3D POINT CLOUD TO BIM: AUTOMATED APPLICATION TO DEFINE IFC ALIGNMENT AND ROADWAY WIDTH ENTITIES FROM MLS-ACQUIRED LiDAR DATA OF MOUNTAIN ROADS

D. Lamas [1, *], A. Justo [1], M. Soilán [2], B. Riveiro [1]

[1] CINTECX, Universidade de Vigo, GeoTECH Group, Campus Universitario de Vigo, As Lagoas, Marcosende, 36310 Vigo, Spain - (daniel.lamas.novoa, andres.justo.dominguez, belenriveiro)@uvigo.gal

[2] Department of Cartographic and Land Engineering, Higher Polytechnic School of Ávila, Universidad de Salamanca, Hornos Caleros 50, 05003 Ávila, Spain – msoilan@uvigo.gal

Commission IV, WG IV/9

KEY WORDS: mobile laser scanning, point cloud processing, infrastructure information models, building information modelling, Industry Foundation Classes, road alignment modelling.

ABSTRACT:

The growing trend of developing standards of information exchange and management processes is leading to Building Information Models (BIM) being adapted to work with linear infrastructure assets. For this reason, the Industry Foundation Classes (IFC) has developed standards for linear infrastructure such as roads. Furthermore, the usage of remote sensing technologies, such as Mobile Laser Scanning (MLS) systems for infrastructure monitoring is increasingly common. This paper presents an automated methodology that takes as input 3D point cloud tiles from an MLS and its trajectory, and outputs an IFC-compliant file that models the alignment of the road and the width of the roadway along the length of the road. The methodology is evaluated in 48 km of mountain roads, in some cases without road markings, using neither intensity nor colour fields.

1. INTRODUCTION

Currently, there is a growing trend toward the development of applications that allow the storage and exchange of data in a practical, user-friendly and reliable way. In most cases, this implies that several disciplines, each with its casuistry and expertise, are looking for a common ground on which to share and structure information. This trend leads to the creation of standards that define a way of structuring data to meet those requirements. This has led the industry to adopt Building Information Modelling (BIM) methodologies (Eastman et al., 2008). The JRC Technical Reports of the European Commission 2017 (Poljanšek, 2017) defines BIM as "a digital tool disrupting the construction industry as a platform for central integrated design, modelling, asset planning running and cooperation. It provides all stakeholders with a digital representation of a building´s characteristic in its whole life-cycle and thereby holds out the promise of large efficiency gains."

In recent years, BIM has been a widely used practice in the building design process or applied in existing buildings, using three-dimensional (3D) digital representations, as well as functional and semantic features of the structure (López et al., 2018).

In the field of transportation infrastructure, BIM is increasingly used over the last years in railways (Huang et al., 2011; Soilán, Justo, et al., 2021; Soilán, Nóvoa, et al., 2021), bridges (Fanning et al., 2015; Kaewunruen et al., 2020), and roads (Biancardo et al., 2020; Chong et al., 2016; Soilán, Justo, et al., 2021; Zhao et al., 2019). This model of information structure is improving the interoperability and the integration of the information of large construction projects (Bradley et al., 2016; Costin et al., 2018; Isikdag et al., 2007).

As in the case of construction, BIM methodology in existing transport infrastructures is also interesting. Point cloud to BIM technics are used to acquire as-built modelling of the asset (Pətrəucean et al., 2015). This is challenging because it must represent the real outcome of the infrastructure, rather than the design or ideal asset.

One of the most widely used technologies for collecting geometrical data is Mobile Laser Scanning (MLS). This technology allows obtaining geometric and radiometric information about the asset in form of point clouds. There is vast literature showing their capabilities in transport infrastructures (Ma et al., 2018; Soilán et al., 2019).

For this reason, several authors have developed algorithms to automatically generate geometry from point clouds for BIM. This process requires an automatic segmentation of the point cloud and the integration of the information in a BIM.

In this regard, it is important to define how the information should be structured in a BIM. The Industry Foundation Classes (IFC) is an open standard for generating BIM, created by *buildingSMART* (*BuildingSMART - The International Home of BIM*, n.d.). Over the past years, they have been working on the transportation infrastructure domain. The newest version of the schema, IFC 4.3, includes linear infrastructure such as road and rail. In the road schema, the positions of the assets of the infrastructure are related to the alignment of the roadway, which typically corresponds to the centre of the roadway.

To define the alignment from point clouds, several authors have developed different methodologies. (Wen et al., 2019) proposed a framework for the segmentation, classification and completion of road markings. They use U-Net to segment the road markings,

* Corresponding author

a Convolutional Neural Network (CNN) to classify them, and a conditional Generative Adversarial Network (cGAN) to complete them. With the information of the road markings on the edges of the roadway, the generation of the alignment is trivial since they define it as the centre of the roadway. (Holgado-Barco et al., 2015) developed a semiautomatic road alignment extraction using intensity and scan angle thresholds for road marking segmentation. In previous works, (Soilán et al., 2020) present a semi-automatic methodology to define IFC alignment entities of highway roads. The methodology is based on the detection of road markings. The road markings are detected automatically, and the alignment is generated as a polyline from a semi-automatic selection of roadway boundaries. Different search windows are applied along the trajectory of the MLS used to record the point clouds by searching for marks using intensity values.

As a follow-up to those works, this paper aims to introduce an algorithm to improve the generation of IFC models for roads, presenting the following contributions:

1. Alignment extraction method which is applicable in complex cases study (mountain roads), where there may be no road markings delimiting the roadway.
2. Calculation of roadway width along the length of the road.
3. A geometry-based methodology that uses neither intensity nor colour.
4. IFC generation including alignment and roadway width information.
5. Fully automatic methodology using as input point clouds stored in a tile structure (commonly used in topography) and the trajectory of the MLS.

This work is structured as follows: First, the case study is presented in section 2. Then, section 3 describes the proposed methodology for the extraction of the alignment and roadway width from point clouds and the IFC generation. Then, the method is evaluated in section 4. Finally, section 5 presents the conclusions of this work.

2. CASE STUDY DATA

The scenario used to validate the methodology presented in this paper consists of 48 km of mountain roads. These mountain roads are winding and two-ways roads, which in some cases have no road markings delimiting neither their centre nor their boundaries, thus posing a relevant challenge for the development of automatic processes.

The data used to analyse this scenario are 3D georeferenced point clouds and the trajectory of the MLS used to record them. In Figure 1 shows point clouds with the corresponding sensor trajectory.

Moreover, the ground truth of the road alignment as a 1 m spacing points polyline is also available.

Figure 1. Point clouds (with colour) and trajectory (in red).

2.1 Trajectory

The trajectory of the MLS is a discrete set of 2D points with the coordinates of the position of the navigation sensor recorded each second.

2.2 Point cloud tiles

The dataset used to validate this methodology is a set of point clouds structured in tiles. The organization in tiles is widely used. It is especially useful for datasets that cover a large area and may consist of data from different sensors stored in different file types. With this organisation, all files with information from a specific tile can be accessed in a simple way. For computational reasons, the complete point cloud dataset is structured in tiles in LASzip (LAZ) format (*LASzip | Rapidlasso GmbH*, n.d.), which is a compressed LASer (LAS) format (*LASer (LAS) File Format Exchange Activities – ASPRS*, n.d.). These tiles have an area of 50x50 m and 1M points on average. In total, the dataset has a size of 10.6 GB formed by 2,598 tiles. As it is shown in Figure 2, the division of the tiles does not correspond to the direction of the road. Their division is done considering their (*x, y*) coordinates. This means that some tiles have incomplete road sections.

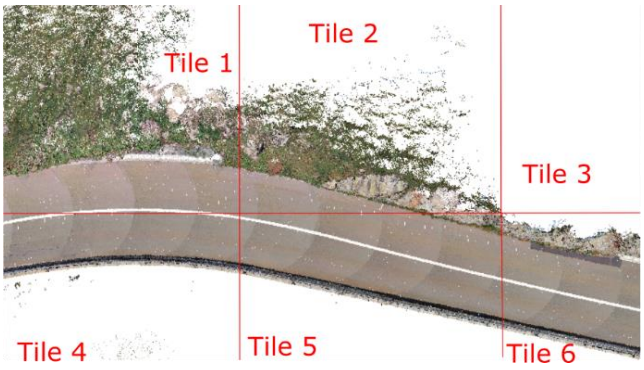

Figure 2. Point cloud tiles.

As it is shown in the figures of this paper, the point clouds are coloured. However, the colour attribute is not used in the methodology, so the method can be generalized to point cloud acquisition without colour.

3. METHODOLOGY

This paper presents an automatic IFC generation of mountain roads using 3D georeferenced point clouds recorded by an MLS. A schematic representation of the methodology is shown in Figure 3.

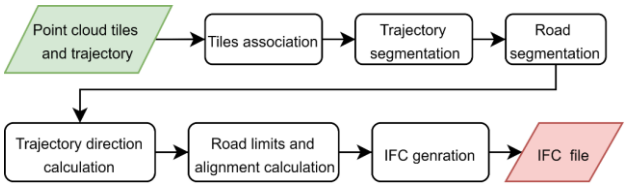

Figure 3. Proposed methodology workflow.

The methodology consists of a tile association process followed by a trajectory segmentation to construct point clouds of complete road sections covering a distance equal to sec_{dist}. Then, the road is segmented. Subsequently, the trajectory is analysed to calculate its perpendicular direction. With both results, the left and right limits of the roadway from the trajectory are determined and its alignment is obtained. Last, the IFC file including the alignment and the width of the roadway is generated.

The process is written in Python. Some functionalities of the Open3D (*CloudCompare*, n.d.) library are used. CloudCompare (Zhou et al., 2018) software is utilised to generate some images.

3.1 Tiles association

The point cloud tiles are not sorted according to any road-related information. However, the trajectory has the coordinates of its points on the roadway, and its points are ordered according to the direction of the road. For this reason, the tiles are sorted by assigning them to points of the trajectory.

A tile is assigned to all the points of the trajectory that are closer than d_{tile} from any point in the tile. To speed up this process, distances to tile points are not calculated. Instead of points, the boundary box limits of each tile are used to calculate whether a trajectory point is close enough to a tile. The boundary box is a field in the LAZ file. Figure 4 shows an example of the result of this process.

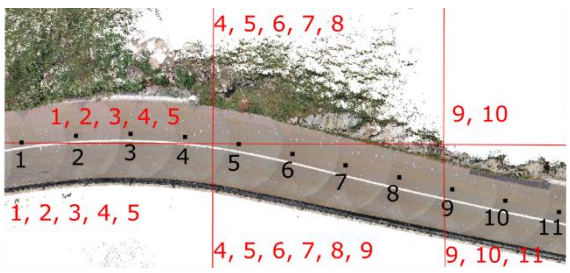

Figure 4. Tiles association. Trajectory points in black. Tiles assignation in red.

3.2 Trajectory segmentation

The processes applied to obtain the alignment and width of the road require the input data to have complete road information. However, the 48 km cannot be processed together.

Therefore, the trajectory is segmented into sections. The distance between each point and the next is calculated. Then, the points are grouped in the same section until the accumulative distance between them exceeds sec_{dist}.

Besides, as tiles are assigned to trajectory points, they are also sectioned. The tiles associated with any trajectory point in a section also belong to that section.

An example of this segmentation process is shown in Figure 5.

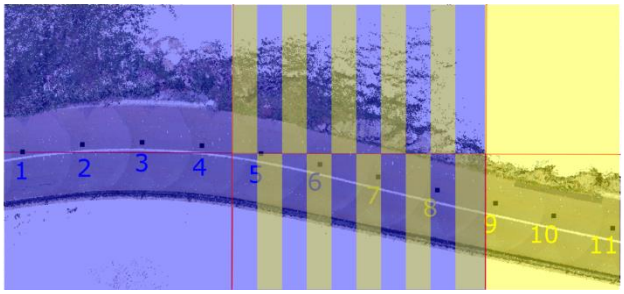

Figure 5. Sectioned trajectory points and tiles. The first section in blue. The second section in yellow. Tiles that belong to both sections are coloured in stripes of both colours.

3.3 Roadway segmentation

Once the trajectory is segmented and the tiles are assigned to their closest trajectory points, each section is analysed with its tiles.

In this step, the roadway is segmented. This process is described in Figure 6.

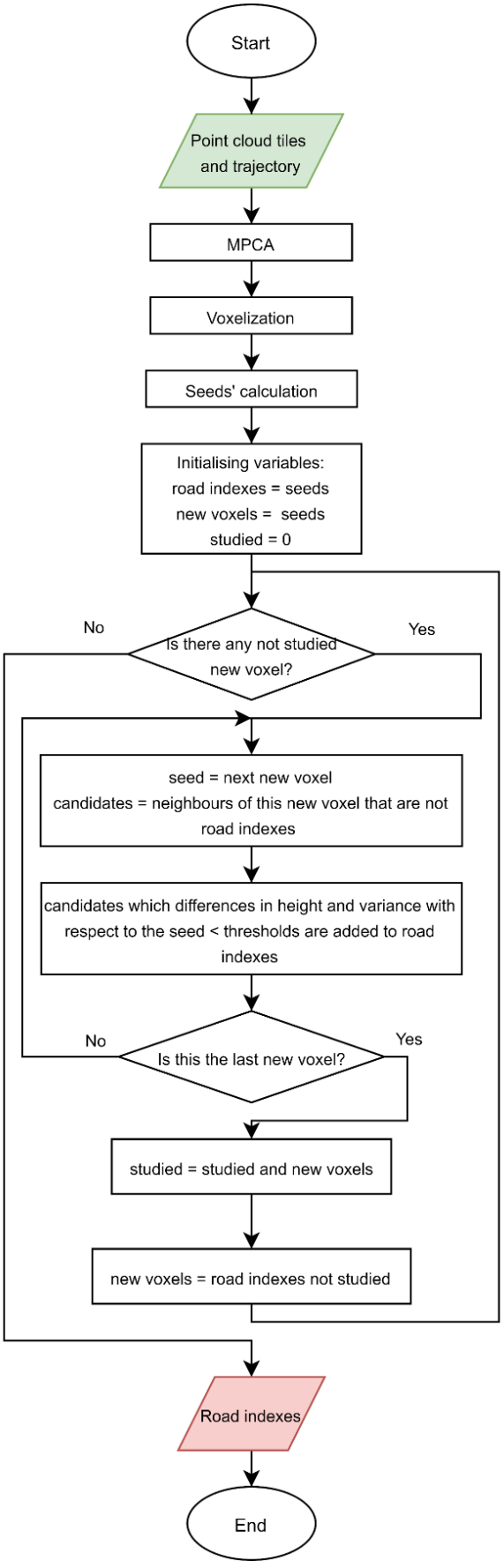

Figure 6. Roadway segmentation diagram.

To do so, the first phase is to apply a Modified Principal Component Analysis (MPCA) described in (Lamas et al., 2021). MPCA is a PCA but modifying the 2nd and the 3rd eigenvectors to adjust the 2nd eigenvector in the XY plane. MPCA is applied to the trajectory section, and its result is used to reorient the trajectory section and its tiles. This allows the flattering of possible slopes of the road, which complicates the roadway segmentation process. The modification of the PCA is necessary because the trajectory is a line, so its 2nd eigenvector must be defined.

Then, the point cloud is voxelised using a g_{vox} grid, calculating the mean and the variance of the vertical coordinate of each voxel. In addition, the neighbourhood of each voxel is also known.

Last, a region growing process explained in (Soilán et al., 2020) is applied. The first step consists of calculating the seeds of the roadway region. These seeds are the voxels closest to each trajectory point, calculated using the K-nearest neighbour algorithm (Silverman & Jones, 1989). In the first loop, the seeds are the new voxels of the roadway region. The neighbours of these new voxels are compared to their seed. Those voxels with a difference in vertical mean and variance lower than z_{road} and v_{road}, respectively, relative to their seed, are added to the roadway region. These are the next new voxels in the next loop. The voxels are only considered once as candidates to be added to the roadway region. The process stops when there are no new voxels added in a loop.

A point cloud with its roadway segmented is shown in Figure 7.

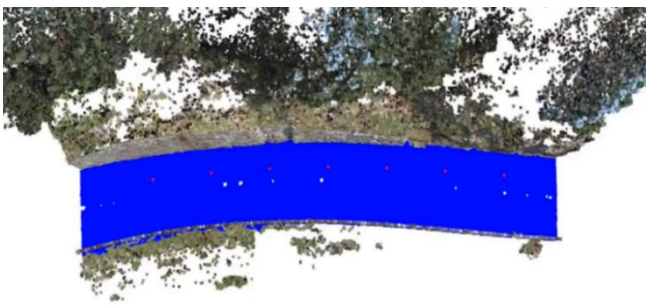

Figure 7. Roadway segmented in blue. The seeds in red are the closest points to each trajectory point.

3.4 Trajectory direction calculation

This section describes the process of calculating the direction perpendicular to the trajectory in the plane of the roadway. The process consists of applying MPCA (Lamas et al., 2021) to each point and its n_d nearest points. If possible, the points considered are half in front and half behind the selected trajectory point.

As a result, at each trajectory point, the 1st eigenvector is the direction of the trajectory, and the 2nd is the perpendicular direction in the plane of the roadway.

A representation of this calculation is shown in Figure 8.

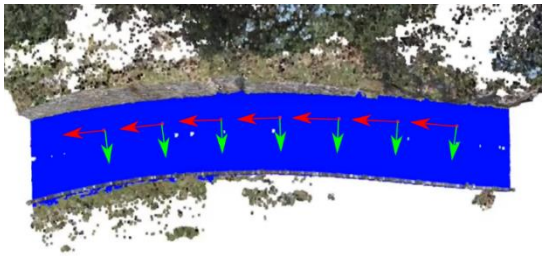

Figure 8. Trajectory direction analysis. Trajectory direction in red. The direction perpendicular to the trajectory in the plane of the roadway in green.

3.5 Alignment and roadway width calculation

Once the roadway is segmented and the trajectory direction is known, the next step is to calculate the alignment and the width of the roadway. The alignment is defined as the central axis of the roadway.

To achieve these objectives, an alignment point with the width of the roadway is calculated for each trajectory point. In this way, the alignment is defined as a polyline, whose points are distributed along the roadway with the same frequency as the trajectory points. Besides, the roadway width information is also known at each point of the polyline.

For this purpose, the following methodology is applied. For each trajectory point, a search window is defined. The window is centre at that point, with a width equal to ***w*** and is extended in the perpendicular direction to the trajectory. Then, the points farthest away from the trajectory are chosen, one on each side, within the window. The centre of the chosen points is the alignment point and their distance is the width of the roadway. This process is shown in Figure 9.

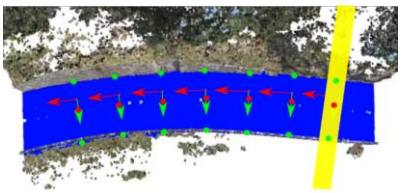

Figure 9. Roadway limits and alignment. Search window in yellow. The green points are the roadway limits. The red points are the alignment.

3.6 IFC generation

The final step of the methodology presented in this work is the generation of an infrastructure model compliant with the IFC data schema. As with any linear infrastructure, the key component of this model is the alignment, obtained in the previous steps as a polyline. This polyline is translated into the *IfcAlignment* entity as its geometrical representation, in the shape of an *IfcAlignmentCurve*. The curve is then defined by its two components, horizontal (*IfcAlignment2DHorizontal*) and vertical (*IfcAlignment2DVertical*). While the curve can be set to have only a horizontal component, and therefore obtain a 2D alignment, the same cannot be done for the vertical component, as it is dependent on the horizontal definition. Nevertheless, both of these components are formed of series of segments, in this case linear (*IfcLineSegment2D* & *IfcAlignment2DVerSegLine*). The *IfcLineSegment2D* requires to set its start point, direction and length. All of these parameters can be directly obtained for the polyline data. In the case of *IfcAlignment2DVerSegLine*, both the start point and the length are given in relationship to the horizontal component, representing distances along its curve. Besides those, it is also necessary to set the start height and gradient, which can also be directly obtained from the polyline.

A schematic representation is shown in Figure 10.

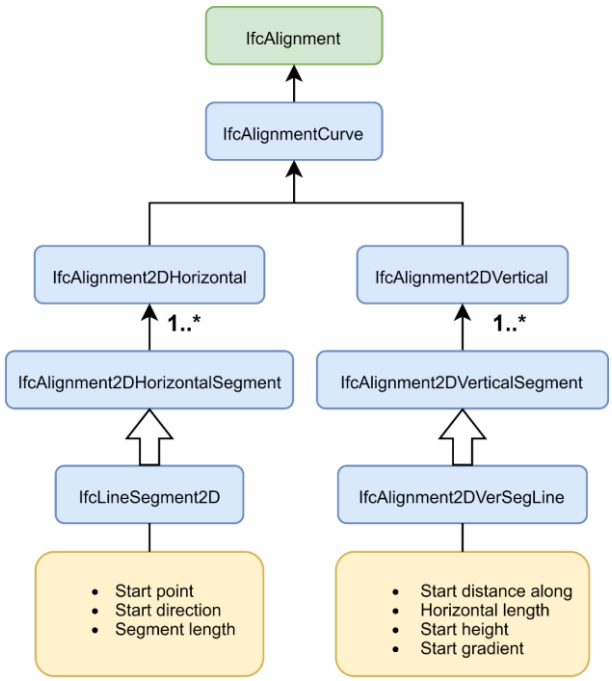

Figure 10. IFC alignment diagram.

Once the alignment has been defined, any other element can use it as means of placement or as the basis for a geometrical representation. For instance, if a traffic sign were to be placed, its position could be given by the kilometric point, and distance from the centre of the roadway. In this case, as the scope is limited to the roadway, the placement is reduced to the same starting point as the alignment, since the entire alignment will be covered by roadway.

On the other hand, the geometrical representation of the asphalt is directly dependent on the alignment. It is described by the *IfcSectionedHorizontalSolid*, which generates a solid by placing a series of profiles along a given curve. Therefore, it requires both the profiles, and the position of said profiles following the curve.

The profiles are set to *IfcRectangleProfileDef* using a predefined depth and the width values obtained from previous steps. Then, the profiles are placed using *IfcDistanceExpression*. In this case, the distance expressions only require the distance along the alignment. An example of the IFC roadway is shown in Figure 11.

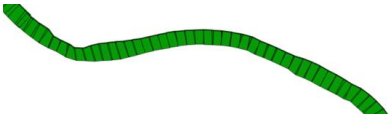

Figure 11. IFC roadway detail: alignment and width.

https://doi.org/10.5194/isprs-annals-X-4-W2-2022-169-2022

4. RESULTS

This section shows the results from the application of the presented methodology to the case study data. Moreover, this section also describes the process used to validate the methodology.

First, the parameters involved in the methodology are shown in Table 1. Some of these parameters are empirically adjusted, while others are based on the road geometry and the sensor used.

Parameter	Value	Parameter	Value
sec_{dist}	100 m	g_{vox}	0.1 m
z_{road}	0.0075 m	v_{road}	0.0075 m^2
n_d	4	w	6* sec_{dist} m

Table 1. Values of parameters involved in the methodology.

The validation of the process is performed on the entire dataset. Since the road alignment is obtained from the roadway boundaries, and the ground truth of the alignment is known but not the roadway width, the validation process is applied to the alignment.

As the ground truth of the alignment is a polyline, the validation process consists in evaluating the distance between the ground truth and the alignment calculated in a polyline format, before the generation of the IFC. The distance between a point of the alignment and the ground truth is defined in Equation (1). For each alignment point, the nearest point on the ground truth alignment is selected. Then, two lines are defined: one with the selected point and the previous one, and the other with the selected point and the next one. The alignment point error is the minimum distance between that point and either of the two lines.

$$d_i = \min\left(\frac{\|\overrightarrow{A_iG_j} \wedge \vec{u}_0\|}{\|\vec{u}_0\|}, \frac{\|\overrightarrow{A_iG_j} \wedge \vec{u}_1\|}{\|\vec{u}_1\|}\right) \quad (1)$$

where:

d_i = distance between the point i of the alignment to the ground truth.
A_i = : i point of the alignment.
G_j = closest ground truth point to A_i.
$\vec{u}_0$ = directing vector of the segment $\overline{G_jG_{j-1}}$.
$\vec{u}_0$ = directing vector of the segment $\overline{G_jG_{j+1}}$.

The distances obtained are shown in the histogram in Figure 12. The mean distance is 0.4938 m and the standard deviation is 0.2634 m. The 23% of the alignment has an error lower than 0.1, the 38% lower than 0.2, and the 70% lower than 0.6 m.

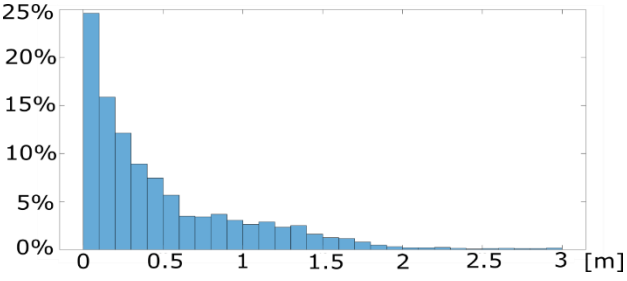

Figure 12. Histogram of the distances between each alignment point and the ground truth.

The result is also analysed in a visual form. As it is shown in Figure 13 and Figure 14, the process works correctly on painted and unpainted mountain roads under normal conditions. However, in areas where the road has road junctions or car lay-bys, such as in Figure 15, the limits of the asphalt are not the limits of the roadway, so the algorithm makes errors.

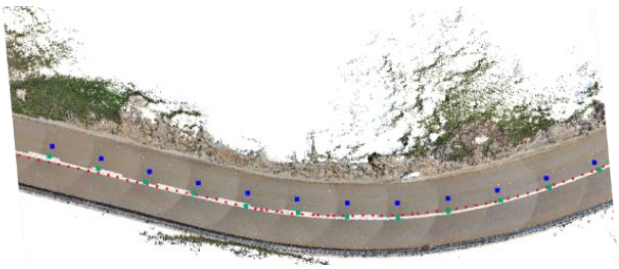

Figure 13. Comparison between the polylines of the ground truth, the alignment and the trajectory. Ground truth in red, alignment in green and trajectory in blue.

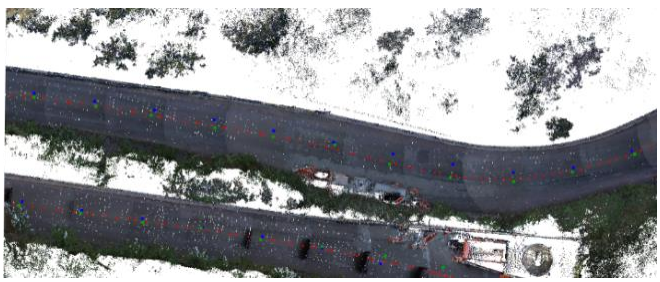

Figure 14. Comparison between the polylines of the ground truth, the alignment and the trajectory in a no painted road. Ground truth in red, alignment in green and trajectory in blue.

Figure 15. Comparison between the polylines of the ground truth, the alignment and the trajectory close to a car lay-by. Ground truth in red, alignment in green and trajectory in blue.

Last, Figure 16 shows a visual representation of the atomically generated IFC model, which contains alignment and roadway width information.

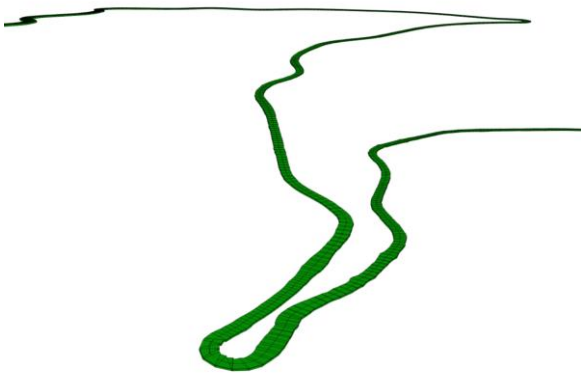

Figure 16. IFC roadway: alignment and width.

5. CONCLUSIONS

This paper presents a fully automated methodology that generates IFC models of roadways, which contains the alignment and the roadway width, calculated from 3D point clouds tiles of mountain roads. First, tiles are associated with trajectory points. Second, the trajectory and its corresponding tiles are divided into sections. Third, the roadway is segmented. Fourth, the perpendicular direction of the trajectory is calculated. Fifth, the alignment and the width of the roadway are calculated. Last, the IFC of the roadway is generated.

The proposed methodology is tested on 48 km of mountain roads. The colour and the intensity values of point clouds are not used. The results obtained are compared with the ground truth of the alignment of the road. The mean error is 0.4938 m and the standard deviation is 0.2634 m.

The difference between the mean and the standard deviation error shows that the process makes considerably large errors but in few areas. In fact, the 38% of the points have an error of less than 0.2 m, which is an admissible error.

Analysing the result visually, we detect that these isolated errors are located in specific areas of the road: car lay-bys and road junctions' areas. This is because the algorithm segments the entire asphalt as roadway.

For these reasons, there are interesting future research lines from this work. Errors could be identified by analysing the rapid variation in the direction of the alignment. These could allow not only to correct them by developing a different methodology for these specific areas, but also identify car lay-bys and road junctions. In this sense, it is also interesting to develop methods to segment different assets, such as signs, and automatically add them to the IFC model.

Finally, the alignment can be improved by parameterising the polyline with arcs, clotoides and lines, following the IFC schema.

ACKNOWLEDGEMENTS

Work produced with the support of a 2021 Leonardo Grant for Researchers and Cultural Creators, BBVA Foundation. The BBVA Foundation takes no responsibility for the opinions, statements and contents of this project, which are entirely the responsibility of its authors.

FUNDINGS

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.

REFERENCES

Biancardo, S. A., Viscione, N., Cerbone, A., & Dessì, E. (2020). BIM-Based Design for Road Infrastructure: A Critical Focus on Modeling Guardrails and Retaining Walls. *Infrastructures 2020, Vol. 5, Page 59*, *5*(7), 59. https://doi.org/10.3390/INFRASTRUCTURES5070059

Bradley, A., Li, H., Lark, R., & Dunn, S. (2016). BIM for infrastructure: An overall review and constructor perspective. *Automation in Construction*, *71*, 139–152. https://doi.org/10.1016/J.AUTCON.2016.08.019

buildingSMART - The International Home of BIM. (n.d.). Retrieved April 25, 2022, from https://www.buildingsmart.org/

Chong, H. Y., Lopez, R., Wang, J., Wang, X., & Zhao, Z. (2016). Comparative Analysis on the Adoption and Use of BIM in Road Infrastructure Projects. *Journal of Management in Engineering*, *32*(6), 05016021. https://doi.org/10.1061/(ASCE)ME.1943-5479.0000460

CloudCompare (2.11). (n.d.). Retrieved July 6, 2022, from https://www.cloudcompare.org/

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*, *94*, 257–281. https://doi.org/10.1016/J.AUTCON.2018.07.001

Eastman, C., Teicholz, P., Sacks, R., & Liston, K. (2008). *BIM Handbook : A Guide to Building Information Modeling for Owners, Managers, Designers, Engineers and Contractors*.

Fanning, B., Clevenger, C. M., Ozbek, M. E., & Mahmoud, H. (2015). Implementing BIM on Infrastructure: Comparison of Two Bridge Construction Projects. *Practice Periodical on Structural Design and Construction*, *20*(4), 04014044. https://doi.org/10.1061/(ASCE)SC.1943-5576.0000239

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*(3), 217–228. https://doi.org/10.1111/MICE.12087

Huang, S. F., Chen, C. S., & Dzeng, R. J. (2011). Design of Track Alignment Using Building Information Modeling. *Journal of Transportation Engineering*, *137*(11), 823–830. https://doi.org/10.1061/(ASCE)TE.1943-5436.0000287

Isikdag, U., Isikdag, U., Aouad, G., Underwood, J., & Wu, S. (2007). Building information models: a review on storage and exchange mechanisms, Bringing ITC knowledge to work. *24TH W78 CONFERENCE MARIBOR*. https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.454.6870

Kaewunruen, S., Sresakoolchai, J., & Zhou, Z. (2020). Sustainability-Based Lifecycle Management for Bridge Infrastructure Using 6D BIM. *Sustainability 2020, Vol. 12, Page 2436*, *12*(6), 2436. https://doi.org/10.3390/SU12062436

Lamas, D., Soilán, M., Grandío, J., & Riveiro, B. (2021). Automatic Point Cloud Semantic Segmentation of Complex Railway Environments. *Remote Sensing 2021, Vol. 13, Page 2332*, *13*(12), 2332. https://doi.org/10.3390/RS13122332

LASer (LAS) File Format Exchange Activities – ASPRS. (n.d.). Retrieved April 19, 2022, from https://www.asprs.org/divisions-committees/lidar-division/laser-las-file-format-exchange-activities#

LASzip | rapidlasso GmbH. (n.d.). Retrieved April 19, 2022, from https://rapidlasso.com/laszip/

López, F. J., Lerones, P. M., Llamas, J., Gómez-García-Bermejo, J., & Zalama, E. (2018). A Review of Heritage Building Information Modeling (H-BIM). *Multimodal Technologies and Interaction 2018, Vol. 2, Page 21*, *2*(2), 21. https://doi.org/10.3390/MTI2020021

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 2018, Vol. 10, Page 1531*, *10*(10), 1531. https://doi.org/10.3390/RS10101531

Pətrəucean, V., Armeni, I., Nahangi, M., Yeung, J., Brilakis, I., & Haas, C. (2015). State of research in automatic as-built modelling. *Advanced Engineering Informatics*, *29*(2), 162–171. https://doi.org/10.1016/J.AEI.2015.01.001

Poljanšek, M. (2017). *Building Information Modelling (BIM) standardization*. https://doi.org/10.2760/36471

Silverman, B. W., & Jones, M. C. (1989). E. Fix and J.L. Hodges (1951): An Important Contribution to Nonparametric Discriminant Analysis and Density Estimation: Commentary on Fix and Hodges (1951). *International Statistical Review / Revue Internationale de Statistique*, *57*(3), 233. https://doi.org/10.2307/1403796

Soilán, M., Justo, A., Sánchez-Rodríguez, A., Lamas, D., & Riveiro, B. (2021). 3D point cloud data processing and infrastructure information models: Methods and findings from safeway project. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, *43*(B2-2021), 239–246. https://doi.org/10.5194/ISPRS-ARCHIVES-XLIII-B2-2021-239-2021

Soilán, M., Justo, A., Sánchez-Rodríguez, A., & Riveiro, B. (2020). 3D Point Cloud to BIM: Semi-Automated Framework to Define IFC Alignment Entities from MLS-Acquired LiDAR Data of Highway Roads. *Remote Sensing 2020, Vol. 12, Page 2301*, *12*(14), 2301. https://doi.org/10.3390/RS12142301

Soilán, M., Nóvoa, A., Sánchez-Rodríguez, A., Justo, A., & Riveiro, B. (2021). Fully automated methodology for the delineation of railway lanes and the generation of IFC alignment models using 3D point cloud data. *Automation in Construction*, *126*. https://doi.org/10.1016/j.autcon.2021.103684

Soilán, M., Sánchez-Rodríguez, A., del Río-Barral, P., Perez-Collazo, C., Arias, P., & Riveiro, B. (2019). *infrastructures Review of Laser Scanning Technologies and Their Applications for Road and Railway Infrastructure Monitoring*. https://doi.org/10.3390/infrastructures4040058

Wen, C., Sun, X., Li, J., Wang, C., Guo, Y., & Habib, A. (2019). A deep learning framework for road marking extraction, classification and completion from mobile laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, *147*, 178–192. https://doi.org/10.1016/J.ISPRSJPRS.2018.10.007

Zhao, L., Liu, Z., & Mbachu, J. (2019). Highway Alignment Optimization: An Integrated BIM and GIS Approach. *ISPRS International Journal of Geo-Information 2019, Vol. 8, Page 172*, *8*(4), 172. https://doi.org/10.3390/IJGI8040172

Zhou, Q.-Y., Park, J., & Koltun, V. (2018). *Open3D: A Modern Library for 3D Data Processing*. https://doi.org/10.48550/arxiv.1801.09847