_{xy} \in R_{\text{road}} \cap \text{sidewalk}\} \) to break the continuity in the region (Fig. 5.b). As a result, a discontinuity is obtained and after a split operation (Section 3.1.2.), the original region is segmented into two different regions \(R_{\text{road}} \cap \text{sidewalk} = R_{\text{sidewalk}} \cup R_{\text{road}}\) (Fig. 5.c). Previous removed points are restored to the nearest region and the region under the MLS trajectory is labelled as road \(R_{\text{road}} = \{R_{\text{road}} \cap \text{sidewalk}: \mathbf{T}_{xy} \in R_{\text{road}} \cap \text{sidewalk}\}\) (Fig. 5.d).

**Algorithm 5:** Road-Sidewalk refinement

Inputs: Ground_Region_Points \{R_{\text{road}} \cap \text{sidewalk}\}, number_poits \(np\), normal_variance_treshhold \(v_{th}\), radio_search \(rds\), k_nearest_neighbours \(k\), mls_trajectory \(\mathbf{T}\)

Outputs: Road_Regions_Points \{R_{\text{road}}\}, Sidewalk_Regions_Points \{R_{\text{sidewalk}}\}

Ordered points in kdtree \{kd-points\} \leftarrow kdtree \{R_{\text{road}} \cap \text{sidewalk}\}
\[ \text{Idx_neighbours} \{\text{IdxKNN}\} \leftarrow \text{knn_search} (\{kd-points\}, k) \]
\[ \text{Normal_Variance} \{\text{VN}\} \leftarrow \text{PCA} (\{R_{\text{road}} \cap \text{sidewalk}\}; \{\text{IdxKNN}\}) \]
\[ \text{Region} \{\text{R}\} \leftarrow \Phi; \]
\[ \text{how variance cloud} \{\text{LV}\} \leftarrow \{R_{\text{road}} \cap \text{sidewalk}: (\{\text{VN}\} < v_{th})\}; \]
\[ \text{high variance cloud} \{\text{HV}\} \leftarrow \{R_{\text{road}} \cap \text{sidewalk}: (\{\text{VN}\} > v_{th})\}; \]
new regions \{R\} \leftarrow \text{SPLIT (\{LV\}, rds, np)}
nearest point \{IdxNP\} \leftarrow \text{knn_search(\{R\}, \{HV\})}
\{R\} \leftarrow \text{Add (HV (IdxNP))}
\{R_{\text{road}}\} \leftarrow \{R\} \text{ contains } T;
\{R\} \leftarrow \text{Remove (R_{\text{road}})}
\{R_{\text{sidewalk}}\} \leftarrow \text{MERGE (\{R\})}
\text{Return } \{R_{\text{road}}, R_{\text{sidewalk}}\}

Fig. 5. Ground plot of Road-Sidewalk segmentation: a) input ground point cloud, b) ground cloud without elevate variance normal points, c) cloud segmented into discontinuous regions and classified (sidewalk in blue and road in purple), d) elevate variance normal points returned to segmented cloud.

3.2. Preliminary geometric classification

Four classes are considered for classifying point cloud regions according to their geometric features: risers and curbs, treads, sidewalks and High Vertical Elements (HVE). HVE class includes walls, façades and trees. Although they are not part of the ground, they contain adjacency information for a posterior topologic analysis. Roads are not considered for classification in this section because they are previously identified in Section 3.1.5. by using the MLS trajectory, as some sidewalks.

In this work, a decision tree based on the ISO-25142 and experimental values is implemented to label regions in the preliminary classes (Fig. 6). In addition to international standards (ISO-25142), each country has specific regulations aiming to conduct the construction of urban scenes in an accessible way. Many urban scenes were constructed previously to these regulations. For this reason, although it is not possible to establish an unequivocal classification based on geometrical parameters, ISO-25142 is set out as a good reference.

![Fig. 6. Decision tree for preliminary geometric classification.](image)

For the preliminary geometric classification, three geometric features are analysed to construct the decision tree: tilt, height and width.

The tilt is calculated as the angle between normal component \( \mathbf{N} \) and horizontal plane (Eq. 1) and it is used to classify regions in horizontal and vertical elements.
\[
    \text{tilt} = \text{abs}(\text{atang} \sqrt{\frac{N_x^2 + N_y^2}{N_z}})
\]

The height is calculated only in vertical regions. When an element (main curbs) is inclined along a street, a direct measure DM of the height will report the difference between the highest Z value in the highest zone of street (right in Fig. 7.a) and the lowest Z value in the lowest zone of the street (left in Fig. 7.a). In a similar way, if an element presents deformations and waves (Fig. 7.b), a DM reports a not desirable measure. To avoid this, regions are segmented in regular stretches and the height is calculated as the mean of all segments sm to minimize the error and correct false measures in elements with inclination or little waved distortions. Vertical elements are divided by their height in HVE or curbs and risers, since 25cm is the maximum height admitted for curbs and risers.

![Fig. 7. Front view. Comparative between height direct measures (DM) and segmented measures (sm) in inclined curb (a) and in waved curb (b).](image)

The horizontal elements are classified by their width. Although the ISO-25142 does not establish a tread maximum width, it recommends a passage width of 1.2m to allow simultaneous pass of two persons. A direct measure can return erroneous information, for example, in the treads with U, C or L form. So the regions are converted to a raster (pixel size of 0.1m) and a morphological erosion is applied (Fig. 8). If raster image is consumed by erosion, it means that the region is narrower than a specific threshold. With this method a boolean parameter was obtained.

![Fig. 8. Narrow analysis on top view. a) Sidewalk. b) Tread in L with same large (x and y) that the sidewalk. c) Sidewalk eroded with maximum tread width: element conserved. d) Tread eroded with maximum tread width: element disappeared.](image)

3.3. Final topologic classification

3.3.1. Adjacency analysis

Adjacency analysis of urban scenes provides the topological relationships between contiguous elements. These relationships can be represented in graphs (Section 3.3.2) and patterns can be observed in the same class of elements. For example, a stair can be represented by a sequence of risers and treads connecting two floors and limited by two walls. This definition can be associated to a graph such as in Fig. 9.
Fig. 9. Stair model and its associated graph. Colour code: sidewalk in olive, risers in green, treads in blue and wall in light grey.

However, extensive and complex graphs involve isomorphism problems (Section 2.1). To avoid isomorphism problem, graphs are reduced to small and meaningful units, easy to compare. For instance, the stair can be defined by the existence of a set of treads and risers and consequently, by the graphs defining these units (Fig. 10).

Fig. 10. Associated elementary graphs to tread and riser extracted of stair associated graph

3.3.2. Graph library

All elementary graphs are stored in a graph library. In this way, the library can be extended and more graphs are considered. This implies that three elements are classified using six elementary graphs: treads, risers and curbs (Fig. 11).

Fig. 11. Graph library: a) tread elementary graphs, b) riser elementary graphs, c) curb elementary graph.

3.3.3. Topologic classification.

Topologic classification is carried out as follows. For each element previously classified, its associated elementary graph is searched in the graph library, and relations and nodes are compared. If all relations and nodes are confirmed, the identity of the element is verified. When there are more than one elementary graph associated with an element, each graph is compared. The comparison of graphs is essential for identifying elements with the same geometric features such as risers or curbs. Although topologic classification reduces the number of false positives produced by geometric classification, there are still some elements wrongly classified, especially elements without ground connection. Topologic classification is combined with a connected component analysis of all classified elements. The elements separated from the ground (not connected or accessible to it) are removed as truth elements.

4. Experiments

4.1. Datasets
The methodology was tested in four real case studies, selected according to their representativeness and the presence of different classes of ground elements. They were all acquired with MLS LYNX Mobile Mapper of Optech [54] and although MLS is a consolidated technology for acquiring urban point clouds, the trajectory of MLS systems is usually restricted to roads since they are typically mounted on vans or cars. This fact affects to the quality of the ground data, parked vehicles typically occlude pedestrian areas such as sidewalks. Moreover, elevated elements such as stairs usually present a lower data density [55].

4.1.1. Dataset 1: Ávila - Humilladero

The first dataset (Fig. 12.a) corresponds to a zone around Humilladero street in the city of Ávila, Spain. It contains stairs connecting it to the Portugal Avenue, two roads and many curbs. The size of the point cloud is 1.6 million of points. Many cars and people were present during acquisition. Both roads are built of cobblestone, avenue sidewalks of regular stones and street sidewalks of irregular stones, because it corresponds to the historical city centre.

4.1.2. Dataset 2: Ávila - Feria

The second dataset (Fig. 12.b) corresponds to La Feria square in Ávila, Spain. It contains a complete square surrounded by three roads with their associated curbs and sidewalks and as long as several building façades. The square has three height levels connected by stairs, also connected with sidewalks. Close to the stairs, there is a zone with five trees. In total, point cloud has 1.5 million of points. This dataset also is acquired in the historical city centre. Roads are built of cobblestones and sidewalks and square floor of regular stone blocks.

4.1.3. Dataset 3: Málaga - Negros

The third dataset (Fig. 12.c) corresponds to Los Negros street in Málaga, Spain. It contains stairs, a sidewalk on the right side of an inclined garden with irregular surfaces, bush and rocks. There is an isolated house close to the stairs and a set of building facades. The size of the point cloud is 850 thousand of points. This dataset is not acquired in a historical city centre. Asphalt is the superficial material of roads, and concrete for sidewalks and stairs.

4.1.4. Dataset 4: Lugo - María

The fourth dataset (Fig. 12.d) corresponds to the union of Santa María square with Bom Xesús and Bispo Basunto streets in Lugo, Spain. It is a pedestrian area in a historical city centre, so there are not road and curb elements. There are stairs to access to a building and to an elevated street. All ground elements are built from stone blocks. The point cloud has 1.4 million points.

Fig. 12. Datasets: a) Ávila – Humilladero, b) Ávila – Feria, c) Málaga – Negros, d) Lugo – María.
4.2. Parameters

The segmentation is based on different thresholds and parameters. The high level of detail needed for the classification involves that the planar segmentation (Section 3.1.1) is improve through four phases of refinement (Sections 3.1.2 to 3.1.4). The value of the parameters in the planar segmentation must guarantee a compromise solution, minimum over-segmentation and under-segmentation, for the posterior refinement processes, especially in the zones with small elements, as stairs. Due to the importance of the first phase, the impact of the thresholds in the segmentation are analysed in a representative zone, the stairs of dataset 1 with two index as employed by [56] (Tables 1 to 3). The over-segmentation index quantifies the number of different regions after the segmentation process, ideally it must be 28 regions. The under-segmentation index quantifies the number of treads and risers of different steps merged in one region, ideally must be 0.

Table 1. Relation between index and \(k\) neighbours (\(nth\) and \(d\) fixed in 0.5 rad and 0.2 m, respectively)

<table>
<thead>
<tr>
<th>(k)</th>
<th>(k = 10)</th>
<th>(k = 25)</th>
<th>(k = 50)</th>
<th>(k = 75)</th>
<th>(k = 100)</th>
</tr>
</thead>
<tbody>
<tr>
<td>under-segmentation index</td>
<td>11</td>
<td>14</td>
<td>11</td>
<td>18</td>
<td>20</td>
</tr>
<tr>
<td>over-segmentation index</td>
<td>193</td>
<td>132</td>
<td>94</td>
<td>77</td>
<td>49</td>
</tr>
</tbody>
</table>

Table 2. Relation between index and \(nth\) (\(k\) and \(d\) fixed in 50 neighbours and 0.2 m, respectively)

<table>
<thead>
<tr>
<th>(nth)</th>
<th>(nth = 0.1) rad</th>
<th>(nth = 0.3) rad</th>
<th>(nth = 0.5) rad</th>
<th>(nth = 0.7) rad</th>
<th>(nth = 0.9) rad</th>
</tr>
</thead>
<tbody>
<tr>
<td>under-segmentation index</td>
<td>7</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>14</td>
</tr>
<tr>
<td>over-segmentation index</td>
<td>156</td>
<td>146</td>
<td>94</td>
<td>90</td>
<td>78</td>
</tr>
</tbody>
</table>

Table 3. Relation between index and \(d\) (\(k\) and \(nth\) fixed in 50 neighbours and 0.5 rad, respectively)

<table>
<thead>
<tr>
<th>(d)</th>
<th>(d = 0.05) m</th>
<th>(d = 0.1) m</th>
<th>(d = 0.2) m</th>
<th>(d = 0.3) m</th>
<th>(d = 0.4) m</th>
</tr>
</thead>
<tbody>
<tr>
<td>under-segmentation index</td>
<td>1</td>
<td>5</td>
<td>11</td>
<td>17</td>
<td>20</td>
</tr>
<tr>
<td>over-segmentation index</td>
<td>258</td>
<td>167</td>
<td>94</td>
<td>86</td>
<td>58</td>
</tr>
</tbody>
</table>

With the increase of the threshold values, the under-segmentation index is increased and the over-segmentation index is decreased. With \(k = 10\) or 25 neighbours, \(nth = 0.1\) rad or 0.3 rad and \(d = 0.05\) m or 0.1 m there is over-segmentation and an increase of noise, especially in risers and in the border transitions between risers and treads that are wrongly segmented in independent regions. Fig 13.a shows results of the segmentation with \(k = 100\) neighbours, \(nth = 0.2\) rad and \(d = 0.2\) m. With \(k = 75\) or 100 neighbours, \(nth = 0.9\) rad and \(d = 0.3\) m and 0.4 m, the under-segmentation is increased, the process is less sensible to noise and local deformations but there are over-segmentation in zones without under-segmentation, different risers and treads are joined each other. Fig 13.b shows results of the segmentation with \(k = 50\) neighbours, \(nth = 0.2\) rad and \(d = 0.05\) m. The values selected is \(k = 50\) neighbours, with \(nth = 0.2\) rad and \(d = 0.2\) m, that offers a balanced solution between over and under-segmentation, stairs are segmented in regular spaces. \(nth = 0.7\) rad can be another possible option.

Fig. 13. Relation between planar segmentation thresholds and planar regions segmented (in each different colour).
The rest of parameters are selected testing over and under-segmentation in dataset 1. The values \( k \) and \( nth \) have the same effect in next phases than in planar segmentation, \( np \) is used to filter regions with few points, \( rds = 0.1 \) m offers a robust solution against small occlusions and low density and the variation of \( vth \), the zone between road and sidewalks are expanded or decreased. Once parameters are fixed for a good segmentation (summarised in Table 4), the same values are employed in the rest of datasets.

Table 4. Values of parameters and thresholds

<table>
<thead>
<tr>
<th>Phase</th>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1.1 Planar segmentation</td>
<td>( k )</td>
<td>50 neighbours</td>
</tr>
<tr>
<td></td>
<td>( nth )</td>
<td>0.5 rad</td>
</tr>
<tr>
<td></td>
<td>( d )</td>
<td>0.2 m</td>
</tr>
<tr>
<td>3.1.2 Split refinement</td>
<td>( rds )</td>
<td>0.1 m</td>
</tr>
<tr>
<td></td>
<td>( np )</td>
<td>25 points</td>
</tr>
<tr>
<td>3.1.3 Merge refinement</td>
<td>( nth )</td>
<td>0.2 rad</td>
</tr>
<tr>
<td>3.1.4 Coplanar refinement</td>
<td>( np )</td>
<td>100 points</td>
</tr>
<tr>
<td>3.1.5 Road-Sidewalk refinement</td>
<td>( k )</td>
<td>100 neighbours</td>
</tr>
<tr>
<td></td>
<td>( vth )</td>
<td>0.0005</td>
</tr>
</tbody>
</table>

4.3. Results

The accuracy of the segmented and classified point clouds is quantified by comparing the results against reference data manually produced [57]. The level of agreement between the result and the ground truth is the indication of the quality of the procedure. To validate the automatic segmentation results, ground truth data were manually generated, segmenting and classifying regions after the planar segmentation process using the software CloudCompare. Two different segmented point clouds per each test were created: one manually segmented and classified beyond each dataset into different elements and the other automatically the result of the methodology applied. The algorithm is implemented in Matlab and processed on an Intel Core i7 CPU 3.40 GHz with 16GB RAM. The processing time is 497 s (about 8 min), 519 s (about 8 min), 268 s (about 4 min) and 716 s (12 min), in datasets 1 to 4, respectively.

Fig. 14 shows all methodology phases applied to the first dataset. After a planar segmentation, ground is segmented into planar regions (Fig. 14.a). Split refinement procedure segments discontinuous and coplanar regions (Fig. 14.b) and merge refinement joins connected regions with a similar normal (Fig. 14.c). As the last riser is in the same region as the vertical wall next to the sidewalk, and roads are in the same region as sidewalks. It is necessary to apply a coplanar refinement (Fig. 14.d) and road-sidewalk segmentation (Fig. 14.e). Fig. 14.f shows point cloud after classification. Colour coding for representing the results is as follows: roads in dark grey, sidewalks in olive, curbs in yellow, risers in green, treads in blue and other elements in light grey.
Dataset 1 in all phases: a) planar segmentation, b) splitted regions, c) merged regions, d) regions with riser segmented from walls, e) regions with road segmented, f) output point cloud with classified elements.

Fig. 15 shows the results for the remaining case studies: dataset 2 (Fig. 15.a), dataset 3 (Fig. 15.b) and dataset 4 (Fig. 15.c).
Evaluation method is based on precision, recall and F1 [58]. Precision represents the percentage between true positives (TP) (Eq. 2) and all positives -true positives (TP) and false positives (FP)-, and recall between true positives (TP) and truth data –true positives (TP) and false negatives (FN)- (Eq. 3). F1 balances precision and recall indices (Eq. 4). It is an effective evaluation method that provides information about segmentation and classification, used by many authors [57,59,60].

\[
\text{precision} = \frac{|\text{TP}|}{|\text{TP}| + |\text{FP}|} \tag{2}
\]

\[
\text{recall} = \frac{|\text{TP}|}{|\text{TP}| + |\text{FN}|} \tag{3}
\]
Points are considered as true positives when they are classified in the same class as the reference data. In the other case, points are false positives (classified points out of reference data) or false negatives (wrong or unclassified points in reference data). Tables 6 to 8 show quantified evaluation results of methodology quality, separated by classified elements. Table 9 shows a summary of all datasets.

The results show a better performance in large elements. Road points are correctly classified in a 99%; the use of trajectory is the reason of the high success rate in this class. Sidewalks also are correctly classified into a 97%, caused by
their large geometry and the greater number of points. Small segments wrongly segmented cause the 3% error. On the other side, smaller elements have a slightly lower success rate. Treads have a higher success rate in precision than in recall, meaning that there are more FN than FP. Points belonging to tread class are correctly classified in 96%. Because treads are horizontal elements, they are more difficult to acquire and they typically present a lower point cloud density. Risers do not have this problem, but an incorrect classification of treads influences their adjacency graphs. Even then, their success rate is around 92%, better than the other small elements, except in the dataset 3, where the number of FP points is elevated because the noise of bushes and an inclined garden. At last, curbs have high point cloud density because of their proximity to the MLS trajectory. Their unambiguous adjacency with roads causes not topological errors. However, curbs have a great number of FN caused by their small size between two larger elements as roads and sidewalks, that produces they are not segmented from these large elements.

The topologic information combined with geometry improves the classification. In Fig. 16, F1 index of treads, risers and curbs (the only elements classified with geometric and topologic information) are compared before and after topologic classification. There is an increment of 10% to 20% on the F1 index with the use of the topology and the graph library.

Fig. 16. F1 index variation with only the use of geometric information (geom) or combined use of geometric and topologic information (geom & top).

For a more detailed error explanation, Table 10 shows the errors classified by source: acquisition, segmentation and classification. In each dataset, the final errors were tracked along the methodology phases and associated to a source when they happen.

Table 10. Classification by error type.

<table>
<thead>
<tr>
<th></th>
<th>Total</th>
<th>Acquisition</th>
<th>Segmentation</th>
<th>Classification</th>
<th>% Acquisition</th>
<th>% Segmentation</th>
<th>% Classification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dataset 1</td>
<td>21033</td>
<td>0</td>
<td>13561</td>
<td>7472</td>
<td>0,00%</td>
<td>64,47%</td>
<td>35,53%</td>
</tr>
<tr>
<td>Dataset 2</td>
<td>22923</td>
<td>2299</td>
<td>18702</td>
<td>1922</td>
<td>10,03%</td>
<td>81,59%</td>
<td>8,38%</td>
</tr>
<tr>
<td>Dataset 3</td>
<td>52876</td>
<td>2744</td>
<td>42574</td>
<td>7558</td>
<td>5,19%</td>
<td>80,52%</td>
<td>14,29%</td>
</tr>
<tr>
<td>Dataset 4</td>
<td>8238</td>
<td>4853</td>
<td>2124</td>
<td>1261</td>
<td>58,91%</td>
<td>25,78%</td>
<td>15,31%</td>
</tr>
<tr>
<td>TOTAL</td>
<td>105070</td>
<td>9896</td>
<td>76961</td>
<td>18213</td>
<td>9,42%</td>
<td>73,25%</td>
<td>17,33%</td>
</tr>
</tbody>
</table>

Many phases of the methodology are based on the continuity and the proximity between points and regions. Therefore, quality of the input data has influence on the results. In urban point cloud near MLS trajectory, the density of the points is 800 points/meter², but working with entire point clouds also there are zones far away from MLS trajectory, many elements cannot be acquired or they are acquired with low density, for example, in the high part of the stairs in dataset 3 (Fig. 17.a) density value fall until 200 points/meter² and sidewalk was correctly classified. In dataset 4 (Fig. 17.b), in a similar case, the density is only 130 points/meter², insufficient to classify. In these cases, adjacent relations between elements do not exist and graphs cannot be verified (Fig. 17.b and 17.c).
Fig. 17. Acquisition with low density and occlusions in the border of dataset 2 (a,c) and dataset 4 (b).

Errors in the first steps of methodology (segmentation) lead to errors in the last steps (classification). Segmentation and refinement errors are found in several datasets (Fig. 18). In several cases, segmentation produces erroneous results due to the quality of the input data. Fig. 18.a shows two treads in the same region as the sidewalk due to the presence of clutter points joining these elements. If small elements are not segmented, they cannot be classified correctly. For example, very small elements such as curbs with less than 5 cm of height are not segmented in Fig. 18.a and Fig. 18.b.

Fig. 18. Examples of wrong segmentation: treads not segmented from sidewalk in dataset 2 (a) and 3 (b).

At last, in urban scenes there can be elements not considered for classification, but with geometry and topology identical to another class such as in case of Fig. 19 where false positives were obtained.

Fig. 19. False classification examples: false tread and risers (a) and false curb (b), both from dataset 2.

4.4. Discussion

The results show high success rate from 82% for class curbs to 99% for class roads. The results are better in large elements such as sidewalks and roads, than in small elements, such as curbs, treads and risers. Large elements have more number of points, their geometry is more defined, less sensitive to occlusions and confusions with another elements than small elements. In any case, these results are enough for a posterior accessibility analysis of urban scene. The methodology works in complex urban zones, the study is not limited to one street with buildings on both sides. All the datasets are zones with several floors with different altitudes, connected by stairs, deformations, historical elements, and with a lot of small occlusions and noise caused by persons and cars. Also, there are areas with trees and with strong variations of density and important occlusions far the MLS trajectory.
The methodology begins with a planar segmentation of the point cloud, based on fixed parameters obtained from a study of their impact on the results. The value of the parameters is a compromise solution between under and over-segmentation for the refinement phases. The segmentation is the process more sensible of the methodology and it is determinant in the classification, the high level of detail needed for the ground classification involves the requirement of four refinement processes. A double classification is based on geometric information with a tree decision. The classification combines the use of geometric information, based on a decision tree with parameters from ISO- 25142, and topologic information, a graph library of elementary graphs of each class. The combined use of geometry and topology improves the final results in a 10 to 20% F1 index variation than only the use of geometry.

The high variation of the density involves different results near and far trajectory. A solution for this problem could be the use of adaptive parameters based on point density, for example, the KNN search after a density analysis of points, and new graphs based on proximity, instead of adjacency, where it exists low density and empty zones. The MLS point cloud have important occlusions that limit a complete analysis of a scene. Actual point clouds (from a MLS in a van) can be combined with point clouds from another mobile system, for example a laser scanner mounted in a drone or a backpack, to complete the scene and to cover the occlusions. Of all phases, the segmentation is the most susceptible to errors. The improvement of this phase will have an important impact on final results and in time processing.

5. Conclusions

We propose a novel methodology to automatically classify urban ground elements from 3D point clouds into five classes: roads, sidewalks, treads, risers and curbs. The detailed classification of urban ground elements is essential to understand in-depth the as-built urban environments, to represent smart city models including geometry, semantic and topologic information [61], and more specifically to accessibility diagnosis and path planning applications.

The methodology is based on combining geometric and topologic information. The point cloud is submitted to a planar segmentation and subsequent to split and merge operations, refining the region detection for a high level of detail classification. Following this, geometrical features are analysed and used to preliminary classify regions according to a decision tree. Finally, adjacency between regions is extracted and elements are verified and definitively classified by comparing their topology graphs with a graph library. The library is composed by a set of predefined graphs, and it could be extended to include more classes.

The methodology was tested in four real case studies acquired with a MLS. Results show an average of 82% to 99% of well-classified points depending on the class. These results are better in big elements such as roads and sidewalks since they have more points and they are less sensitive to false positives and false negatives. The combination of topology and geometry improves a 10% to 20% the success rate obtained with only the use of geometry. The origin of errors is also analysed and the results conclude that segmentation is the weak step of the methodology (73.25%) due to irregularities on the as-built infrastructures and to the quality of the input data. An erroneous segmentation generally implies an erroneous classification. Errors purely due to classification are around 17.33%. They mostly correspond to classes that are not considered in the graph library.

Finally, the results show enough information to classify urban areas according to accessibility rules. Future lines aim to improve the segmentation methodology (adaptive parameters or new features) to deal with the irregularities of the built environment and with the quality of input data. Furthermore, the graph library could be extended to include more events and classes as vegetation (trees or bushes) or elements connected to the ground (benches, bollards, traffic signals, lamppost, etc.) and including, relations of proximity or no-connectivity.

6. Acknowledgements

Authors would like to thank to SCI-EYSA for the financial support given through the University Chair in Smart cities and road safety (“Catedra SCI-EYSA, Smart cities e seguridud vial”), to “Xunta de Galicia” for the financial support given through human resources grant (ED481B 2016/079-0), to the “Ministerio de Economia y Competitividad -Gobierno de España” - (TIN2016-77158-C4-2-R) and to the European Union –H2020- (ENGINENCY). The statements made herein are solely the responsibility of the authors.
7. References and Notes


