

## PEDESTRIAN PATHFINDING IN URBAN ENVIRONMENTS: PRELIMINARY RESULTS

G.López-Pazos<sup>a</sup>, J.Balado<sup>b,1</sup>, L. Díaz-Vilariño<sup>b,c</sup>, P. Arias<sup>b</sup>, M. Scaioni<sup>d</sup>

<sup>a</sup> School of Industrial Engineering, Campus Lagoas-Marcosende, Vigo, 36310 Spain - [gulopez@alumnos.uvigo.es](mailto:gulopez@alumnos.uvigo.es)

<sup>b</sup> Applied Geotechnologies Group, Dept. Natural Resources and Environmental Engineering, University of Vigo, Campus Lagoas-Marcosende, Vigo, 36310 Spain - ([jbalado](mailto:jbalado@uvigo.es), [lucia\\_parias](mailto:lucia_parias@uvigo.es))@uvigo.es

<sup>c</sup> Faculty of Engineering of University of Porto, Research Centre for Territory, Transport and Environment (CITTA), Rua Dr. Roberto Frias, s/N, Porto, Portugal - [lucia@fe.up.pt](mailto:lucia@fe.up.pt)

<sup>d</sup> Dept. of Architecture, Built Environment and Construction Engineering, Politecnico di Milano, via Ponzio 31, 20133 Italy - [marco.scaioni@polimi.it](mailto:marco.scaioni@polimi.it)

Commission V, WG V/7

**KEY WORDS:** Accessibility, As-built 3D, Dijkstra Pathfinding Algorithm, Graphs, Obstacle detection, Point Cloud Classification, Pedestrian, Smart Cities, Topology

### ABSTRACT:

With the rise of urban population, many initiatives are focused upon the *smart city* concept, in which mobility of citizens arises as one of the main components. Updated and detailed spatial information of outdoor environments is needed to accurate path planning for pedestrians, especially for people with reduced mobility, in which physical barriers should be considered. This work presents a methodology to use point clouds to direct path planning. The starting point is a classified point cloud in which ground elements have been previously classified as roads, sidewalks, crosswalks, curbs and stairs. The remaining points compose the obstacle class. The methodology starts by individualizing ground elements and simplifying them into representative points, which are used as nodes in the graph creation. The region of influence of obstacles is used to refine the graph. Edges of the graph are weighted according to distance between nodes and according to their accessibility for wheelchairs. As a result, we obtain a very accurate graph representing the as-built environment. The methodology has been tested in a couple of real case studies and Dijkstra algorithm was used to pathfinding. The resulting paths represent the optimal according to motor skills and safety.

### 1. INTRODUCTION

Smart cities are a new model of integrating information that people may need to live in them. One of the most important things included in smart cities is related to the information about transport and the movement of the citizens: an effective route calculation may save money and time. The main research in this field focuses the efforts on vehicles, either in autonomous driving (Djahel et al., 2015; Ngoc Nha et al., 2012) and in parking assistance (Atif et al., 2016; Orrie et al., 2015; Rhodes et al., 2014), leaving aside pedestrian needs in their displacements.

Most of pathfinding tools used in the urban environment are designed for cars. If the user is changed to bicycles or pedestrians, the algorithm only adapts travel speed while the network is the same. So, updated spatial models representing the as-built environment, especially in terms of ground elements are needed for accurate path planning. Díaz-Vilariño et al. (2016) developed a methodology for 3D navigation in indoor environment that includes modelling and obstacle detection. Ivanov (2017) focuses on an indoor navigation system for visually impaired people based on Building Information Models. Persons with reduced mobility (PRM) and people with disabilities are especially vulnerable to social exclusion because they have more limitation to use routes generated for systems non-specific for pedestrians (Savogin et al., 2006). In fact, any step may be an insurmountable barrier, even though they must have the same rights as the rest of citizens (UN, 2014). In this context, it is important to differentiate motor skills of pedestrians

(Hashemi and Karimi, 2010). Some projects, have tried to offer solutions for pedestrian mobility. Prandi et al., (2014) present a project for collecting points of interest that are important for the mobility with smartphones, such as obstacles or crosswalks.

At the same time, in the field of robotics, route calculation is also carried out so that robots may define their trajectories autonomously (Oßwald et al., 2011). They are often mounted on platforms with legs or wheels, which implies great similarity to people motor skills. Also, the algorithms for pathfinding are well known in Real Time Strategy games (Hu et al., 2012), where the movement of the units are conditioned by their different motor skill, the terrain features and obstacles.

Most of the time, an accessibility diagnosis is a previous requisite before route calculation. It can be a curb detection (Serna and Marcotegui, 2013) or completed with the detection of ramps, sidewalks and roads. Even the diagnosis should be extended to the building entrances as areas of interest that allow the transition between indoor and outdoor environment (Balado et al., 2017a).

Geometry information and more detailed models are necessary for evaluation of urban environments. Mobile Laser Scanning (MLS) systems are a consolidated technology to acquire large urban zones with high point density and precision that make possible the detection even of small steps (Balado et al., 2017c). The analysis of urban point clouds involves dealing

<sup>1</sup> Corresponding author

with a large amount of unstructured data with strong density variations and occlusions.

The recent progress in urban classification may help in the process of route calculation, although the main efforts have been done in object classification (Golovinskiy et al., 2009; Ramirez Pedraza et al., 2015; Yang et al., 2017) or in a low level of detail, classifying buildings, ground and few objects (Babahajiani et al., 2014; Weinmann et al., 2015). The ground is not passable to the same extent and the different elements that make it may have diverse influence on route calculation. Balado et al. (2017b) classify urban ground elements in point clouds (sidewalks, roads, curbs and stairs) with a high level of detail. Riveiro et al. (2015) detect zebra crossings, what is very relevant to locate the areas where a pedestrian may cross roads without endangering his life and cause traffic accidents.

The aim of this work is to develop a real outdoor 3D path planning methodology to provide safe and viable routes for pedestrians with different motor skills and considering barriers for persons with reduced mobility. The initial point cloud is transformed to a graph, whose nodes are weighted according to the type of element they are representing. Elements are splitted with connected components, a robust way against deformations that appear in as-built environment, and subsampled with K-means to reduce the final number of nodes of the graph. The region of influence of the obstacles is deleted from ground elements to not interfere in route calculation. The road regions are not considered as part of the final graph because are not safe transit areas. The edges of the graph are assigned values depending on the distance between points and if the elements are accessible for wheelchairs.

The rest of this paper is organized as follows. Section 2 provides an overview of the classification method developed for urban ground elements. In Section 3 the concept of using point clouds directly to pathfinding is described. Section 4 explains the conducted experiments while Section 5 is aimed to conclude this work.

## 2. OVERVIEW OF URBAN GROUND CLASSIFICATION

Intense efforts have been made in the recent years in terms of point cloud classification. Despite of the fact that a considerable number of approaches deal with classification of objects such as buildings, roads, poles, cars and trees, a very few works address the classification of ground elements. The modelling of the ground environment is essential to understand somehow people may move inside the urban environment and more specifically, it is of a great importance for those people with reduced mobility.

This section briefly describes the point cloud classification approach, whose results constitute the starting point of this work. The methodology developed by Balado et al. (2017b) classifies ground elements in five classes: roads, sidewalks, curbs and stairs (divided into risers and treads). The rest of urban elements (walls, façades, pedestrian, cars, etc.) are classified in another category as obstacles.

The input consists of an urban point cloud without RGB or intensity information and the trajectory of the MLS vehicle during data acquisition. The first phase of the classification workflow implements a planar segmentation technique, since most of the ground elements can be modelled as planar regions. The planar segmentation must be refined with a sequence of

processes in order to obtain a high level of accuracy. Split and merge operations are used to generally refine the segmentation. Then, a coplanar refinement is used to segment two coplanar and continuous elements based on height difference. And finally, a road-sidewalk refinement is implemented to separate road from sidewalk when they are initially merged in the same planar region due to the presence of small deformations on the as-built surfaces.

Once the point cloud is segmented in individual elements, next phase implements a geometric classification based on a decision tree according to parameters defined by the ISO-25142 (ISO, 2011). In the last phase, the preliminary classification is verified from the comparison of topological relationships between contiguous elements. These topological relationships are represented in graphs, and compared to a graph library. To avoid isomorphism problem, graphs are reduced to small and meaningful units, which are easy to compare. The objective of the topologic classification is to distinguish elements with the same geometry such as risers and curbs.

Other elements very important to pedestrian routing in urban environments are crosswalks, since they allow pedestrians to use them as safe zones when crossing roads according to traffic rules. Crosswalks can be detected on roads by analysing point cloud intensity (Riveiro et al., 2015), which is an additional information still scarcely used in point cloud segmentation and classification, but able to provide relevant support (Crosilla et al., 2014). As traffic and road signs have a high value of intensity in comparison with the rest of points in the point cloud, they may be easily detected. In this approach, after implementing a point cloud rasterization, the Hough Transform is applied on the grid to detect lines of zebra crossings. The result of the application of previous methodologies on an urban point cloud is shown in Figure 1.

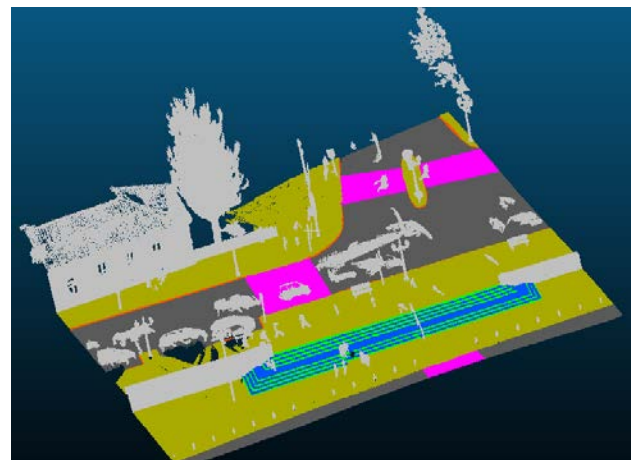


Figure 1. Point cloud classified by type of elements. Colour code: sidewalks in olive, risers in green, treads in blue, curbs in orange, crosswalks in rose, road in dark grey, and obstacles in light grey.

## 3. METHODOLOGY FOR GRAPH RECONSTRUCTION FROM POINT CLOUDS

The main purpose is to create a graph from the classified point cloud to enable direct pathfinding. In this way, the graph would represent the as-built ground with a very high level of detail, and

semantics could be used to weight the graph according to the pedestrian motor skills, of special relevance to people with reduced mobility such as wheelchair people.

The classified point cloud constitutes the starting point. First of all, ground elements are simplified to representative nodes. Afterwards, obstacle class is used to refine those nodes. Finally, nodes are used to create a graph, in which pathfinding algorithms may be applied to calculate the minimum cost path. Figure 2 shows the general workflow of the developed methodology for graph reconstruction from point clouds. In the following paragraphs, different tasks in the processing workflow will be described in detail.

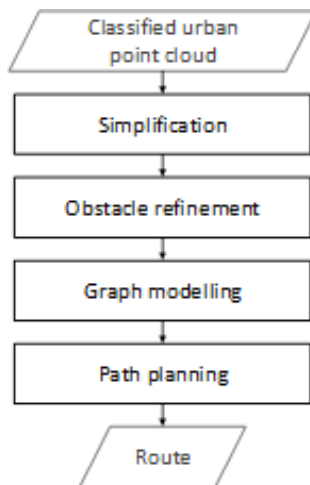


Figure 2. Workflow of the methodology for graph reconstruction from point clouds.

### 3.1 Point cloud simplification

In this approach, the navigation graph is directly created from the point cloud. However, the size of input data is directly related to the running time of algorithms used to create the graph and to pathfinding functions. Therefore, point cloud simplification into meaningful points is the key phase of this approach, because meaningful points will be used as nodes in the graph. According to the class, a different process of simplification is used.

**3.1.1 Sidewalks and crosswalks:** Firstly, each class is processed to individualize different elements within the class. For this purpose, *connected components* clustering is used, clustering data into groups by proximity (see Fig. 3).

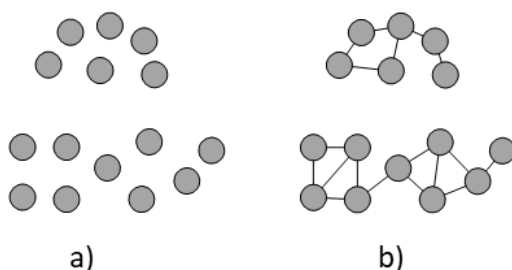


Figure 3. Points before (a) and after (b) the application of *connected component* clustering.

In the following step, K-means algorithm (Samet, 2006) is used to group points within each element by medium value of coordinates. The number of clusters is calculated according to an estimated area of a human step  $s = 0.5$  m. Eventually, the centroid of each cluster is calculated to finish the simplification of these two classes (see Fig. 4).

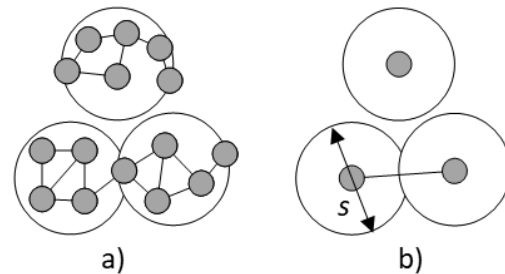


Figure 4. Point cloud after (a) and before (b) simplification based on K-means.

**3.1.2 Curbs:** Meaningful points of this class are those near crosswalks because adjacency with roads is not interesting when considering navigable nodes.

As in Section 3.1.1, connected components are used to individualize different elements within the class. Since just curb points connecting sidewalks and crosswalks are meaningful for navigation, they should be extracted. For this purpose, an exterior buffer for every crosswalk is created. Curb points within the buffer are considered for further processing since they are assumed to be the curb points connecting sidewalks and crosswalks (see Fig. 5).



Figure 5. Curbs before simplification (a) and after simplification (b). The same colour code adopted in Figure 1 are used here.

**3.1.3 Stairs:** Stairs are classified into two classes: risers and treads. In contrast to treads, risers are not considered navigable parts of the stairs, so the points of this class are not used to build the graph.

The first processing step applied to tread class is connected components clustering. This algorithm is used one time to individualize different stairs, and twice to individualize treads themselves.

It is possible that, due to occlusions or noise caused during point cloud acquisition, one tread is oversegmented. To solve this, using the middle value of the Z coordinate for each part, it is possible to group them, leaving a minimum margin for possible minimal variations. Next step is K-means clustering and calculation of the centroids of each group, as in case of sidewalks and crosswalks (see Fig. 6).

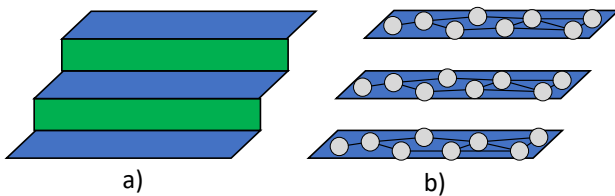


Figure 6. Stairs modelled: a) initial point cloud of stairs, b) treads grouped by *connected components* clustering and reduced. Colour code: treads in blue and risers in green.

An alternative to this process is to make a histogram of the  $z$  coordinates as proposed in Luo et al. (2013), which represents the accumulation of points according to height. The main problem is the error caused by misalignment in the construction of stairs, so many data is lost in the process due to values in the histogram with low concentration, ignored when maxima are taken.

### 3.2 Obstacle refinement

One of the goals in route mapping is to avoid obstacles. Therefore, obstacle class is used in this approach to delete those ground points that could be affected by the presence of obstacles.

First of all, the size of the obstacle class is reduced with a number-point division filter. Furthermore, only those points belonging to the obstacle class and with a relative height below  $h=2$  m are considered for potential obstruction of pedestrian navigation. The remaining points with height less than  $h$  are projected along  $z$  direction and they are converted to a boundary polygon.

The polygons of the obstacles are compared to the points obtained after simplification step (Subsect. 3.1). If a point falls into a polygon, this point is deleted from the simplified model. This check is applied to all classes in the final model (stairs, curbs and sidewalks) except crosswalk, where the authors assume that there should not be any obstacles. Figure 7 shows the process of obstacle refinement.

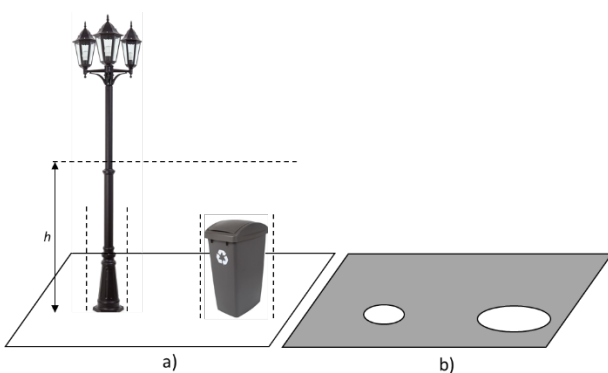


Figure 7. Obstacle refinement: a) obstacles with  $h$  limitation and projected, b) region without polygon belonging to obstacles.

### 3.3 Graph modelling

Once the point cloud is simplified into meaningful points for navigation, it is used to directly create the graph. It should be noted that roads and obstacles are not taken into account for not

being considered navigable. In this way, every point of the simplified data is a node of the graph.

The maximum size of the edge joining two nodes is the main variable for the graph creation. A too small may results in a disconnected graph, as can be seen in Figure 8a. On the other hand, a very large edge size may create too many joints (Fig. 8b), increasing the running time of the search algorithms, and joining non-adjacent nodes. Besides, it is necessary to distinguish between edges in the stairs and the rest of classes because there is greater accumulation of nodes. For that, it is necessary to establish relations between nodes at the edge of the stairs and nodes of the other classes.

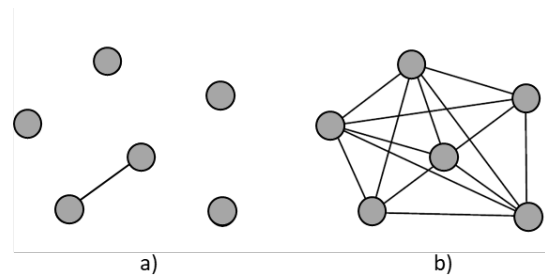


Figure 8. Graph creation in the case of too short (a) or too long (b) edges.

The creation of the graph entails a cost matrix to evaluate the efficiency of every edge. The cost typically is the Euclidean distance between nodes, so the coordinates of every point are taken into account. One of the purposes of this work is to avoid the stairs if the user is a disabled person, so an infinite cost is associated to every edge that contains a stair or curb class node.

### 3.4 Pathfinding

Search algorithms use typically graphs for the pathfinding. To ensure the effectiveness of finding the optimum way it is important to establish a cost criterion as real as possible, for example, Euclidean distance between nodes.

One of these functions is Dijkstra's pathfinding algorithm (Soltani et al., 2002; Kang et al., 2008), which computes the optimal route from a node to the rest of the nodes in a graph, or between two established nodes, taking into account the cost associated to every edge on the basis of the cost matrix.

## 4. RESULTS

The methodology has been tested in two data sets. Results are illustrated and discussed in the following subsections. In both examples, the methodology was implemented in Matlab® running on a computer based on Intel Core i7-4712MQ, CPU 2.3GHz with 8GB RAM processor.

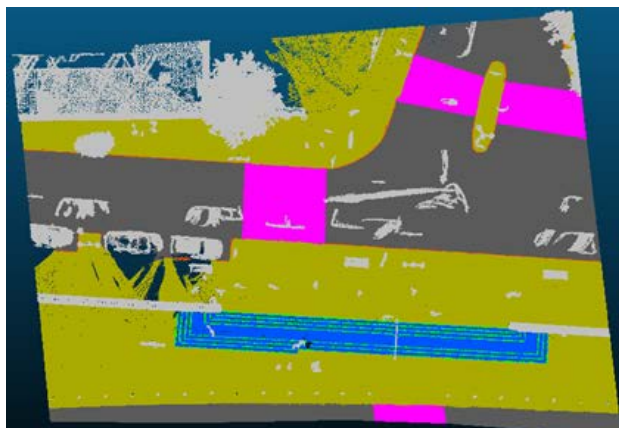
### 4.1 Data Set 1

The first data set consists of a point cloud made up of 1.6 million points of the Humilladero street in Ávila (Fig. 9a). It is a historical centre with four crosswalks (one of them is not complete), several sidewalks built with stone and stairs with some deformations. With the simplification, the number of total points is reduced significantly, which facilitates the creation of

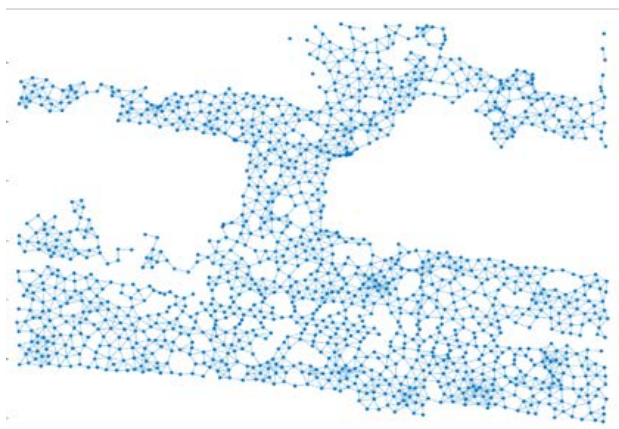
the graph (Fig. 9b). The processing time of the methodology application is shown in Table 1.

The variation in the point cloud involves that the final graph has different number of edges with respect to the optimal solution. In the central top of Figures 9a and 9b, a zone with low density is shown, where the number of nodes in the final graph is visibly smaller than in other zones with higher density. The problem is more serious with the occlusions. In the left zone of the Figure 9a, occlusions caused by parked cars are shown. As result, in the same zone of Figure 9b, the final graph has, in addition to low density, an absence of continuity in the graph nodes. This break may lead to errors in pathfinding.

The pathfinding algorithm implemented on Data set 1 was the Dijkstra and took 0.5 sec to find a route between the bottom of the stairs and the farthest crosswalk (Figure 10). The route is perfectly viable and safe for pedestrians, crossing sidewalks, stairs and crosswalks in a real way without the interference of obstacles.



a)



b)

Figure 9. Results of Data Set 1: in the upper window (a) the classified point cloud is shown (the same colours adopted in Figure 1 have been applied to distinguish classes); in the lower window, the connection graph obtained with the methodology described in Section 3.

Process	Time (s)
Pre-processing	35.1
Simplification	29.4
Sidewalks	4.4
Crosswalks	1.0
Curbs	4.9
Stairs	19.0
Obstacles	1.0
Graph creation	0.5
<b>Total processing time</b>	<b>66.1</b>

Table 1. Time processing of the methodology applied to Data Set 1.

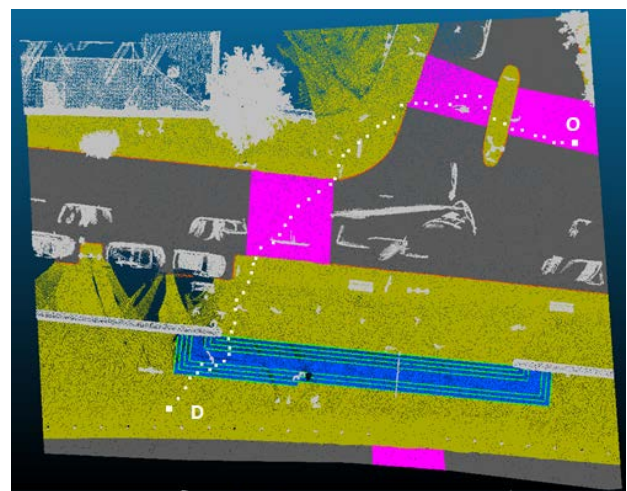


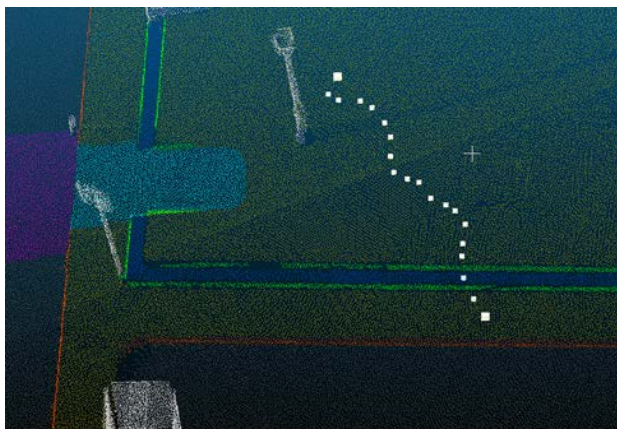
Figure 10. Final route computed in Data Set 1 as demonstration of the application of the developed methodology.

#### 4.2 Data Set 2

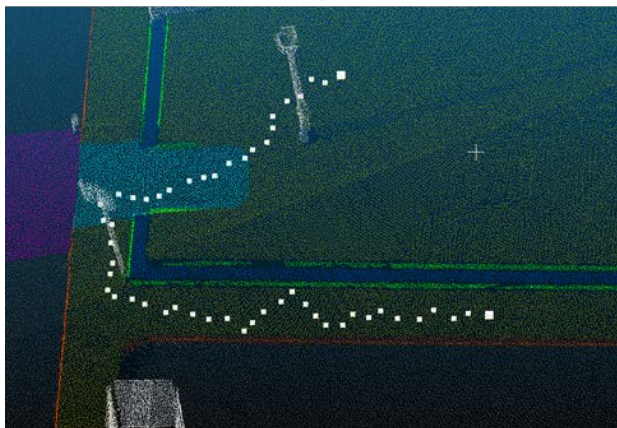
The second data set corresponds to a smaller point cloud, located in La Feria square, in Ávila, with the objective to test variations in a route for wheelchair users. Associating infinite cost to the edges of the stairs, the search algorithms get to calculate the route avoiding these urban elements, as shown in the difference between Figure 11a, where the route is viable for pedestrians crossing stairs, and Figure 11b, where the route cross a ramp to avoid the stairs. Both cases were tested with Dijkstra pathfinding algorithm. As it can be seen in Figure 11b, the route follows a trajectory excessively near of the border of the ramp. This is because the algorithm searches for the closest route. In response to this situation, the nodes of the border of the elements should not be considered for pathfinding.

#### 5. CONCLUSIONS AND FUTURE DEVELOPMENTS

In this work, a methodology for direct path planning from point clouds is introduced. This methodology is based on simplification and modelling of a classified point clouds to create graph models where traditional pathfinding algorithms can be applied, e.g., Dijkstra algorithm.



a)



b)

Figure 11. Final route calculation with Dijkstra pathfinding algorithm in Data Set 2 in the case of (a) pedestrian route and (b) wheelchair route. Ramps are painted in light blue

In this way, graphs represent the as-built environment in a very accurate way, not only from a geometric point of view but also from a semantics perspective since edges can be weighted according to the accessibility of the element they are representing. The methodology considers the avoidance of obstacles depending on the motor skills of pedestrians, including either normal and disabled people on wheelchairs. The methodology was tested in two case studies obtaining positive preliminary results. The computing time of graph creation is small considering the number of points processed and paths calculated by the pathfinding algorithm are viable and safe, for both pedestrians or wheelchairs.

Future work involves refining the process of graph creation, making it less sensitive to variations of density, especially in zones of low density and occlusions. Also the computing time will be determined and related with the density and the size of the point cloud. At final, the methodology will be tested in other real case studies with more elements and different challenges to evaluate its robustness.

#### ACKNOWLEDGEMENTS

Authors would like to thank to SCI-EYSA for the financial support given through the University Chair in Smart cities and

road safety (Cátedra SCI-EYSA. Smart cities e seguridade vial), to Xunta de Galicia for the financial support given through human resources grant (ED481B 2016/079-0), to the Ministerio de Economía, Industria y Competitividad -Gobierno de España- (TIN2016-77158-C4-2-R, RTC-2016-5257-7) and to the European Union –H2020- (ENGINENCY). The statements made herein are solely the responsibility of the authors.

#### REFERENCES

- Atif, Y., Ding, J., and M.A. Jeusfeld, 2016. Internet of Things Approach to Cloud-based Smart Car Parking. *Procedia Computer Science* 98, 193–198, DOI:10.1016/j.procs.2016.09.031.
- Babahajiani, P., Fan, L., and M. Gabbouj, 2014. Object Recognition in 3D Point Cloud of Urban Street Scene. In: Proc. "12<sup>th</sup> Asian Conf. on Computer Vision (ACCV 2014)," Singapore, 1-5 November 2014, pp. 1-14.
- Balado, J., Díaz-Vilariño, L., Arias, P., and I. Garrido, 2017a. Point Clouds To Indoor / Outdoor Accessibility Diagnosis. *The ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Wuhan, China, Vol. IV, Part 2/W4: 18-22, DOI: 10.5194/isprs-annals-IV-2-W4-287-2017.
- Balado, J., Díaz-Vilariño, L., Arias, P., and H. González-Jorge, 2017b. Automatic classification of urban ground elements from mobile laser scanning data. *Automation in Construction*, Vol. 84, DOI:10.1016/j.autcon.2017.09.004.
- Balado, J., Díaz-Vilariño, L., Arias, P., and M. Soilán, 2017c. Automatic building accessibility diagnosis from point clouds. *Automation in Construction*, Vol. 82: 103-111, DOI:10.1016/j.autcon.2017.06.026.
- Crosilla, F., Macorig, D., Scaioni, M., Sebastianutti, I., and D. Visintini, 2013. LiDAR data filtering and classification by skewness and kurtosis iterative analysis of multiple point cloud data categories. *Applied Geomatics*, Vol. 5(3): 225-240, DOI: 10.1007/s12518-013-0113-9.
- Díaz-Vilariño, L., Boguslawski, P., Khoshelham, K., Lorenzo, H., and L. Mahdjoubi, 2016. Indoor navigation from point clouds: 3D modelling and obstacle detection. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Prague, Czech Republic, Vol. XLI, Part B4: 275-281, DOI:10.5194/isprsarchives-XLI-B4-275-2016.
- Djahel, S., Doolan, R., Muntean, G.-M., and J. Murphy, 2015. A Communications-Oriented Perspective on Traffic Management Systems for Smart Cities: Challenges and Innovative Approaches. *IEEE Communications on Surveys and Tutorials*, Vol. 17: 125-151, DOI:10.1109/COMST.2014.2339817.
- Golovinskiy, A., Kim, V.G., and T. Funkhouser, 2009. Shape-based recognition of 3d point clouds in urban environments. In: Proc of "IEEE 12<sup>th</sup> Int. on Computer Vision," Kyoto, Japan, 29 September - 2 October, pp. 2154-2161, DOI:10.1109/ICCV.2009.5459471.
- Hashemi, M., and H.A. Karimi, 2010. Indoor Spatial Model and Accessibility Index for Emergency Evacuation of People with Disabilities. *Journal of Computing in Civil Engineering*, Vol. 30(4): 1-16, DOI:10.1061/(ASCE)CP.1943-5487.0000534.

- Hu, J., Gen Wan, W., and X. Yu, 2012. A pathfinding algorithm in real-time strategy game based on Unity3D. In: Proc. of "2012 Int. Conf. on Audio, Language, Image Processing," Shanghai, China, 16-18 July 2012, pp. 1159-1162, DOI:10.1109/ICALIP.2012.6376792.
- ISO, 2011. *ISO-25142 Building construction - Accessibility and usability of the built environment*. ISO Int. Organization of Standardization.
- Ivanov, R., 2017. An approach for developing indoor navigation systems for visually impaired people using Building Information Modeling. *Journal of Ambient Intelligence and Smart Environment*, Vol. 9(4): 449-467, DOI:10.3233/AIS-170441.
- Kang, H. Il, Lee, B., and K. Kim, 2008. Path Planning Algorithm Using the Particle Swarm Optimization and the Improved Dijkstra Algorithm. In: Proc. of "2008 IEEE Pacific-Asia Work. Computational Intelligence and Industrial Applications (PACIIA '08)," Wuhan, China, 19-20 December 2008, pp. 1002-1004, DOI:10.1109/PACIIA.2008.376.
- Luo, R.C., Hsiao, M., and C.W. Liu, 2013. Multisensor integrated stair recognition and parameters measurement system for dynamic stair climbing robots. In: Proc. of "2013 IEEE Int. Conf. on Automation Science and Engineering (CASE 2013)," Madison, WI, United States, 17-20 August 2013, pp. 318-323, DOI:10.1109/CoASE.2013.6654026.
- Ngoc Nha, V.T., Djahel, S., and J. Murphy, 2012. A Comparative Study of Vehicles' Routing Algorithms for Route Planning in Smart Cities. In: Proc. of "2012 First Int. Work. on Vehicular Traffic Management for Smart Cities (VTM)," Dublin, Ireland, 20-22 November 2012, pp. 1-6, DOI:10.1109/VTM.2012.6398701.
- Orrie, O., Silva, B., and G.P. Hancke, 2015. A wireless smart parking system. In: Proc. of "41<sup>st</sup> Annual Conf. of the IEEE Industrial Electronic Society (IECON 2015)," Yokohama, Japan, 20-22 November 2015, pp. 4110-4114, DOI:10.1109/IECON.2015.7392741.
- Oßwald, S., Gutmann, J.S., Hornung, A., and M. Bennewitz, 2011. From 3D point clouds to climbing stairs: A comparison of plane segmentation approaches for humanoids. In: Proc. of "2011 11<sup>th</sup> IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)," Bled, Slovenia, 26-28 October 2011, pp. 93-98, DOI:10.1109/Humanoids.2011.6100836.
- Prandi, F., Soave, M., Devigili, F., De Amicis, R., and A. Astyakopoulos, 2014. Collaboratively collected geodata to support routing service for disabled people. In: Proc. of "11<sup>th</sup> Int. Symp. on Location-Based Services (LBS 2014)," Vienna, Austria, 26-28 October 2011, 26-28 November 2014, DOI: 10.13140/2.1.2937.1203.
- Ramirez Pedraza, A., Gonzalez Barbosa, J.J., Hurtado Ramos, J.B., Garcia Moreno, A.I., Ornelas Rodriguez, F.J., and E.A. Gonzalez Barbosa, 2015. Detection and segmentation of 3D objects in urban environments using indexation. *IEEE Latin America Transactions*, Vol. 13(4): 1120-1128, DOI:10.1109/TLA.2015.7106365.
- Rhodes, C., Blewitt, W., Sharp, C., Ushaw, G., and G. Morgan, 2014. Smart routing: A novel application of collaborative path-finding to smart parking systems. In: Proc. of "2014 IEEE 16<sup>th</sup> Conf. on Business Informatics (CBI)," Geneva, Switzerland, 14-17 July 2014, pp. 119-126, DOI:10.1109/CBI.2014.22.
- Riveiro, B., González-Jorge, H., Martínez-Sánchez, J., Díaz-Vilariño, L., and P. Arias, 2015. Automatic detection of zebra crossings from mobile LiDAR data. *Optics & Laser Technology*, Vol. 70: 63-70, DOI:10.1016/j.optlastec.2015.01.011.
- Samet, H., 2006. *Foundations of Multidimensional and Metric Data Structures*. Morgan Kaufmann, 1024 pages.
- Savogin I., Scaioni M., and C. Fasoli, 2006. Portable Monitoring and Localisation Systems for Disabled Patients. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Dresden, Germany, Vol. 36, Part 5, 6 pages (e-doc).
- Serna, A., and B. Marcotegui, 2013. Urban accessibility diagnosis from mobile laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 84: 23-32, DOI:10.1016/j.isprsjprs.2013.07.001.
- Soltani, A.R., Tawfik, H., Goulermas, J.Y., and T. Fernando, 2002. Path planning in construction sites: Performance evaluation of the Dijkstra, A\*, and GA search algorithms. *Advanced Engineering Informatics*, 16(4): 291-303, DOI:10.1016/S1474-0346(03)00018-1.
- UN, 2014. *United Nations. World Urbanization Prospects: The 2014 Revision*. New York, United States, DOI:10.4054/DemRes.2005.12.9.
- Weinmann, M., Jutzi, B., Hinz, S., and C. Mallet, 2015. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 105: 286-304, DOI:10.1016/j.isprsjprs.2015.01.016.
- Yang, B., Dong, Z., Liu, Y., Liang, F., and Y. Wang, 2017. Computing multiple aggregation levels and contextual features for road facilities recognition using mobile laser scanning data. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 126: 180-194, DOI:10.1016/j.isprsjprs.2017.02.014.