Geocarto International

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tgei20

Visibility analysis for the occlusion detection and characterisation in street point clouds acquired with Mobile Laser Scanning

Jose Barros-Ribademar, Jesús Balado, Pedro Arias & Silvia María González-Collazo

To cite this article: Jose Barros-Ribademar, Jesús Balado, Pedro Arias & Silvia María González-Collazo (2022): Visibility analysis for the occlusion detection and characterisation in street point clouds acquired with Mobile Laser Scanning, Geocarto International, DOI: 10.1080/10106049.2022.2032392

To link to this article: https://doi.org/10.1080/10106049.2022.2032392

Published online: 08 Feb 2022.

GEOCARTO INTERNATIONAL
https://doi.org/10.1080/10106049.2022.2032392

Taylor & Francis
Taylor & Francis Group

Visibility analysis for the occlusion detection and characterisation in street point clouds acquired with Mobile Laser Scanning

Jose Barros-Ribademar[a], Jesús Balado[b], Pedro Arias[b] and Silvia María González-Collazo[b]

[a]Universidade de Vigo, Escola de Enxeñaría Industrial, Vigo, Spain; [b]CINTECX, Universidade de Vigo, GeoTECH Group, Vigo, Spain

ABSTRACT

Although multi-temporal acquisitions are a widely accepted way to reduce occlusions in point clouds, occlusion analysis continues to be a visual analysis. The objective of this work is the design of an automatic method for the detection and characterization of street point cloud occlusions acquired with Mobile Laser Scanning (MLS). The proposed method consists of four main phases: alignment of the point cloud, rasterization, generation of occluded point clouds and visibility analysis. The proposed method was tested with point clouds of TerraMobil-ita/iQmulus dataset. The result shows a correct detection of 98.71% of the occlusions on ground and 83.37% on façades. All detected occlusions were correctly characterized. In the case studies, cars generated the largest amount of occluded area on the ground, with an occluded area per object of 15.36 m2. The main source of occlusions on façades was the building geometry and windows.

ARTICLE HISTORY
Received 31 August 2021
Accepted 17 January 2022

KEYWORDS
LiDAR; occlusion recognition; occlusion regeneration; 3D data processing; point cloud completion; urban environment

Introduction

Mobile Laser Scanning (MLS) technology is widely used to acquire urban streets with high accuracy in a short time. Acquisitions on the move save a lot of time, and although the MLS accuracy and precision are not as high as with Terrestrial Laser Scanning, the data quality is enough for a correct information extraction and modelling. As a vehicle-mounted LiDAR moves forward, the street is acquired from the vehicle's point of view. MLS data in urban environments are mainly used to study three urban elements: urban objects, ground, and buildings (Sun et al. 2018; Che et al. 2019; Wang et al. 2019; Luo et al. 2020b; Han et al. 2021).

Street point clouds acquired with MLS have some limitations, with occlusions being the most relevant for certain applications (Chen and Yang 2016). Usually, objects in the urban environment are concentrated in a zone between road traffic and pedestrian traffic. Consequently, all these objects (parked cars, furniture, vegetation, etc.) generate occlusions on part of the road, kerbs, sidewalks, curbs, façades, and even other objects (Figure 1).

CONTACT Jesús Balado jbalado@uvigo.es

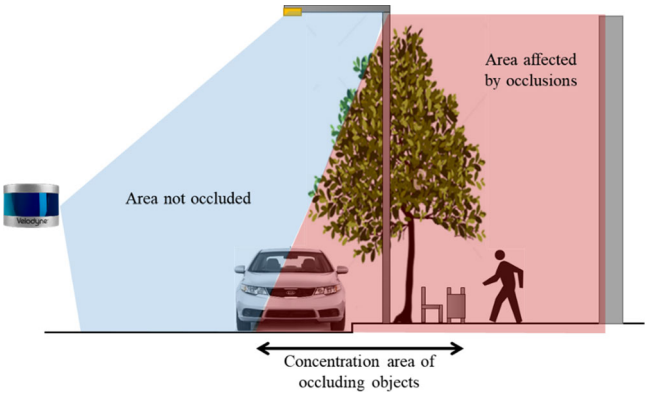


Figure 1. Urban scene configuration and occlusions.

Considering the point of view of the existence of occlusions, point cloud processing applications can be divided into affected, semi-affected and unaffected by occlusions.

- Not occlusion-affected applications are those related to the processing of elements on (or nearest to) the road and they are among the most studied. This category includes pavement classification, marking detection, surface damage detection, drainage analysis, potholes search and road edge delimitation (Soilán et al. 2017; De Blasiis et al. 2020; Rastiveis et al. 2020; Zhong et al. 2020a). Also, traffic signs and traffic light, are occlusion-free given they must have direct visibility from vehicles to ensure road safety (Huang et al. 2017; Soilán et al. 2018).
- Semi occlusion-affected applications focus on the processing of urban objects whose location depends on the configuration of the urban street but between road traffic and pedestrian traffic. This category includes the processing of objects in the first visibility line from MLS (such as parked vehicles) as well as in the second line (furniture and vegetation) (Serna and Marcotegui 2014; Kunyuan et al. 2015; Yang et al. 2015; Li et al. 2019). Vehicles parked in a row are acquired laterally according to their bilateral symmetry, while vehicles parked in a battery are only front or rear partially acquired. Parked cars also occlude road edges (Serna and Marcotegui 2013). The furniture identification is affected by the existence of parked vehicles in front of them. If parked vehicles are present, the objects are partially or fully occluded in their lower part, while the upper part is not affected by occlusions (Balado et al. 2021).
- Occlusion affected applications process elements behind the object zone. Sidewalks and curbs are used to model and analyse pedestrian mobility and accessibility (Balado et al. 2018, 2019). The existence of occlusions implies breaks in the connectivity of the models and distortion of accessibility. Façades are used to digitise and model buildings in detail (Li et al. 2017; Xia and Wang 2019). Occlusions caused by cars or trees lead to missing façade data, breaking topological coherence of models and causing undesired behaviour in algorithms.

In view of the numerous applications affected by occlusions, whether on the ground, objects, or façades, it is important to quantify, locate, and analyse the causes of occlusions. In this way, it is possible to know in advance the occluded surface associated to urban objects, as well as to design efficient urban mapping plan with MLS, saving a considerable amount of time and resources.

The aim of this work is to detect and characterise occlusions caused by urban objects in street point clouds acquired with MLS. The proposed method takes advantage of the main cause of occlusions: the existence of an object (occluder) located between the MLS and the target. Therefore, to automatically analyse the visibility, synthetic points are generated to fill the holes of the point cloud in ground and façades. The visibility analysis search for collisions between the synthetic points and the MLS trajectory. The occlusion characterisation is performed from the class label of each point in the cloud. For this purpose, semantic segmentation of the point cloud is previously applied by means of Deep Learning. In case of collision with an urban object, the label of the occluder is identified and assigned to the occlusion. If there is no collision, the cause of the object is considered to be the geometry of the built environment or a reflective surface (e.g. windows). As a result, a new point cloud is generated that completes the occluded surface with information about the label of each object that obstructed the acquisition.

Related work

Obtaining complete point clouds has been a long-studied problem. Mainly two approaches can be distinguished here. The first one used in large point clouds is by performing multi-temporal acquisitions, relying on dynamic objects changing positions and minimizing occlusions. The second is through the completion of point clouds with synthetic data. This approach is mainly applied to small point clouds of partially self-occluded objects.

Multi-temporal approaches

Multi-temporal acquisitions are often used to detect and eliminate occlusions (Kukko et al. 2012; Qin and Gruen 2014; Xiao et al. 2015). However, multiple acquisitions multiply the acquisition cost and even so, multiple acquisitions do not guarantee the disappearance of occlusions. Dynamic occlusions (occlusions caused by dynamic objects) are minimized with each acquisition, while static occlusions are maintained (Chen and Yang 2016).

Acquiring data from a different perspective can overcome the visibility limitation of the MLS in areas occluded by static objects (Wang et al. 2019). Static terrestrial (TLS) and handled (HMLS) laser scanning devices allow for greater mobility in areas not accessible by MLS. However, acquisition with TLS and HMLS is considerably slower (Bauwens et al. 2016). UAVs allow data acquisition from aerial perspective (Mohd Noor et al. 2020). Such a perspective is very useful for roof acquisition. The main limitation of UAVs is their short flying autonomy and the legislative restrictions for their use in urban areas.

Completion approaches

Another alternative to complete point clouds is to try to correct occlusions by generating synthetic data. In image processing, image inpainting consists of regenerating the image background when an object is removed from the image (Buyssens et al. 2015). The first approaches to point cloud completion were based on the geometric characteristics of objects, such as symmetry and extrusion parameters (Kroemer et al. 2012). The main application of these first works was filling holes (Salamanca et al. 2008; Doria and Radke 2012). The leap from 2D to 3D involved a correct redimension of the space. Voxels were used because of their similarity to the 3D image structure, but voxels generate a large

amount of empty data and are not the final output of the process, so obtaining the final point cloud has a consequent loss of resolution. The first point based neural network (PointNet) allowed that point cloud features could be extracted with invariance to permutation and robustness to perturbation (Charles et al. 2017). Point Completion Network was the first network for shape completion to generate point clouds of more than 2048 points (Yuan et al. 2018). Since then, many network architectures were designed for the same purpose based on different paradigms: Cascaded Refinement (Wang et al. 2020), Point Fractal (Huang et al. 2020), Geometry-Aware Transformers (Yu et al. 2021), Adversarial Rendering (Xie et al. 2021), Point Moving Paths (Wen et al. 2021), View-Guided (Zhang et al. 2021), Recurrent Forward (Huang et al. 2021)

Currently, the completion approaches have some problems. First, the high computational cost required by the neural network, which can be seen in the small size of the objects in the datasets, which are far away in number of points from an urban environment acquired with an MLS. For this reason, some authors have opted to rasterize the point cloud of building façades in order to apply 2D inpainting and then reconstruct the 3D point cloud again (Chen et al. 2020). Another problem is the veracity of the synthetic points generated, as there is no guarantee that these points correspond to reality.

Contribution

Given the difficult and expensive solutions to remove occlusions, before making new acquisitions, it is recommended to know what percentage of the scene was acquired, where the occlusions are located, and which objects cause them. However, this knowledge is relegated to the expertise of the acquisition technician, since, to the best of the authors' knowledge, no work has been found that counts occlusions and identifies their cause except for (Balado et al. 2020), which focuses exclusively on urban ground with a non-georeferenced image output. The new contributions with respect to the previous work are:

- No need to use multi-temporal and multiple device data to locate occlusions.
- No need for training datasets to identify occlusions in the built environment (ground and façades).
- Direct application to entire streets, not only to isolated self-occluded objects.
- Runnable application on personal computers without resorting to the use of external servers or expensive graphics cards.
- 3D occluded surface detection and visualisation.
- Use of visibility for occlusion detection, instead of hit-or-miss transform, so no spatial adjacency between the occluding object and the occlusion is needed.

Method

The proposed method is composed of a sequence of processes shown in Figure 2. The input data is a point cloud $P = (P_X, P_Y, P_Z, P_L) \in \mathbb{R}^3$ of a street with one line of façades and segmented into ground, façades and objects $P = (G \cup F \cup C)$. In turn, the point cloud of objects must be segmented according to the occlusions to be characterized. There are numerous methods to segment an urban point cloud and obtain the input data for the proposed method. The current trend is to use Deep Learning (DL) for point cloud semantic segmentation. DL approaches are based on new network architectures that achieve high success rates in urban environments, such as Superpoint Graphs (Landrieu

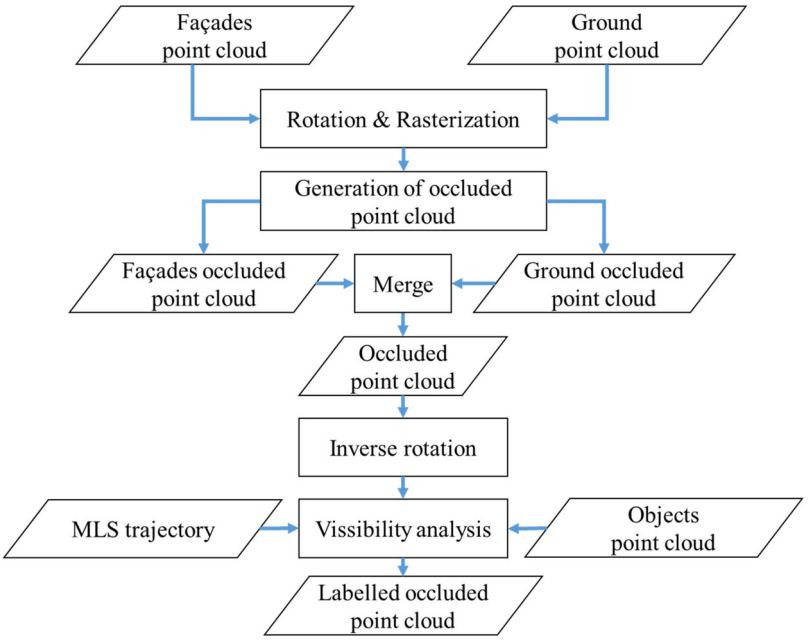


Figure 2. Workflow of the method.

and Simonovsky 2018) that achieved an Overall Accuracy of 94.0% in the Semantic3D dataset or the GRanD-Net (Ku et al. 2020) with 97.8% in the 3D Street dataset. In general, most networks can be applied to urban scenes with good results (even if such networks were not specifically designed for that purpose): DeePr3SS (Lawin et al. 2017), SnapNet (Boulch et al. 2017), SegCloud (Tchapmi et al. 2017), Dense PointNet++ (Lian et al. 2019), 3 P-RNN (Ye et al. 2018), or MS-RRFSegNet (Luo et al. 2020a), among others.

The method starts with a rotation of the point cloud to align it with the coordinate axes and the rasterization of façades and ground point clouds. On the raster images, the corresponding occlusions in façades and ground are detected as images, which are then converted to point clouds. The occluded point clouds of ground and façades are merged and rotated inversely to recover their original position. Finally, the object caused each point occlusion is identified by a visibility analysis between the MLS trajectory, the point cloud of the objects and the occluded point cloud.

Rotation

The purpose of the rotation is to align the point clouds of the façades F and the ground G to the coordinate axes for subsequent rasterization. The ground can be assumed as horizontal surface, or with a negligible tilt, and can be projected on the Z-plane directly. However, the line of façades can be oriented in any direction, so it is necessary to reorient the façades so that the maximum extent coincides with the Y-plane.

Since reorienting the façades is the cause of the rotation, the façades point cloud $F = (F_X, F_Y, F_Z) \in \mathbb{R}^3$ is used to obtain the orientation of façades on the Z-plane (assuming verticality and omitting the third component F_Z) by applying Principal Component Analysis calculated with the covariance (Eq. (1)). Then the rotation angle α is obtained by applying Eq. (2). The rotation matrix R is generated for Z-axis rotation (Eq. (3)). The

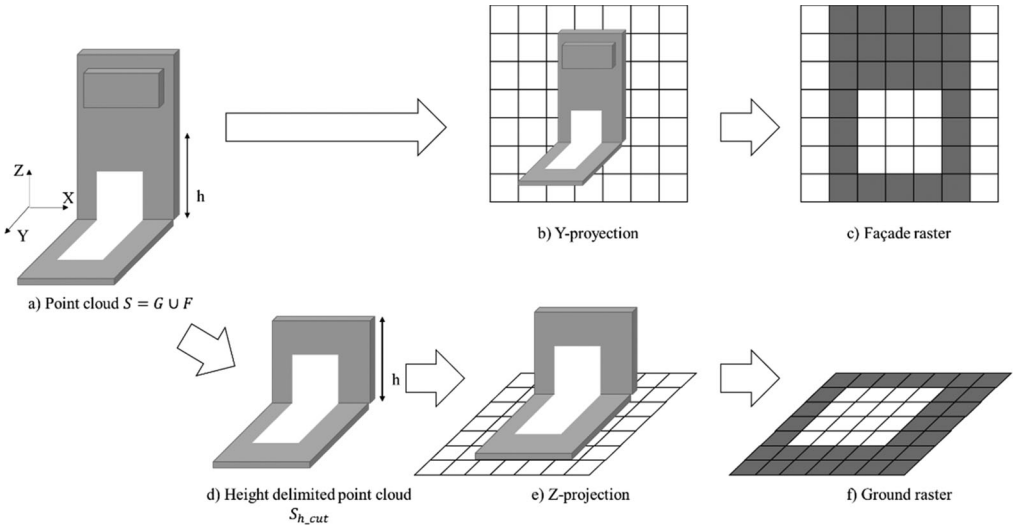


Figure 3. Rasterization in two directions.

rotation matrix is applied to façades point cloud and ground point cloud to obtain the rotated point clouds $F_{rot} = FR$ and $G_{rot} = GR$.

$$cov(\overline{F_{XY}}) = \begin{bmatrix} c_{11} & c_{12} \\ c_{21} & c_{22} \end{bmatrix} = \frac{\overline{F_{XY}}\ \overline{F_{XY}}^{T}}{n-1} \tag{1}$$

Where $\overline{F_{XY}} = F_{XY} - \ mean(F_{XY})$

$$\alpha = atan\left({}^{c_{12}}/_{c_{11}}\right) \tag{2}$$

$$R = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

Rasterization

Rasterization is the process of converting the point cloud into a 2D image (Balado et al. 2017). It is used for the subsequent detection of occlusions. The point clouds for rasterization are those of the ground G and façades F simultaneously $S = G \cup F$, as their combined use allows the delimitation of occlusions in the built environment. In addition, for the detection of occlusions in the ground, the point cloud S is delimited to a height h from the minimum G_z, because the elements protruding from the façades (such as balconies and ornamentation) hinders the projection of occlusions. In Figure 3, an overview of the projections for the rasterization is shown.

Since the rasterization process is identical for both projections, only changing the direction of the projection and the input point clouds, the following is an explanation of the rasterization process on the Z-plane with the height delimited point cloud $S_{h_{cut}}$ of façades and ground. The other rasterization is performed on the Y axis and the point cloud S without height delimitation. First, the resolution of the 2D grid (pixel size r) is selected. The value of r must be low because high values produce pixilation, so the geometric shapes in vector data are adjusted to a grid losing precision. But value of r also must be sufficient so that empty pixels are not generated between consecutive points and occlusions are not enclosed. Therefore, r

depends on the point density of the point cloud. Second, the maximum and minimum edges of the point cloud $S_{h_{cut}}$ on the Z plane. The 2D bounding box of $S_{h_{cut}XY}$ coordinates are calculated and divided into a grid with a step r in X and Y directions. Third, the ratio of the projected points in each pixel is calculated, according to their X and Y coordinates. And fourth, a binary value is assigned to the pixel according to Eq. (4). The result is the generation of a binary image pixels with value 1 if the pixel contains projected points and 0 otherwise. Together with the image, an array of pixel-point correspondences is generated where the id of the points contained into to each pixel is saved.

$$pixel_{value}i = \begin{cases} 1 \ if \ \exists \ S_{XY} \ within \ pixel_{border}i \\ 0 \ if \ \nexists \ S_{XY} within \ pixel_{border}i \end{cases} \tag{4}$$

Generation of occluded point cloud

The generation of the occluded point cloud consists of several steps. First, the occlusion is detected in the raster image I. In the generated raster images, occlusions are identified as holes (pixels with value 0) surrounded by pixels with points (pixels with value 1). To eliminate salt-and-pepper noise due to density variations and false reflections, a 2×2 median filter is applied. Then a gap-filling algorithm is applied to the binary raster image to fill the holes. To identify the pixels corresponding to the occlusions, the filtered image is subtracted from the filled image.

Once the occlusion pixels have been detected, new points are generated in the center of each occlusion pixels to obtain the occluded point cloud. The coordinates of the points are obtained from the array of pixel-point correspondences stored in the rasterization process. Only two coordinates can be extracted from the raster image I for the new occluded point cloud, the point cloud F (if the cloud of façades occlusions is being generated) or G (if the cloud of ground occlusions is being generated) is used to find the other component of the occluded points (Eqs. (5) and (6)). For each new point, a 2D nearest neighbor search with the *knn* algorithm with k = 1 is applied to the cloud G or F as appropriate. The missing component of the nearest point of G or B is assigned to the new point of the occlusion. The result of these operations is the generation of dots on the empty surfaces left by the occlusions (Figure 4). In addition, since the occluded point cloud was generated from the rotated ground and façades clouds, an inverse rotation R^{-1} is applied to position it correctly.$O_{Building} = \left(O_{X_{from_I}}, O_{Y_{from_F}}, O_{Z_{from_I}}\right)(5) where \ O_{Y_{from}F} \leftarrow F_Y : \left(dist|O_{F_XZ}, F_{XZ}|, \geq\right)$

$$O_{Ground} = \left(O_{X_{from}I}, O_{Y_{from}I}, O_{Z_{from}G}\right) \tag{6}$$

where $O_{Z_{from}G} \leftarrow G_Z : \left(dist|O_{G_XY}, G_{XY}|, \geq\right)$

Visibility analysis

Finally, the occlusion is characterized by assigning to each point of the occluded point cloud O the identifier of the object class that caused the occlusion. This operation is performed by a visibility analysis between each point O_i of the occluded point cloud and the trajectory $T = (T_X, T_Y, T_Z) \in \mathbb{R}^3$, looking for collisions in the object cloud $C = (C_X, C_Y, C_Z, C_L) \in \mathbb{R}^3$ (Figure 5). For each point Oi, the nearest point of the trajectory T is searched by the *knn* search algorithm with k = 1. Then, both points are joined by a straight line *l*, such that l is perpendicular to T and contains O_i. The line l is divided into d spaced points and the distance from the line of points to the object point cloud C is

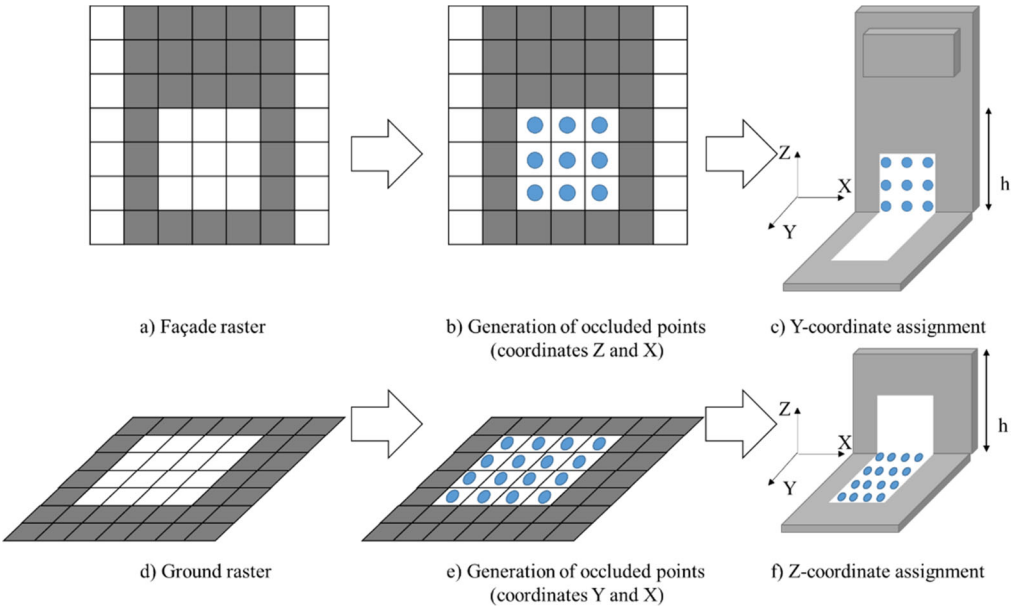


Figure 4. Generation of occluded point clouds for façades and ground.

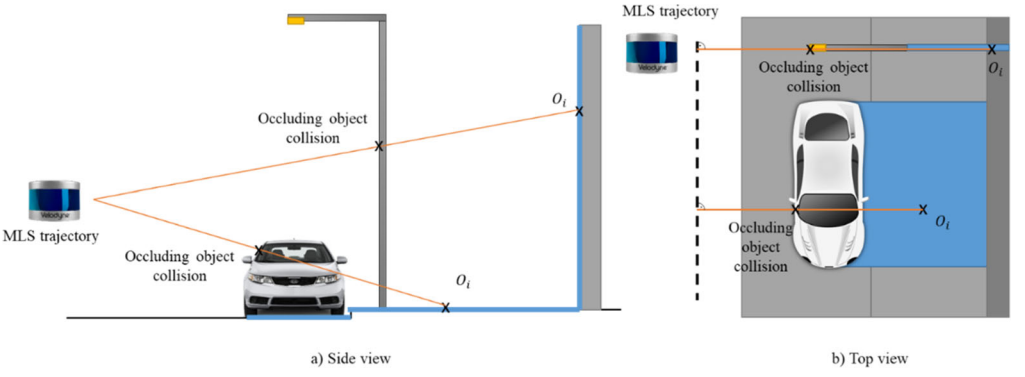


Figure 5. Visibility analysis between each point O_i of the occluded point cloud and the MLS trajectory with side view (a) and top view (b). Occluded point cloud is coloured in blue.

calculated. If any point is less than d from the points on the line l, it is considered as a collision, and its label C_l is assigned to the occluded point Oi. The visibility analysis is repeated for each point of the occluded point cloud. If there are no object points close (within a distance d) to the line, it is considered that the beam does not collide with an object and the occlusion is due to other causes, such as the geometry of the built environment or entry spaces (doors and windows).

Experiments

Case study

The proposed method was tested on five point clouds from TerraMobilita/iQmulus dataset in the city of Paris (France) (Vallet et al. 2015) acquired with the MLS Stereopolis II.

Table 1. Information of the case studies.

Case study	Number of points	Ground area (m^2)	Façade area (m^2)	Objects
1	1.8 M	730	1200	Bus, cars, pedestrians
2	0.5 M	330	670	Bikes, pedestrians, bollards, poles
3	0.7 M	530	950	Cars, motorbikes, poles
4	1.5 M	1200	2200	Cars, pedestrians, poles
5	1.0 M	480	1500	Cars, bikes, motorbikes, poles
TOTAL	5.5 M	3270	6520	Bus, cars, pedestrians, bollards, poles, bikes, motorbikes

Table 1 collects information in terms of number of points per case study, area tested and object classes. The density of each cloud varies from 1000 points/m^2 on the ground closest to the MLS trajectory to 300 points/m^2 at the top of the façades. The trajectory was generated following the indications of (Zhong et al. 2020b). The ground truth of the occlusions was generated manually by polygonising all existing holes in ground and façades and contrasting the object generating each occlusion.

The parameters for the automatic method execution were as follows. The height value h to delimit the façades and not overlap ground occlusions was $h = 4$ m, except for case study 1, which due to the bus, was set at $h = 7$ m to allow the closure of the occlusion. The pixel value r was set to 10 cm × 10 cm. This value ensures pixel allocation, both for the ground (where the distance between points is approximately 2 cm for consecutively acquired points and a distance between scanlines of 5 cm) and for the façade, where the distance between points is approximately 7 cm. The same criterion was used for the selection of $d = 10$ cm in the visibility analysis.

Results and analysis

Figure 6 shows the case studies before and after applied the proposed method. Tables 2–6 recompile the detection of the occluded area per object class and case study, as well as the accuracy of the proposed method. Table 7 summarizes the statistics of all case studies. The overall detection obtained high accuracy of 98.71% for ground occlusions and 83.37% for façades. The identification of the occluded area with the occluding class was perfect for all detected occlusions and in all case studies.

Errors in the detection of occlusions caused by objects are in isolated situations. The undetected occlusions at the junction of ground and façades (Figure 7a) on the ground were due to the adjustment of parameter h in the rasterization process to preserve of some protrusions or balconies. These elements were projected over the ground and overlapped some occlusions, which were therefore undetected. To a lesser extent, there are also very narrow occlusions (Figure 7b). In the rasterization process, accuracy was lost due to the adjustment of a vector data to a mesh. The undetected narrow occlusions have a maximum width of pixel size $r = 10$ cm. However, given the size of these occlusions, they can be considered negligible compared to those generated by larger objects. On the façades, all errors were centered on the upper part of the buildings. Due to the low density and visibility of the MLS, large areas were left open and the gap filling algorithm could not be applied correctly (Figure 7c). The origin of these empty spaces was due to the geometry of the building, and not to street furniture.

The behaviour of occlusions not generated by objects (Class None) was drastically opposite in ground and façades. On the ground, practically all occlusions were caused by objects. Only 0.5% of all ground occlusions were due to geometry of the ground, drainage inclination or potholes. On the façades, 82.7% of the occlusions were due to the balconies,

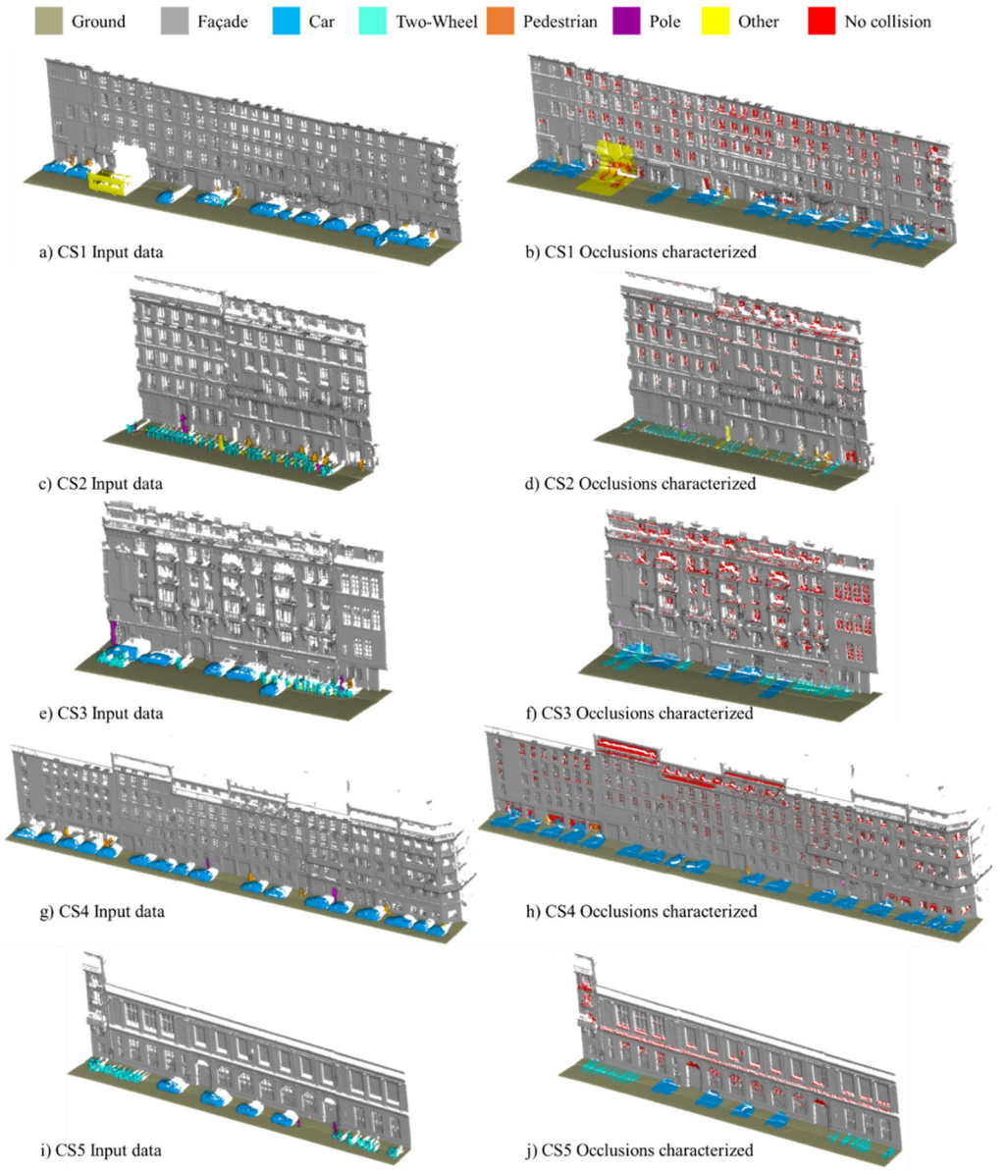


Figure 6. Input point clouds (left) of the case studies and results (right) with detected and characterized occlusions.

windows and decorative elements. Except for pole-like objects and bus, all occlusions produced by the collision of the LiDAR beam with an object were concentrated in the lower part of the façades. Table 8 compiles the occluded area in relation to the number of objects and classes.

In all tables, the variation of the occluded area according to the class of objects can be observed. Most of the occluded ground area corresponds to the 37 cars, followed by far by the area occluded by two-wheel vehicles, with 62 objects counted between motorcycles and bicycles. Although he estimated floor area occluded by cars was 15.36 m2, each two-wheel vehicle only reached 3.62 m2. Therefore, cars are the objects that cause the greatest amount of occlusion in number and size. Occlusions caused by pole and pedestrians were

Table 2. Occluded surface for case study 1.

	Ground			Façades		
Class	Detected (m^2)	Missed (m^2)	Accuracy	Detected (m^2)	Missed (m^2)	Accuracy
Car	121.66	0.75	99.39%	20.04	0	100%
Two-wheels	1.39	0	100%	0.49	0	100%
Pedestrian	0.12	0	100%	1.48	0	100%
Pole-like	0	0	100%	0	0	–
Others	36.96	0	100%	29.28	0	100%
None	3.89	0	100%	50.42	0	100%

Table 3. Occluded surface for case study 2.

	Ground			Façades		
Class	Detected (m^2)	Missed (m^2)	Accuracy	Detected (m^2)	Missed (m^2)	Accuracy
Car	–	–	–	–	–	–
Two-wheels	18.06	1.08	94.36%	0.7	0	100%
Pedestrian	0.98	0.27	78.40%	2.32	0	100%
Pole-like	0.14	0.12	53.85%	0.23	0	100%
Others	2.48	0.93	72.73%	0.68	0	100%
None	0.12	0	100%	31.43	10.88	74.29%

Table 4. Occluded surface for case study 3.

	Ground			Façades		
Class	Detected (m^2)	Missed (m^2)	Accuracy	Detected (m^2)	Missed (m^2)	Accuracy
Car	50.54	3.45	93.61%	7.16	0	100%
Two-wheels	45.03	2.75	94.24%	4.65	0	100%
Pedestrian	–	–	–	0.11	0	100%
Pole-like	0.09	0	100%	0.36	0	100%
Others	0.97	0	100%	0.12	0	100%
None	–	–	–	66.12	5.97	91.72%

Table 5. Occluded surface for case study 4.

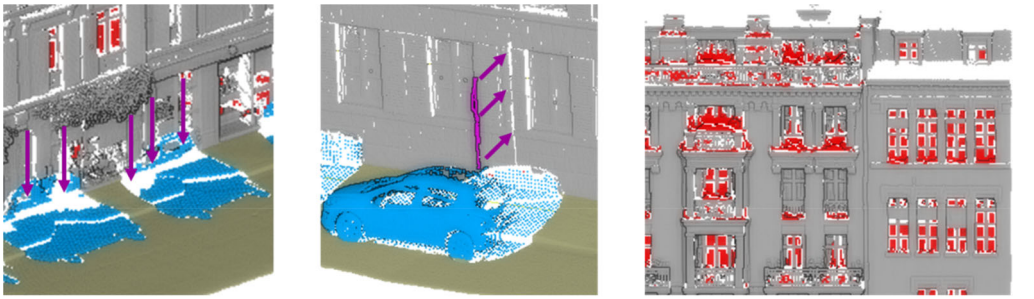
	Ground			Façades		
Class	Detected (m^2)	Missed (m^2)	Accuracy	Detected (m^2)	Missed (m^2)	Accuracy
Car	180.69	1.54	99.15%	26.23	0	100%
Two-wheels	–	–	–	–	–	–
Pedestrian	0.29	0	100%	5.81	0	100%
Pole-like	–	–	–	0.32	0.05	86.49%
Others	–	–	–	–	–	–
None	–	–	–	216.26	50	81.22%

Table 6. Occluded surface for case study 5.

	Ground			Façades		
Class	Detected (m^2)	Missed (m^2)	Accuracy	Detected (m^2)	Missed (m^2)	Accuracy
Car	215.29	0	100%	4.72	0	100%
Two-wheels	160.03	0	100%	0.65	0	100%
Pedestrian	–	–	–	–	–	–
Pole-like	0.59	0.05	92.19%	0.05	0.1	33.33%
Others	–	–	–	–	–	–
None	0.8	0	100%	39.62	34.58	53.40%

Table 7. Overall occluded surface for all case studies.

	Ground			Façades		
Class	Detected (m^2)	Missed (m^2)	Accuracy	Detected (m^2)	Missed (m^2)	Accuracy
Car	568.18	5.74	99.00%	58.15	0	100%
Two-wheels	224.51	3.83	98.32%	6.49	0	100%
Pedestrian	1.39	0.27	83.73%	9.72	0	100%
Pole-like	0.82	0.17	82.83%	0.96	0.15	86.49%
Others	40.41	0.93	97.75%	30.08	0	100%
None	4.81	0	100%	403.85	101.43	79.93%
All	**840.12**	**10.94**	**98.71%**	**509.25**	**101.58**	**83.37%**



a) Undetected occlusion due to balconies

b) Undetected occlusion due narrow object

c) Undetected occlusion due error in border delimitation

Figure 7. Occlusion detection errors due to preservation of balconies (a), narrow objects (b) and openings in façades (c).

the lowest (less than 0.1 m^2 both). The class 'others' include very diverse categories. Although all 'other' objects were compiled in Table 8 in the same line, the bus alone produced an occlusion of 36.96 m^2. Considering this, the remaining 21 objects in the other class (mostly bollards) produced a total of 4.02 m^2, approximately 0.2 m^2 per object.

In façades, occlusions caused by objects are less numerous. Car occlusions, for example, only occupy 10% of the area they occupy on the ground. The only object that maintains the size of the occlusion is the poles, in addition to the bus. The occlusions produced by pedestrians increase in the façades with respect to the ground (0.51 m^2 per pedestrian versus only 0.07 m^2). This is due to the position of pedestrians closer to the façades than other objects since the pedestrian transit zone is immediately in front of the façades.

The area occluded by each object depends on the position of the object on the street If the object is close to the façade, the occlusion on the façade will be higher and, on the ground, lower. On the other hand, if the object is far from the façade, the occlusion will occur exclusively on the ground. All objects generate occlusions on the ground, regardless of their proximity to buildings. Also, objects acquired in motion showed occlusions with shapes and sizes different from their static equivalent (Figure 8).

Discussion

From the presented method it is possible to obtain information that allows to evaluate the feasibility of the case study according to the occluded surface. Although in this paper, the classes of study were three, if the point cloud is segmented at a higher level of detail (for example, the ground can be divided into road, sidewalk, and curb), it is possible to know

Table 8. Average area occluded per object.

	Case study						Ground		Façades	
Class	1	2	3	4	5	All	Occluded detected area (m^2)	Area per object (m^2)	Occluded detected area (m^2)	Area per object (m^2)
Car	11	0	5	17	4	37	568.18	15.36	58.15	1.57
Two-wheels	1	26	16	0	19	62	224.51	3.62	6.49	0.10
Pedestrian	6	5	1	7	0	19	1.39	0.07	9.72	0.51
Pole-like	0	3	2	2	2	9	0.82	0.09	0.96	0.11
Others	1	14	6	1	0	22	40.41	1.84	30.08	1.37

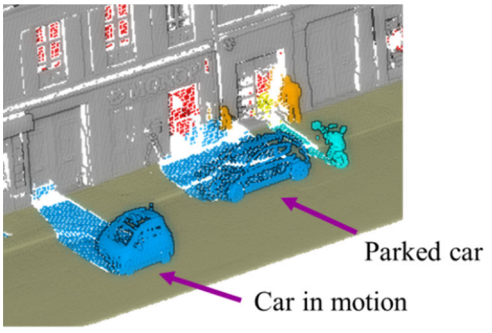


Figure 8. Comparison of occlusion produced by a car in motion and parked cars.

the most affected subclass, or its connectivity with the rest of the environment due to the occlusions caused by impacts. In addition, since each object implies an occluded area, as shown above by calculating the average per class, it is possible to approximate the occluded area if object class and amount are known beforehand.

From the results obtained from the method it is also possible to evaluate whether new multitemporal acquisitions would reduce occlusions. Since most of the occlusions were produced by parked cars, parked motorcycles and the geometry of the environment, a new acquisition would only reduce the occlusion caused by pedestrians, moving vehicles (in ground and façades of our case studies about 95 m^2 including pedestrians, one car in motion and the bus). Empty parking spaces would also decrease the occlusions, although it is difficult to know their exact quantity in advance the acquisition. If parking is prohibited in streets during the acquisition, the occlusions of parked vehicles would be reduced (in our case studies about 935 m^2 considering ground and façades). Only poles, furniture, and geometries of the environment would continue to produce non-removable occlusions.

The operating principle of the method is based on the problem of occlusions: the lack of direct visibility between LiDAR and occluded surface. Although the characterization is direct for one occluding object, it can happen that multiple objects occlude the same area (Figure 9). The proposed method considered the object closest to the MLS as the one causing the occlusion, so the first object occludes the subsequent one, therefore the line of visibility would not collide with points of the object behind. However, the human-knowledge of the problem and of the urban environment indicate us that this occlusion has multiple causes not included in the proposed method.

It is important to mention that the proposed method is not a reconstruction of the environment. The objective is to estimate the occluded area and its cause; therefore, the precision and the realism of the generated occluded point clouds are not evaluated. The generated points did not always coincide in height with the rest of the ground (Figure 10a and 10b) or in depth with the façade (Figure 10c) and small sporadic

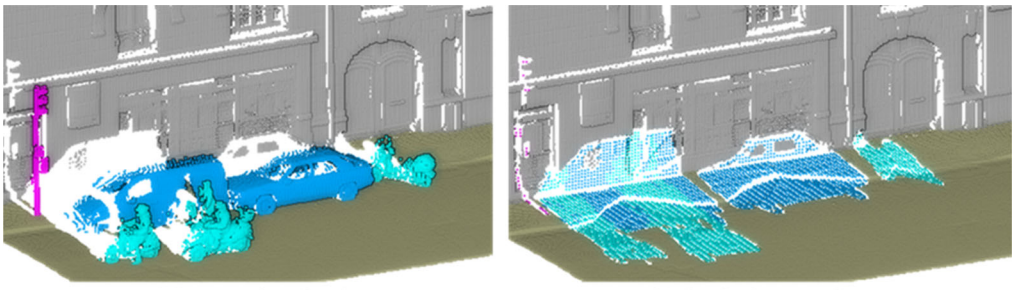


Figure 9. Overlay of objects (a) in the MLS visibility and proposed characterization of the occluded area (b).

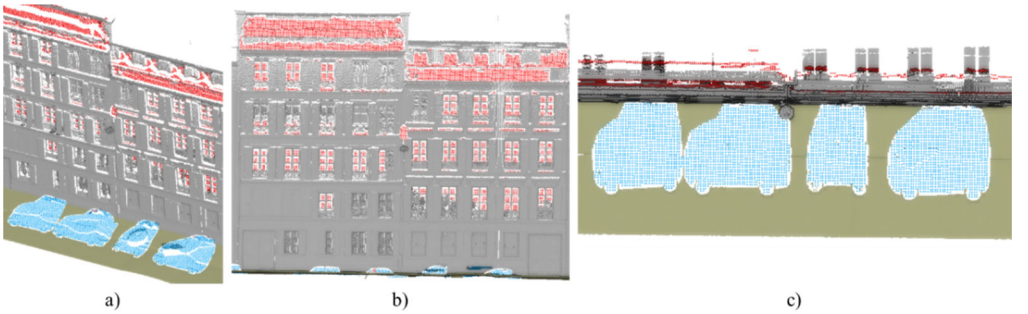


Figure 10. Misalignments in the height and depth assignment of the ground (blue) and façade (red) occluded points in perspective (a), frontal (b) and top (c) views.

malformations can be observed. However, the precision of the method was robust for the analysis of visibility between occluded surface and MLS. Moreover, the generated points were correctly distributed regularly, as can be seen in Figure 10(b) and 10(c), allowing estimate the occluded area.

A clear limitation of the method presented in this work is the condition of perpendicularity between the trajectory and the beam of the MLS to the occlusion. Although for the MLS used in the TerraMobilita/iQmulus dataset, the method showed a perfect characterization of occlusions, there are also obliquely LiDAR setups that would not be suitable for the application of this method. For this reason, in future work it is contemplated to select the angles of the visibility analysis considering the mounting of the LiDAR heads in the vehicle. Another relevant feature of the LiDAR setup is the height. LiDAR in MLS is often installed on the roof of vans. Setups at a lower height would increase the size of the occlusions in ground and façades since the lower viewpoint would increase the shadow from parked vehicles and urban objects. Whereas higher setups, close to a top view, would minimize the occlusions generated by smaller objects, although the same amount of occluded area would be preserved in case of pole like objects. The proposed method allows the study of occlusions for different MLS heights.

Finally, urban areas are extremely complex environments. This work focused on an urban geometry of the street type enclosed by buildings near to and parallel to the road. This allows rasterization to obtain frontal images of façades. However, in organic urban distributions, it would not be possible to detect occlusions on facades, due to their non-parallelism with the street. It is also possible that there are no buildings, such as in green areas, so the detection of occlusions on façades would not make sense. Furthermore, in this work, façades are used to delimit the extent of the urban ground. The non-existence

of façades could lead to incorrect detection of ground occlusions. This can be related to the point density at the cloud border. Since the pixel size in the rasterization process is highly point density-dependent, in the case of façades close to the MLS, density values are bounded. However, in open urban areas, the point density decays gradually, so that limiting a density value to fix the pixel size r will necessarily incur errors in the detection of the case study borders. Another urban element with a characteristic generation of occlusions is vegetation. In particular, treetops and bushes produce LiDAR multi-return, so the occlusions are less well defined, generating noisy occlusions of varying point density due to the last return reached on the façade. To evaluate all these possibilities of the urban environment, more complex datasets will be evaluated in future work (Tan et al. 2020; Zhu et al. 2020).

Conclusions

This paper presents an automatic method for the detection, characterization, and accounting of occluded areas in MLS-acquired street point clouds. The proposed method is based on the generation of point clouds corresponding to occluded areas with a regular point distribution from the rasterization of the input point cloud and gap filling. Visibility analysis between the occluded cloud and the MLS trajectory is used to obtain the collisions with the objects that cause occlusions. The method was tested with five point clouds from the TerraMobilita/iQmulus dataset, and it requires point clouds classified as input data.

The method detected the 98.71% of the occlusions on ground and 83.37% on façades. All detected occlusions were correctly characterized with the class of the object that caused each occlusion. From the results of the case studies, it was concluded that cars are the objects that cause the largest amount of occluded area due to their number and size. Each car occluded an average area 15.36 m^2 of ground, but only 1.57 m^2 of façades. The distribution of occlusion between façade and ground depends on the location of the object on the street. The proposed method allows to optimize multi-temporal acquisitions because the occlusions are classified according to the objects that caused, so static or dynamic behavior of the occluding object is known. In addition, the identification of the occluded area per object allows approximate the global occluded are only with information of the number of objects and the class.

Future work will focus on adapting the method for operation with scan angles not perpendicular to the trajectory, as well as eliminating the trajectory as input. Another way of detecting occlusions in the point cloud will be implementing directly in the 3D point cloud due to the limitations presented by the rasterization process. In terms of implementation of DL techniques, such as generation adversarial networks, future work will focus on adapting such networks for application in urban environments, considering the needs in terms of processing/number of points. Also, the conclusions of this work (occlusion size per object class) can be used to generate synthetic occlusions and train neural networks.

Acknowledgments

This research was funded by the Xunta de Galicia, grant number ED481B-2019-061 and ED431C 2020/01; and by the Ministerio de Ciencia, Innovación y Universidades—Gobierno de España-, grant number (PID2019-105221RB-C43/AEI/10.13039/501100011033). This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 769255. This document reflects only the views of the author(s). Neither the Innovation and Networks Executive Agency (INEA) or the European Commission is in any way responsible for any use that may be made of the information it contains. The statements made herein are solely the responsibility of the authors.

Disclosure statement

No potential conflict of interest was reported by the authors.

ORCID

Jesús Balado http://orcid.org/0000-0002-3758-3102
Pedro Arias http://orcid.org/0000-0002-3547-8907
Silvia María González-Collazo http://orcid.org/0000-0002-0381-0856

References

Balado J, Arias P, Lorenzo H, Meijide-Rodríguez A. 2021. Disturbance analysis in the classification of objects obtained from urban LiDAR point clouds with convolutional neural networks. Remote Sens. 13(11):2135.

Balado J, Díaz-Vilariño L, Arias P, González-Jorge H. 2018. Automatic classification of urban ground elements from mobile laser scanning data. Autom Constr. 86:226–239.

Balado J, Díaz-Vilariño L, Arias P, Lorenzo H. 2019. Point clouds for direct pedestrian pathfinding in urban environments. ISPRS J Photogramm Remote Sens. 148:184–196.

Balado J, Díaz-Vilariño L, Arias P, Soilán M. 2017. Automatic building accessibility diagnosis from point clouds. Autom Constr. 82:103–111.

Balado J, González E, Verbree E, Díaz-Vilariño L, Lorenzo H, Balado J. 2020. Automatic detection and characterization of ground occlusions in urban point clouds from Mobile Laser Scanning data. ISPRS Ann Photogramm Remote Sens Spat Inf Sci. 6.

Bauwens S, Bartholomeus H, Calders K, Lejeune P. 2016. Forest inventory with terrestrial LiDAR: a comparison of static and hand-held mobile laser scanning. Forests. 7(6):127.

Boulch A, Saux BL, Audebert N. 2017. Unstructured point cloud semantic labeling using deep segmentation networks. Proceedings of the Eurographics Workshop on 3D Object Retrieval [Internet]. Goslar Germany, Germany: Eurographics Association. p. 17–24..

Buyssens P, Daisy M, Tschumperle D, Lezoray O. 2015. Exemplar-based inpainting: technical review and new heuristics for better geometric reconstructions. IEEE Trans Image Process. 24:1809–1824.

Charles RQ, Su H, Kaichun M, Guibas LJ. 2017. PointNet: deep learning on point sets for 3D classification and segmentation. 2017 IEEE Conference on Computer Vision and Pattern Recognition. p. 77–85.

Che E, Jung J, Olsen MJ. 2019. Object recognition, segmentation, and classification of mobile laser scanning point clouds: a state of the art review. Sensors. 19(4):810. https://www.mdpi.com/1424-8220/19/4/810.

Chen C, Yang B. 2016. Dynamic occlusion detection and inpainting of in situ captured terrestrial laser scanning point clouds sequence. ISPRS J Photogramm Remote Sens. 119:90–107. http://www.sciencedirect.com/science/article/pii/S0924271616300922.

Chen J, Yi JS, Kahoush M, Cho ES, Cho YK. 2020. Point cloud scene completion of obstructed building facades with generative adversarial inpainting. Sensors. 20(18):5029.

De Blasiis MR, Di Benedetto A, Fiani M. 2020. Mobile laser scanning data for the evaluation of pavement surface distress. Remote Sens. 12(6):942. https://www.mdpi.com/2072-4292/12/6/942.

Doria D, Radke RJ. 2012. Filling large holes in LiDAR data by inpainting depth gradients. 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Work. p. 65–72.

Han X, Dong Z, Yang B. 2021. A point-based deep learning network for semantic segmentation of MLS point clouds. ISPRS J Photogramm Remote Sens. 175:199–214. https://www.sciencedirect.com/science/article/pii/S0924271621000642.

Huang P, Cheng M, Chen Y, Luo H, Wang C, Li J. 2017. Traffic sign occlusion detection using mobile laser scanning point clouds. IEEE Trans Intell Transport Syst. 18(9):2364–2376.

Huang Z, Yu Y, Xu J, Ni F, Le X. 2020. PF-Net: point fractal network for 3D point cloud completion. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Huang T, Zou H, Cui J, Yang X, Wang M, Zhao X, Zhang J, Yuan Y, Xu Y, Liu Y. 2021. RFNet: recurrent forward network for dense point cloud completion. Proceedings of the IEEE/CVF International Conference on Computer Vision. p. 12508–12517.

Kroemer O, Amor HB, Ewerton M, Peters J. 2012. Point cloud completion using extrusions. 2012 12th IEEE-RAS International Conference on Humanoid Robot (Humanoids 2012). p. 680–685.

Ku T, Veltkamp RC, Boom B, Duque-Arias D, Velasco-Forero S, Deschaud J-E, Goulette F, Marcotegui B, Ortega S, Trujillo A, et al. 2020. SHREC 2020: 3D point cloud semantic segmentation for street scenes. Comput Graph. 93:13–24. http://www.sciencedirect.com/science/article/pii/S0097849320301400.

Kukko A, Kaartinen H, Hyyppä J, Chen Y. 2012. Multiplatform mobile laser scanning: usability and performance. Sensors. 12(9):11712–11733.

Kunyuan C, Ming C, Menglan Z, Xinqu C, Yifei C, Jonathan L, Hongshan N. 2015. Automated object extraction from MLS data: a survey. 2015 10th International Conference on Intelligent Systems and Knowledge Engineering. p. 331–334.

Landrieu L, Simonovsky M. 2018. Large-scale point cloud semantic segmentation with superpoint graphs. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. p. 4558–4567.

Lawin FJ, Danelljan M, Tosteberg P, Bhat G, Khan FS, Felsberg M. 2017. Deep projective 3D semantic segmentation. In: Felsberg M, Heyden A, Krüger N, editors. Computer analysis of images and patterns. Cham: Springer International Publishing. p. 95–107.

Li F, Lehtomäki M, Oude Elberink S, Vosselman G, Kukko A, Puttonen E, Chen Y, Hyyppä J. 2019. Semantic segmentation of road furniture in mobile laser scanning data. ISPRS J Photogramm Remote Sens. 154:98–113. https://www.sciencedirect.com/science/article/pii/S092427161930142X.

Li Z, Liqiang Z, Mathiopoulos PT, Liu F, Liang Z, Li S, Liu H. 2017. A hierarchical methodology for urban facade parsing from TLS point clouds. ISPRS J Photogramm Remote Sens. 123:75–93. https://www.sciencedirect.com/science/article/pii/S0924271616305408.

Lian Y, Feng T, Zhou J. 2019. A dense Pointnet++ architecture for 3D point cloud semantic segmentation. IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium. p. 5061–5064.

Luo H, Chen C, Fang L, Khoshelham K, Shen G. 2020a. MS-RRFSegNet: multiscale regional relation feature segmentation network for semantic segmentation of urban scene point clouds. IEEE Trans Geosci Remote Sens. 58(12):8301–8315.

Luo H, Khoshelham K, Fang L, Chen C. 2020b. Unsupervised scene adaptation for semantic segmentation of urban mobile laser scanning point clouds. ISPRS J Photogramm Remote Sens. 169:253–267. https://www.sciencedirect.com/science/article/pii/S0924271620302744.

Mohd Noor N, Ibrahim I, Abdullah A, Abdullah AA. 2020. Information fusion for cultural heritage three-dimensional modeling of Malay cities. ISPRS Int J Geo-Information. 9(3).

Qin R, Gruen A. 2014. 3D change detection at street level using mobile laser scanning point clouds and terrestrial images. ISPRS J Photogramm Remote Sens. 90:23–35. https://www.sciencedirect.com/science/article/pii/S0924271614000100.

Rastiveis H, Shams A, Sarasua WA, Li J. 2020. Automated extraction of lane markings from mobile LiDAR point clouds based on fuzzy inference. ISPRS J Photogramm Remote Sens. 160:149–166. https://www.sciencedirect.com/science/article/pii/S0924271619302977.

Salamanca S, Merchán P, Hernández E, Adan A, Cerrada C. 2008. Filling holes in 3D meshes using image restoration algorithms. 4th International Symposium on 3D Data Processing, Visualization and Transmission, 3DPVT 2008.

Serna A, Marcotegui B. 2013. Urban accessibility diagnosis from mobile laser scanning data. ISPRS J Photogramm Remote Sens. 84:23–32.

Serna A, Marcotegui B. 2014. Detection, segmentation and classification of 3D urban objects using mathematical morphology and supervised learning. ISPRS J Photogramm Remote Sens. 93:243–255.

Soilán M, Riveiro B, Martínez-Sánchez J, Arias P. 2017. Segmentation and classification of road markings using MLS data. ISPRS J Photogramm Remote Sens. 123:94–103. http://linkinghub.elsevier.com/retrieve/pii/S0924271616303173.

Soilán M, Riveiro B, Sánchez-Rodríguez A, Arias P. 2018. Safety assessment on pedestrian crossing environments using MLS data. Accid Anal Prev. 111:328–337. http://www.sciencedirect.com/science/article/pii/S0001457517304475.

Sun Z, Xu Y, Hoegner L, Stilla U. 2018. Classification of MLS point clouds in urban scenes using detrended geometric features from supervoxel-based local contexts. ISPRS Ann Photogramm Remote Sens Spat Inf Sci. IV-2:271–278. https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/IV-2/271/2018/.

Tan W, Qin N, Ma L, Li Y, Du J, Cai G, Yang K, Li J. 2020. Toronto-3D: a large-scale mobile LiDAR dataset for semantic segmentation of urban roadways. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Work [Internet]. http://dx.doi.org/10.1109/CVPRW50498.2020.00109.

Tchapmi L, Choy C, Armeni I, Gwak J, Savarese S. 2017. SEGCloud: semantic segmentation of 3D point clouds. 2017 International Conference on 3D Vision. p. 537–547.

Vallet B, Brédif M, Serna A, Marcotegui B, Paparoditis N. 2015. TerraMobilita/iQmulus urban point cloud analysis benchmark. Comput Graph. 49:126–133. http://dx.doi.org/10.1016/j.cag.2015.03.004.

Wang X, Jr, Marcelo HA, Lee GH. 2020. Cascaded refinement network for point cloud completion. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Wang Y, Chen Q, Zhu Q, Liu L, Li C, Zheng D. 2019. A survey of mobile laser scanning applications and key techniques over urban areas. Remote Sens. 11(13):1543.

Wen X, Xiang P, Han Z, Cao Y-P, Wan P, Zheng W, Liu Y-S. 2021. PMP-Net: point cloud completion by learning multi-step point moving paths. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. p. 7443–7452.

Xia S, Wang R. 2019. Façade separation in ground-based LiDAR point clouds based on edges and Windows. IEEE J Sel Top Appl Earth Observ Remote Sens. 12(3):1041–1052.

Xiao W, Vallet B, Brédif M, Paparoditis N. 2015. Street environment change detection from mobile laser scanning point clouds. ISPRS J Photogramm Remote Sens. 107:38–49. http://www.sciencedirect.com/science/article/pii/S0924271615001215.

Xie C, Wang C, Zhang B, Yang H, Chen D, Wen F. 2021. Style-based point generator with adversarial rendering for point cloud completion. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. p. 4619–4628.

Yang B, Dong Z, Zhao G, Dai W. 2015. Hierarchical extraction of urban objects from mobile laser scanning data. ISPRS J Photogramm Remote Sens. 99:45–57. http://www.sciencedirect.com/science/article/pii/S092427161400255X.

Ye X, Li J, Huang H, Du L, Zhang X. 2018. 3D recurrent neural networks with context fusion for point cloud semantic segmentation. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y, editors. Computer vision – ECCV 2018. Cham: Springer International Publishing. p. 415–430.

Yu X, Rao Y, Wang Z, Liu Z, Lu J, Zhou J. 2021. PoinTr: diverse point cloud completion with geometry-aware transformers. Proceedings of the IEEE/CVF International Conference on Computer Vision. p. 12498–12507.

Yuan W, Khot T, Held D, Mertz C, Hebert M. 2018. PCN: point completion network. 2018 International Conference on 3D Vision. p. 728–737.

Zhang X, Feng Y, Li S, Zou C, Wan H, Zhao X, Guo Y, Gao Y. 2021. View-guided point cloud completion. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. p. 15890–15899.

Zhong M, Sui L, Wang Z, Hu D. 2020a. Pavement crack detection from mobile laser scanning point clouds using a time grid. Sensors. 20(15):4198.

Zhong M, Sui L, Wang Z, Yang X, Zhang C, Chen N. 2020b. Recovering missing trajectory data for mobile laser scanning systems. Remote Sens. 12(6):899.

Zhu J, Gehrung J, Huang R, Borgmann B, Sun Z, Hoegner L, Hebel M, Xu Y, Stilla U. 2020. TUM-MLS-2016: an annotated mobile LiDAR dataset of the TUM city campus for semantic point cloud interpretation in urban areas. Remote Sens. 12(11):1875.