# DEEP LEARNING-BASED ROAD SEGMENTATION OF 3D POINT CLOUDS FOR ASSISTING ROAD ALIGNMENT PARAMETERIZATION

M. Soilán<sup>1\*</sup>, H. Tardy<sup>1</sup>, D. González-Aguilera<sup>1</sup>

<sup>1</sup> Department of Cartographic and Terrain Engineering, University of Salamanca, Calle Hornos Caleros 50, 05003 Ávila, Spain; msoilan@usal.es, hugo.tardy@usal.es, daguilera@usal.es

#### Commission II, WG II/3

KEY WORDS: Mobile Mapping System, 3D point cloud processing, Deep Learning, Point Transformer, Road alignment, BIM

#### **ABSTRACT:**

The need for transportation infrastructure digitalization is becoming more important, and efficient data collection and processing workflows have to be established and pose a great research challenge. This paper presents a fully automated method for the geometric parametrization of the road alignment from 3D point clouds acquired with a low-cost mobile mapping system. It exploits the Point Transformer Deep Learning architecture in order to segment the 3D point cloud in four different classes, which include road markings. Those markings are then used as a reference to extract the alignment trajectory path, classify its geometries (straight lines, circular arcs, and clothoids) and then parametrize it, extracting data to easily generate alignment data that may follow the standard schema of the Industry Foundation Classes (IFC). Both the deep learning architecture and the geometry parametrization process show promising results to develop automatic workflows that extract precise as-built data of the infrastructure from 3D point clouds.

### 1. INTRODUCTION

Transportation infrastructure digitalization is a current challenge, as the technology development and society needs are evolving towards the necessity of delivering safer and more efficient solutions for road design and life management, as well as for a better user experience. A digital model of the infrastructure will have several benefits, not only reducing costs but also increasing safety and decreasing uncertainty (Azhar, 2011). Building Information Modelling (BIM) is a collaborative methodology to exchange and share information of a building project, in an interoperative manner, and during the whole life cycle of a project. While BIM is well established in the building construction industry, is it still being slowly introduced in civil engineering projects of transportation infrastructure (road, railway) (Costin et al., 2018). A great effort to move towards a standardization of information exchange formats is currently being done by organizations such as buildingSMART, by adding the transportation infrastructure domain to the Industry Foundation Classes (IFC) standard schema (BuildingSMART, 2018). Although BIM-based projects are expected to become the norm for public administrations, the digitalization of built infrastructure poses another challenge as the existing data of the built infrastructure may have different levels of information and completeness. For that reason, the usage of remote sensing surveying technologies is key to collect as-built infrastructure data towards its digitalization.

The main objective of this work is to develop a fully automated method to parametrize the horizontal alignment of a road using a 3D point cloud acquired with a Mobile Mapping System (MMS). The relevance and wide range of applications of these systems have been largely discussed and reviewed in the literature (Gargoum and El-Basyouny, 2017; Guan et al., 2016; Ma et al., 2018). To be able to parametrize the road alignment is also relevant, as the alignment is the basis for other infrastructure data models (road, railway, bridge) in the IFC schema (Amann and Borrmann, 2015). Previous works have proposed methodologies for road alignment extraction: Holgado-Barco et al. (2015) propose a method similar to the one presented in this paper, where road markings and road axis are extracted in a first place, and then a parameterization of the alignment is done based on the axis curvature. However, the first step is a semiautomated method based on rather simple heuristics, and the case study validates only four geometries of a highway. Martín-Jiménez et al. (2018) improve the workflow for the classification of the horizontal alignment using RANSAC to detect circular arcs and straight lines and show a more complete case study. However, the weakness of using heuristic-based methods for road marking extraction still remains.

Being able to detect road features in a 3D point cloud is essential to extract reliable information to create information models on an automated manner. In this regard semantic segmentation of 3D point clouds tackles this issue through the association of each point to a specific semantic class. It is largely studied in the literature with applications in autonomous driving, inventory processes and construction of digital models.

To this end, regular machine learning methods with region growing or RANSAC proved to be an effective and computationally accessible solution (Vo et al., 2015). More complex workflows making use of voxelization methods are still in use to this day in semantic segmentation of roads and railways (Lamas et al., 2021). On the other side, bypassing the constraints of defining processes highly dependent on domain application, Deep Learning methods bloomed with the pioneering PointNet (Qi et al., 2016) by taking directly point clouds as input. The growing interest in deep learning and its potential in semantic point cloud segmentation is also shown through the increasing availability of larger and more diverse datasets. The scientific community released datasets featuring indoors areas (S3DIS, Armeni et al., 2016) as well as street environments (Semantic3D, Hackel et al., 2017) and roads (Paris-Lille, Roynard et al., 2018;

<sup>\*</sup> Corresponding author

Toronto3D, Tan et al., 2020) as researchers focused on handling endlessly growing points clouds or on inference speed (RandLa-Net, Hu et al., 2020).

Road markings detection are discussed in the literature due to their importance in autonomous driving (Hata and Wolf, 2014) but their segmentation in 3d point clouds is less addressed. Previous works proposed approaches relying on the utilisation of images in addition to the point cloud data (Danescu and Nedevschi, 2010) or using heuristic-based methods (Kumar et al., 2014). While multiple deep learning approaches exist for semantic segmentation of road environments, few were tested on their capacity to segment roads markings from intensity alone as Toronto3D is the only dataset featuring road markings as a distinct class.

As 3D point cloud segmentation is still an active topic in the literature, the state of the art is constantly being challenged and pushed further. One of these recent architectures is PointTransformer by Zhao et al. (2021). By transposing the use of self-attention layers usually indicated in the extraction of information in Natural Language Processing to point cloud features extraction, it reaches results on par with the state of the art on the popular S3DIS dataset. With a novel approach based on transformers and distinct from its predecessors this work explores its versatility in MMS point cloud segmentation.

With this context, the motivation of this work is to explore the capabilities of novel deep learning architectures for 3D point cloud segmentation and integrate them in an enhanced workflow where parametric information of the road horizontal alignment is obtained so it can be directly used to build information models. Thus, the contributions of this work are twofold:

- 1. Exploitation of a version of a Deep Learning model architecture based on Point Transformer for the semantic segmentation of 3D point clouds of road environments.
- 2. Development of a robust algorithm to classify and parametrize the road horizontal alignment with straight lines, circular arcs, and clothoids. These parameters should be easily ingested by an information model schema such as IFC.

The remaining work is organized as follows. Case study data is presented in Section 2. Section 3 describes the methodology, and its results are shown in Section 4. Finally, Section 5 wraps up the paper offering conclusions and future lines of work.

# 2. CASE STUDY DATA

This work employs data acquired with a custom Mobile Mapping System, based on a Phoenix Scout Ultra 32 laser scanner (Figure 1a). It is a low-cost system equipped with a Velodyne VLP-32C laser scanner, with 32 laser beams, and horizontal and vertical field of view of 360° and 40° respectively. Although it is a lowcost system, it has a scan rate of 600,000 measurements per second (PhoenixLidar, 2021), thus it is able to obtain dense 3D point clouds. It was mounted in a van, with an inclination of 45°. Data acquisition took place in Ávila (Spain) in July 2021, with a speed of approximately 80km/h, in a 6-km section of a conventional road (AV-110) (Figure 1b).

Furthermore, the alignment of the scanned road was offered by the local administration as a text file, exported by the software ISTRAM. It will serve as a ground truth to validate the proposed methodology. As for the 3D points cloud, ground truth data was obtained through a manual labelling to test the deep learning architecture. The construction of a ground truth through manual labelling is a tedious process, so the training dataset is relatively small, featuring 3M points for training and 3.5M for the test set after subsampling, each of them being divided into 4 classes: asphalt, road markings, road signs, and 'other'.

It should be noted that the point cloud also contains outliers, although negligible in quantity, and artefacts resulting from the passing of cars, which were manually labelled as belonging to the class 'other'. Consequently, the dataset is highly unbalanced towards the classes road and 'other'. Note that RGB colors are not provided.



(a)



(b)

Figure 1. (a) Mobile Mapping system. (b) Trajectory of the case study, a 6-km long road section near Ávila (Spain).

# 3. METHODOLOGY

This section is composed of two methodological blocks, addressing each paper contribution as outlined in Section 1. First, the Deep Learning architecture employed for point cloud segmentation is presented. Then, its results are employed to define a method for the parametrization of the road alignment.

## 3.1 Point cloud segmentation

**3.1.1 Data preparation:** As the point cloud is acquired from a round-trip, the density on the sides of the road is uneven. To account for this characteristic, point clouds are merged in order to ensure a uniform density on both sides of the road. Moreover, to reduce computations and homogenize point cloud density, the merged point clouds are subsampled with a space criterion of 3 cm between points. This results in a training set of approximately 3M points and a distinct test set of 3,5M points (Figure 2). The training set is composed of a traffic roundabout featuring numerous instances of the following 4 classes: asphalt, road markings, road signs and 'other'. The test set depicts a 250m long road segment with the same features.







Figure 2. Point cloud labelling on training set (a) and test set(b): each point is labelled defining four classes: asphalt (blue), road markings (green), road signs (yellow), other (red).

**3.1.2 Deep learning architecture:** Segmentation is a necessary step to extract elements of interest from complex environments such as roads. To this end, the Point Transformer network introduced by Zhao et al. (2021) is explored. Point Transformer is a network based on self-attention layers which use a concept analogous to queries, keys and values to enrich the input with contextual information. This architecture processes point clouds directly and shows state-of-the-art results for semantic segmentation on the S3DIS dataset.

More specifically, this work operates on an adaptation of the original Point Trasformer code. The original model (POSTECH-CVLab, 2021) was adapted to fit in the Pytorch Geometric library (Fey and Lenssen, 2019). The model follows the original architecture and features five encoders and five decoders, each one working on a cardinality reduced by a factor 4 at each encoder layer with the furthest sampling algorithm.

**3.1.3 Training parameters:** To fit point clouds into memory, two points are defined randomly as cluster centres and the other points are assigned to the closest cluster. This operation is repeated recursively until the number of points in a cluster is below a fixed threshold of 1500 points. Clusters are then batched together and fed to the neural network. When used in a context of inference, the batch size is reduced to 1 with a unique cluster of 48k points.

**3.1.4 Model training:** The model is trained from scratch with Adam for 80 epochs while the learning rate initially fixed to 0.001 is decayed by a 0.1 factor every 20 epochs, with a batch size of 32. To account for the class imbalance, a weighted cross entropy loss is used with weights inversely proportional to the number of points from each class in the training set. Only the best performing model on the test set is kept.

The intensity is used as an additional feature. To augment the data, a random rotation around the Z-axis is applied, as well as a flipping of the positions and slight rotations of 15 degrees around the X and Y axis for each batch.

## 3.2 Road alignment parametrization

There are many applications that can be considered using as input the results from the point cloud segmentation shown in Section 3.1. This work will be focused on the parametrization of the road alignment as a set of straight lines, circular arcs, and transition curves or clothoids, following the specifications of IFC standard. According to national normative (Ministerio del Fomento, 2016), the alignment of a conventional road such as the one in the case study is given by the road marking that divides both traffic flow directions, not having into account any eventual, additional lane. The previous segmentation of the road platform and road markings is key for this section of the methodology, which is summarized in Figure 3.



Figure 3. Road alignment parameterization workflow.

Data pre-processing: The input to this part of the 3.2.1 methodology is a segmented 3D point cloud, with an additional attribute that is added to each point, defining its class. The first step selects only those points corresponding to asphalt and road marking classes. Although the segmentation results are good, there are still false positive points that may jeopardize future steps if they are not filtered out. For that reason, the point cloud is rasterized (projected on a squared grid with a size of 20cm on the XY plane, obtaining an image with enough resolution to isolate false positives), and a binary image is computed such that pixels occupied by at least one point are set as True (Figure 4a). Then, pixels that correspond to pavement are selected with two steps: (1) Selecting the largest connected component, and (2) computing a closing operation with a structural element with diamond shape and 5 pixels of radius. (Figure 4b). Then, only those points segmented as road markings whose projection lies inside a pavement pixel are considered for further processing. Furthermore, as there may be false positives on points from overpasses or cars that are also over the pavement, a height filter is set such that road marking points whose height with respect to its closest pavement point in the XY projection is larger than 20cm.



Figure 4. Data pre-processing. (a) Rasterization of pavement and road marking points. (b) Morphological operations are carried out to filter false positives.

Road centreline detection: At this stage, points labelled 3.2.2 as road markings offer a reliable description of the road lines and edges. To build the road alignment, the objective is to extract the road centreline, according to the national normative. In order to decrease the complexity on the point cloud processing side, the MMS vehicle drove along the closest road lane to the centreline during the whole dataset acquisition. This way, the vehicle trajectory is employed to extract an ordered set of points that belong to the road centreline. For each trajectory point, the 3D point cloud that contains road markings is transformed such that the y-axis corresponds to the vehicle heading and the x-axis points perpendicularly to the right in the direction of the movement. Then, points in the interval [-4, -1]m in x-axis direction and  $[0, y_2]$  in the y-direction are selected, where  $y_2$  is the y-coordinate of the next trajectory point (Figure 5a). Finally, for each trajectory point, the closest point among those selected in the previous step is extracted, obtaining a set of ordered points that follow the road centreline (Figure 5b).



Figure 5. Road centreline detection. (a) Schema of the road centreline extraction using the vehicle trajectory as a reference. Points inside the dashed rectangle are selected for further processing. (b) The closest road marking point to each trajectory point is extracted.

**3.2.3** Centreline curvature refinement: The road curvature is a parameter which defines the type of geometry of the road alignment at a given point. However, computing the curvature of the points extracted on the previous point would be too noisy to obtain reliable results. For that reason, the curvature of the centreline is refined in two steps. First, the 2D coordinates (XY) of the road centreline points are transformed to a Frenet coordinate system, as they provide a smooth and continuous trajectory where any point can be expressed as:

$$[x, y, \theta, k, \delta k, s] \tag{1}$$

Where (x, y) are the coordinates of the point,  $\theta$  is the tangent of the curve with respect to the x-axis,  $(k, \delta k)$  are the curvature and its derivative, and *s* is the distance along the curve from its origin. This approximation allows a uniform sampling of centreline coordinates as it transforms a discrete set of points into a continuous curve along the centreline (Figure 6a). Thus, a set of (x, y) coordinates are computed by sampling points along the Frenet reference path from s = 0 to s = L, where L is the total length of the path, with a step of 10 centimeters.

Finally, in order to obtain a robust and reliable definition of the curvature, the approach in Lin et al. (2010) is considered. There, line integrals are used to make the curvature estimation inherently robust to noise. Furthermore, the data window (radius) for the line integrals is adjusted via wild bootstrapping. This approach improves considerably the robustness of the curvature estimation (Figure 6b), enabling the next step of the process.



Figure 6. Centreline curvature. (a) A discrete number of points from the centreline is transformed to a Frenet trajectory path and seen as a continuous curve. (b) The curvature of the alignment is computed and refined using the robust approximation from Lin et al. (2010).

**3.2.4** Alignment geometry classification: This step aims to classify the ordered, uniformly sampled (x, y) points from the road centreline into the three geometry classes that define the road alignment: Straight lines, circular arcs, and clothoids. The curvature is used as the only feature to perform this classification as it univocally defines the geometry class:

- Straight lines: Curvature is constant and equal to zero.
- Circular arcs: Curvature is constant and not zero.
- Clothoids: Curvature is not constant.

However, the classification is not straightforward. Although the curvature estimation is robust, the input data comes from a 3D point cloud so there will always be a certain noise level. For that reason, a heuristic process is followed to classify the geometries. First, the curvature, expressed as a vector, is smoothed using a gaussian filter, with a window of 100 points (given the point sampling, this is equivalent to 10 meters) which was empirically defined (Figure 7a). Then, maxima and minima of the curvature are computed, and they are assumed to be points that belong to a circular arc if their curvature value is larger, in absolute value, than a small threshold  $\epsilon = 10^{-4}$ . Subsequently, a region growing process is applied. In the first iteration, the point with a peak curvature and its two closest neighbours are selected. Then, a circular arc is fitted and the distance between the arc and each point is computed. If any distance is larger than a threshold that

indicates that the fitting has a relevant error  $(d_{th} = 5cm)$ , the region growing stops. This process is repeated until this condition is reached, and it is done separately in both directions (to the right and to the left of the peak point).

The process for classifying straight lines is similar. First, points whose curvature is smaller than  $\epsilon$  are clustered. Then, for each cluster, the same region growing strategy than for circular arcs is applied, adding more points to the initial cluster as long as the distance between the fitted straight line and any point is smaller than  $d_{th}$ .

Finally, clothoids are straightforwardly classified as the remaining sets of consecutive points that are not classified either as straight line or as circular arc, so the road alignment is defined as a set of point clusters whose geometry is known (Figure 7b). In practice, this output is obtained as an array of geometric objects from the Clothoids Toolbox (Bertolazzi, 2021). This toolbox allows to define the alignment geometry with tangential continuity, ensuring a smooth transition between different geometries.



(b)

Figure 7. Alignment geometry classification. (a) The curvature plot is smoothed to remove noise and simplify the geometry classification process. (b) The alignment points are grouped according to three possible geometries: straight lines (green), circular arcs (red) and clothoids (blue). **3.2.5** Alignment geometry parametrization: The last step consists in extracting meaningful geometric parameters of each geometry. This is done considering the specifications of IFC standard so an ifcAlignment instance can be easily defined from the output of this method. The required parameters depend on the geometry, and can be summarized as:

- Straight lines: First point of the segment (x, y), direction of the segment, as an angle with respect to the x-axis (α), and length of the segment.
- *Circular arcs*: First point of the segment (x, y), direction of the tangent on that point  $(\alpha)$ , length of the segment (L), radius (R) and a Boolean indicating whether the segment is defined clockwise or counterclockwise (*isCCW*).
- *Clothoids*: First point of the segment (x, y), direction of the tangent on that point  $(\alpha)$ , length of the segment (L), radius at the beginning  $(R_0)$  and at the end  $(R_L)$  of the segment, and two Booleans indicating whether the segment is defined clockwise or counter clockwise at the beginning  $(isCCW_0)$  and at the end  $(isCCW_L)$  of the segment.

This way, from a raw point cloud, this process outputs, in an automated manner, a set of parameters that allow the generation of the road alignment in IFC – but also can be used to express the alignment following a different schema.

#### 4. **RESULTS**

### 4.1 Point cloud segmentation

Following metrics which are common in the literature, we mainly use the mean Intersection over Union metric to choose the best performing model. We additionally compute the following metrics class-wise: precision, recall and f-score. They are resumed in Table 1.

The two most represented classes, namely road and 'other' are the best segmented classes. Thanks to the use of intensity, road markings are mostly well segmented and road signs constitute the main difficulty of the dataset.

Table 1. Class-wise and average metrics of point cloud

segmentation.					
Metric/	Asphalt	Marking	Sign	Other	Avg
class					
Precision	0.97	0.70	0.88	1.00	0.89
Recall	0.96	0.95	0.74	0.97	0.91
F-score	0.96	0.74	0.81	0.98	0.89
IoU	0.93	0.67	0.67	0.93	0.81

By taking a closer look at the confusion matrix represented in Table 2, the road signs are mostly misclassified as 'other', which in the test set can be explained by the model relying mostly on the intensity factor to classify them. Indeed, the signs whose reflective panel is captured by the sensor are correctly classified while the others are not, as illustrated in (Figure 8).

As for the road markings, which also rely on intensity, they are most of the time mistaken for asphalt, which makes sense considering their proximity geometrically speaking.

Table 2. Confusion matrix.

The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLIII-B2-2022 XXIV ISPRS Congress (2022 edition), 6–11 June 2022, Nice, France

GT/ Prediction	Asphalt	Marking	Sign	Other
Asphalt	1546113	61371	5	1547
Markings	5557	146783	0	2534
Signs	0	5	4069	1424
Other	45258	2593	536	1679523



(b)

Figure 8. Road sign confusion illustration in a complex part of the test set. The model relies highly on intensity to classify road signs. (a) Point cloud visualization based on intensity (b) Classification of the model, with *signs* in yellow and *other* class as red.

Overall, the model reaches a mean IoU of 0.81 and allows to use the points classified as asphalt and road markings as a basis for the road alignment parametrization.

#### 4.2 Road alignment parameterization

To validate the methodology from Section 3.2, the road alignment ground truth data introduced in Section 2 is employed. It is a text file, exported by the software ISTRAM, that represents the road horizontal alignment as a set of geometries (straight lines, circular arcs and clothoids). Each geometry has several properties (initial point, radius, azimuth...) that makes possible to reconstruct the horizontal alignment in the same format as the output of the proposed methodology, by using Bertolazzi (2021) Clothoids toolbox.

This way, it is possible to quantify the precision of the results from the automated process, by comparing the distances between both alignments. For that purpose, the horizontal alignment resulting from the automated process is sampled, extracting a point each 10 centimetres. Then, the distance between each point and the ground truth alignment is computed. Figure 9 shows a histogram of the distances. The error metrics are shown in Table 3.



Figure 9. Histogram that represents the distances between the ground truth and the alignment obtained with the proposed method.

Table 3. Average and median errors for the positioning of the horizontal alignment

_	nonzontai angiment	
	Average error (cm)	Median error (cm)
	31.47	18.71

There are different error sources that can be discussed. First, this process takes as reference to build the alignment points in the edge of the road centreline, while the actual alignment is theoretically defined its midpoint. Second, the heuristics presented in the geometry classification process define a set of geometries that fit the centreline points but may slightly differ from the actual geometries in the ground truth data, especially in its initial and final points. Third, largest errors in the dataset were found to be due to discrepancies between the designed and the actual alignments. Figure 10a shows both alignments as a set of sampled points over a satellite view, and Figure 10b zooms in to highlight this last error source.



Figure 10. Visualization of the results in Google Earth. (a) Alignment computed by the proposed method (yellow) and ground truth data (red) are sampled over the satellite image. (b) A source of large errors are discrepancies between the ground truth data and the actual road centreline.

# 5. CONCLUSIONS

This work presents a fully automated workflow for road alignment parametrization using 3D point clouds acquired by a low-cost mobile mapping system. First, a deep learning model based on a recent architecture (Point Transformer) is applied using a small sample from the case study, which is a conventional road section of 6 kilometres of length. That sample is manually labelled and employed to segment the complete dataset with good quantitative results, proving its capability and viability in segmenting road assets and, specially, road markings.

The results from this automatic segmentation were exploited to develop a heuristic process in which the road centreline is extracted and regularly sampled. Then, its local curvature is computed and used to classify the road geometry in three geometric classes: straight lines, circular arcs and clothoids. Then, each curve is parametrized, extracting geometric information that can be used to easily generate the road alignment following standards such as IFC.

This workflow (exploiting state-of-the-art deep learning architectures to simplify the modelling of road infrastructure) is especially interesting in a context where BIM methodologies and digitalization are becoming more common in the linear infrastructure. Future research should be able to effectively extract more information that is already available in this work, such as the road pavement or vertical traffic signs, in order to obtain richer digital models of the as-built infrastructure with minimum user interaction, from geomatic data acquired with mobile mapping systems.

# 6. ACKNOWLEDGEMENTS

The research leading to these results received funding from the Strategic Program of Research Consortia National Business (CDTI) at the INROAD 4.0 project (IDI-20181117) under the call 2018 for the strategic program CIEN. This work has been partially supported by the Spanish Ministry of Science and Innovation through the grant FJC2018-035550-I funded by MCIN/AIE/ 10.13039/501100011033.

### 7. REFERENCES

Amann, J., Borrmann, A., 2015. Creating a 3D-BIM-compliant road design based on IFC alignment originating from an OKSTRA-accordant 2D road design using the TUM Open Infra Platform and the OKSTRA class library. *6th OKSTRA-Symposium.* 

Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S., 2016. 3D Semantic Parsing of Large-Scale Indoor Spaces. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1534–1543. https://doi.org/10.1109/CVPR.2016.170

Azhar, S., 2011. Building information modeling (BIM): Trends, benefits, risks, and challenges for the AEC industry. *Leadership* and Management in Engineering 11, 241–252. https://doi.org/10.1061/(ASCE)LM.1943-5630.0000127

Bertolazzi, E., 2021. Clothoids Toolbox. URL https://github.com/ebertolazzi/Clothoids (accessed 12.20.21).

BuildingSMART, 2018. buildingSMART - Official Website. URL https://www.buildingsmart.org/ (accessed 16.3.20).

Costin, A., Adibfar, A., Hu, H., Chen, S.S., 2018. Building Information Modeling (BIM) for transportation infrastructure – Literature review, applications, challenges, and recommendations. *Automation in Construction*. https://doi.org/10.1016/j.autcon.2018.07.001

Danescu, R., Nedevschi, S., 2010. Detection and classification of painted road objects for intersection assistance applications. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC.* pp. 433–438. https://doi.org/10.1109/ITSC.2010.5625261

Fey, M., Lenssen, J.E., 2019. Fast Graph Representation Learning with PyTorch Geometric. *International Conference on Learning* https://doi.org/10.48550/arxiv.1903.02428

Gargoum, S., El-Basyouny, K., 2017. Automated extraction of road features using LiDAR data: A review of LiDAR applications in transportation. *4th International Conference on Transportation Information and Safety, ICTIS - Proceedings* 563–574. https://doi.org/10.1109/ICTIS.2017.8047822

Guan, H., Li, J., Cao, S., Yu, Y., 2016. Use of mobile LiDAR in road information inventory: A review. *International Journal of Image* and Data Fusion. https://doi.org/10.1080/19479832.2016.1188860

Hackel, T., Savinov, N., Ladicky, L., Wegner, J.D., Schindler, K., Pollefeys, M., 2017. Semantic3D.net: A new Large-scale Point Cloud Classification Benchmark. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* IV-1/W1, 91–98. https://doi.org/10.5194/isprs-annals-IV-1-W1-91-2017

Hata, A., Wolf, D., 2014. Road marking detection using LIDAR reflective intensity data and its application to vehicle localization. *17th IEEE International Conference on Intelligent Transportation Systems, ITSC* 584–589. https://doi.org/10.1109/ITSC.2014.6957753

Holgado-Barco, A., González-Aguilera, D., Arias-Sanchez, P., Martinez-Sanchez, J., 2015. Semiautomatic extraction of road horizontal alignment from a mobile LiDAR system. *Computer-Aided Civil and Infrastructure Engineering* 30, 217–228. https://doi.org/10.1111/mice.12087

Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., Markham, A., 2020. Randla-Net: Efficient semantic segmentation of large-scale point clouds. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 11105–11114. https://doi.org/10.1109/CVPR42600.2020.01112

Kumar, P., McElhinney, C.P., Lewis, P., McCarthy, T., 2014. Automated road markings extraction from mobile laser scanning data. *International Journal of Applied Earth Observation and Geoinformation* 32, 125–137. https://doi.org/10.1016/j.jag.2014.03.023

Lamas, D., Soilán, M., Grandío, J., Riveiro, B., 2021. Automatic point cloud semantic segmentation of complex railway environments. *Remote Sensing* 13, 2332. https://doi.org/10.3390/rs13122332

Lin, W.Y., Chiu, Y.L., Widder, K.R., Hu, Y.H., Boston, N., 2010. Robust and accurate curvature estimation using adaptive line integrals. *Eurasip Journal on Advances in Signal Processing* 240309. https://doi.org/10.1155/2010/240309

Ma, L., Li, Y., Li, J., Wang, C., Wang, R., Chapman, M.A., 2018. Mobile laser scanned point-clouds for road object detection and extraction: A review. *Remote Sensing* 10, 1–33. https://doi.org/10.3390/rs10101531

Martín-Jiménez, J.A., Zazo, S., Arranz Justel, J.J., Rodríguez-Gonzálvez, P., González-Aguilera, D., 2018. Road safety evaluation through automatic extraction of road horizontal alignments from Mobile LiDAR System and inductive reasoning based on a decision tree. *ISPRS Journal of Photogrammetry and Remote* Sensing 146, 334–346. https://doi.org/10.1016/j.isprsjprs.2018.10.004

Ministerio del Fomento, 2016. Norma 3.1 -IC Trazado, Gobierno de España.

PhoenixLidar, 2021. Phoenix Scout-32 Spec sheet. URL https://www.phoenixlidar.com/wp-content/uploads/2020/06/PLS-ScoutUltra\_Spec-Sheet\_2020.pdf

(accessed 4.3.22). POSTECH-CVLab, 2021. Point Transformer. URL

https://github.com/POSTECH-CVLab/point-transformer (accessed 15.3.22).

Qi, C.R., Su, H., Mo, K.K., Guibas, L.J., 2016. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation, *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR.* 601–610. https://doi.org/10.1109/CVPR.2017.16

Roynard, X., Deschaud, J.E., Goulette, F., 2018. Paris-Lille-3D: A large and high-quality ground-truth urban point cloud dataset for automatic segmentation and classification. *International Journal of Robotics Research* 37, 545–557. https://doi.org/10.1177/0278364918767506

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. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops* 797–806. https://doi.org/10.1100/CVPPW50408.2020.00100

https://doi.org/10.1109/CVPRW50498.2020.00109

Vo, A.-V., Truong-Hong, L., Laefer, D.F., Bertolotto, M., 2015. Octree-based region growing for point cloud segmentation. *ISPRS Journal of Photogrammetry and Remote Sensing* 104, 88– 100. https://doi.org/10.1016/j.isprsjprs.2015.01.011

Zhao, H., Jiang, L., Jia, J., Torr, P., Koltun, V., 2021. Point transformer. Preprint. https://doi.org/https://doi.org/10.48550/arXiv.2012.09164