Point Cloud and BIM-based Quality Check of Building Structures

Alojz KOPÁČIK, Ján ERDÉLYI and Richard HONTI, Slovakia

Key words: Engineering survey, Laser scanning, Quantity surveying, BIM, Modelling, Quality check, Building structures

SUMMARY

Automation in the building industry has increased significantly over the last few years. The base of this effort creates both the broad application of BIM (Building Information Modelling) and the common usage of modern data acquisition and modelling technologies. In cases when the BIM model is disposable, this could be used as the as-planned model of the building (structure). For an effective in-situ check of the construction quality, semi-automated control of the building's produced elements (parts) is implemented in the construction process. The fast and effective data acquisition using different scanning methods and automated or semi-automated modelling built the base of these processes. Converting point clouds, which are products of the scanning process, into a BIM model is often called "scan-to-BIM." In cases when integrated into this process, the quality check is called "scan-vs-BIM". The method includes data acquisition by scanning and point cloud adjustment with registration, which enables the creation of two data sets and models in a common coordinate system. Next, the models are compared and evaluated. Finally, deviations between the models are calculated and presented. The paper describes the developed methodology and shows the results of its practical application during the construction of a polyfunctional building with an existing BIM model.

Point Cloud and BIM-based Quality Check of Building Structures

Alojz KOPÁČIK, Ján ERDÉLYI and Richard HONTI, Slovakia

1. INTRODUCTION

With the increased availability of instruments needed for measurement, the popularity of point cloud usage is also growing. Point clouds can have an essential role in creating high-quality 3D models of objects in a variety of areas, e.g., interior (exterior) design, building information modelling (BIM), urban information systems, documentation of objects (Pavelka, 2021), 3D cadaster, deformation analysis (Štroner, 2018), (Marendić, 2017), etc. With the currently available technology, massive data sets (millions of points) can be collected relatively quickly and in a short time.

Identification and segmentation of geometric primitives in point clouds have been investigated for a long time. The methods and algorithms that have been suggested can generally be divided into five categories (Nguyen, 2013), (Grilli, 2017):

- Edge-based methods, based on detecting the boundaries of separate sections in a point cloud to obtain some segmented regions, for example (Nguyen, 2013), (Grilli, 2017), (Bhanu, 1986), and (Sappa, 2001)
- Region-based methods use neighbourhood information to merge close points with identical properties to obtain segregated regions and consequently find dissimilarities between separate parts (Vosselman, 2004), (Vo, 2015), and (Yuan, 2020)
- Attribute-based methods (alternatively clustering-based methods) consist of two separate steps. The first step is the attribute's computation; in the second step, the point cloud is clustered based on the characteristics computed, e.g., (Biosca, 2008) and (Murtivoso, 2019)
- Model-based methods use geometric shapes (e.g., planes, spheres, cylinders, and cones) to organise points. Points that have the same mathematical representation are grouped as one segment. Two of the algorithms most widely used in this category are the random sample consensus (RANSAC) (Fischler, 1981) and the Hough transform (HT) (Hough, 1962). Various modifications of the original RANSAC algorithm can be found in (Li, 2017), (Schnabel, 2007), and (Xu, 2022)
- Graph-based methods deal with point clouds in terms of a graph. For example, (Strom, 2010) extended a graph-based method to segment coloured 3D laser data. Other approaches for segmentation based on graph-based methods are presented in (Cheng, 2021), (Nurunnabi, 2012), (Luo, 2021), and (Pierdicca, 2020).

In recent years, numerous methods and approaches based on deep learning have been introduced for point cloud processing, e.g., 3D shape classification, 3D object detection and tracking, and 3D point cloud segmentation (Guo, 2020). The segmentation methods can generally be divided into four groups: projection-based, discretization-based, point-based, and hybrid. Some critical approaches are presented in (Charles, 2017a), Charles, 2017b), and (Mi, 2020).

In some shape segmentation approaches, the point cloud of each structural element is first manually separated from the point cloud of the whole scanned structure. This step significantly reduces the processing efficiency and increases the time required. This paper presents an algorithm capable of automated detection and segmentation of several geometric shapes at once without the necessity of pre-segmentation. The segmentation process is effective in the case of complex point clouds with uneven density and a large number of outliers and noise. Furthermore, the proposed seed point selection technique and validation steps minimise the results' dependency on the seed point's choice and the local surface characteristics in the neighbourhood of this point.

2. APPROACH FOR POINT CLOUD SEGMENTATION

The presented approach combines the modified RANSAC algorithm with the region-growing method and the seed point selection technique, proposed based on local normal vector variation for each shape type. Three types of shapes can be segmented with the algorithm presented: planes, spheres, and cylinders. In addition, the robust approach enables the automated segmentation of geometric shapes or their combination.

In the case of the segmentation of planes, spheres, and cylinders at once, the algorithm process is illustrated by the flowchart in Figure 1.

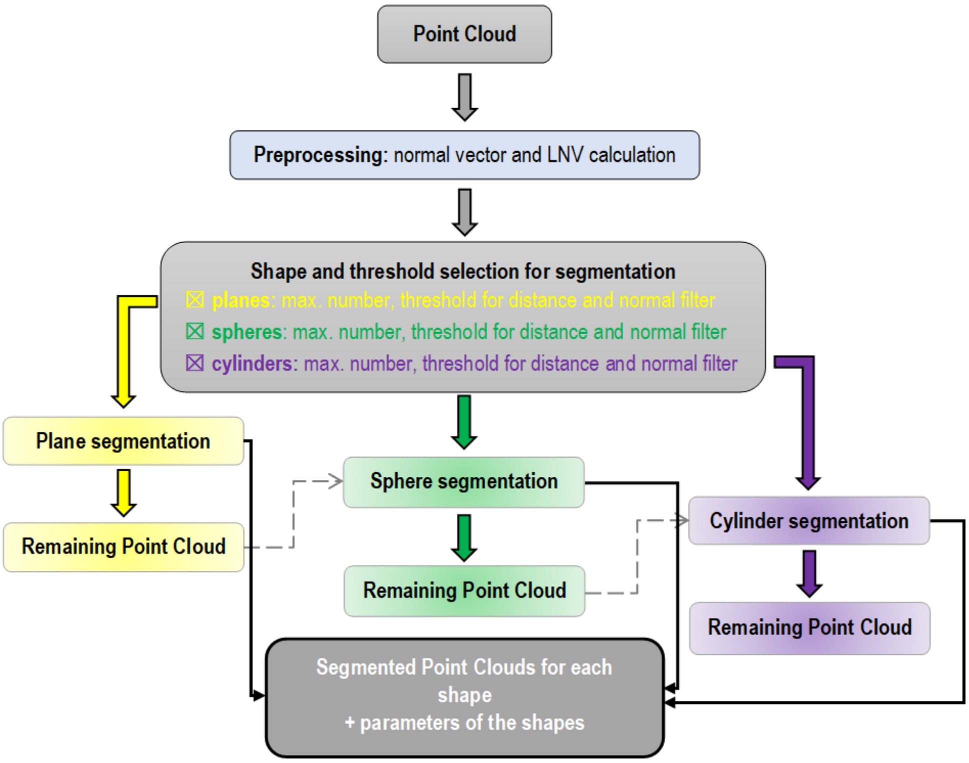

Point Cloud
Preprocessing: normal vector and LNV calculation
Shape and threshold selection for segmentation
☒ planes: max. number, threshold for distance and normal filter
☒ spheres: max. number, threshold for distance and normal filter
☒ cylinders: max. number, threshold for distance and normal filter
Plane segmentation
Remaining Point Cloud
Sphere segmentation
Remaining Point Cloud
Cylinder segmentation
Remaining Point Cloud
Segmented Point Clouds for each shape
+ parameters of the shapes

Figure 1 Flowchart of the algorithm (Honti, 2022)

Before the segmentation procedure itself, some preprocessing steps are performed. First, the normal vectors at each point of the point cloud are calculated using small local planes, calculated from the 3D coordinates of the given point and the k-nearest neighbours. Then, orthogonal regression is used for the estimation of local planes.

When the normal vectors are calculated, a new seed point selection technique is performed based on the local normal variation (LNV) value, which is determined as the average value of the scalar products of the normal vectors from the *k*-nearest points based on

$$\boldsymbol{LNV}_i = 1 - \frac{1}{n}\sum_{i=1}^{n} abs(\boldsymbol{dotNorm_i}) \qquad (1)$$

where $abs(\boldsymbol{dotNorm}_i)$ is the absolute value of the scalar product of the normal vector at a given point and the normal vectors at the neighbouring points, based on the LNV value it is possible to determine points where the occurrence of a planar surface is assumed or where a curved surface is expected to occur. The segmentation procedure is divided into three stages based on the shape types.

2.1 Plane segmentation

The algorithm for segmenting planes from the point cloud combines a modified RANSAC algorithm and the region-growing method. The plane segmentation starts with the selection of the seed points. The seed point candidates are determined based on the proposed seed point selection technique. The selected seed points are the points where the value of LNV is less than 1° (i.e., the orientation of the normal vector at the given point is approximately parallel to the normal vectors at the k-nearest neighbours). This seed point is used as a starting point for the plane estimation. The first plane is estimated using the n_{est} nearest points. The number (n_{est}) of the closest points depends on the local point density (LPD) at the selected seed point. This means choosing the points at a distance of 50 mm from the seed point. Orthogonal regression is used to estimate plane parameters, which minimises the orthogonal distances to the estimated plane. In the next step, the inliers for the given plane are identified (i.e., the points lying on the surface of the estimated plane) by testing the estimated regression plane against the nearest neighbours. Inlier selection is performed using two criteria:

- distance-based criterion: only the points that are closer to the estimated plane than the selected threshold values are considered inliers;
- normal-based criterion: inliers are the points where the angle between the normal vector at a given point and between the normal vector of the regression plane is less than the threshold value.

The plane re-estimation and the inlier selection are performed iteratively, gradually increasing the number of the neighbouring points tested. Then, it is repeated until the number of points belonging to the plane stops rising.

Since the plane estimation strongly depends on the seed point selection and its neighbourhood, it was necessary to introduce several validation steps to eliminate incorrect estimates. These validation steps are based first on determining whether there are enough inliers at each iteration, i.e., after the second iteration, at least two times more inlier points than the number of points from the first estimation (n_{est}). The number of inliers (points on the segmented surface) from the surroundings is expected to increase gradually if the selected seed point lies on a planar surface. Then, it is determined whether the plane has sufficient point coverage. The local point density (LPD) value is calculated at each inlier point. In addition, the theoretical value of the LPD is calculated (if the plane has a uniform ideal coverage). The criterion is that at least 50% of the points need a higher LPD (with a certain tolerance) than the ideal LPD. This value (50%) was determined empirically based on testing the algorithm on several point clouds with various densities, complexities, and different levels of noise. This criterion is necessary in some cases, e.g., when processing a point cloud from an indoor building environment with several objects (e.g., furniture, PC accessories, etc.). In such cases, a plane can be estimated from some subsets of points that are lying on different objects (not lying on the planar surface), i.e., the result of the estimation can be a plane (though this plane is not a real one), since these points are from a separate dense subset of points lying on a surface of any object. The mentioned cases are eliminated from the estimation by this criterion.

The number of searched entities (planes) in the point cloud has yet to be discovered in advance. Therefore, a technique is proposed to stop the calculation when all the planes located in the point cloud are segmented. The results of the plane segmentation are the segmented point clouds for each plane and the parameters of the planes, which are: the parameters of the normal vector of the plane (a, b, c), parameter d, the number of inliers, the standard deviation of the plane estimation (calculated from the orthogonal distances of the inlier points from the best-fit regression plane). After this part of the algorithm, further processing is performed only on the remaining point cloud (i.e., the points belonging to the segmented planes are excluded from the initial point cloud). This step indeed contributes to increasing the efficiency of the algorithm.

2.2 Sphere segmentation

The part of the algorithm with the sphere segmentation role is also partially inspired by the RANSAC algorithm. The least-square spherical fit is used to calculate the sphere parameter. The first step is the selection of seed points based on the LNV values at each point (seed point candidates for spheres are the points where the LNV value is greater than 5°). This seed point selection technique significantly increases the efficiency of the algorithm. In cases of processing complex point clouds that contain several walls (planar surfaces, where the points have slight local curvature – usually up to 5 degrees, because of undulation of the planar surface in some cases), with this step, the points lying on these surfaces are removed from the seed point candidates for sphere estimation. This technique is mainly for a rough removal of the points, where it is assumed that no sphere object can be found.

Then, the first approximate parameters of the sphere are calculated using the n_{est} number of nearest points to the selected seed point. The value of the n_{est} is calculated in the same way as in the case of planes. The estimation is based on a least-square spherical fit, which minimises the perpendicular distances of the points from the sphere. The iterative fitting and extraction process then uses the sphere's approximate estimated parameters. Thus, extracting the inliers for the estimated sphere candidate is performed based on distance-based and normal-based filtration (similar to the plane segmentation part). In contrast to the plane algorithm, the extraction process is performed on the whole point cloud at once. Finally, the iterative re-estimation is performed until all the points of the detected sphere have been selected. It means that if no new point is added in 3 consecutive iterations, and there is no difference in the parameters of the sphere, the calculation is automatically stopped.

From experiments (processing of several point clouds containing sphere objects), it was found that coverage of only approximately 40 – 50% of the scanned sphere surface (e.g., when scanning only from a single position of the instrument) is sufficient for extraction of the sphere with the algorithm depicted. However, a certain LPD is needed, i.e., at least 100 points must be homogeneously distributed on the scanned part of the sphere surface.

2.3 Cylinder segmentation

The most complex part of the segmentation process is cylinder segmentation. The algorithm developed belongs to model-based algorithms, based partially on the Hough transformation. The algorithm is applied to the remaining point cloud data sets. It starts with selecting seed points based on the calculated LNV values (the seed point candidates are the points where the LNV is greater than 3°). Next, the first cylinder is estimated from the n_{est} number of closest neighbours to the seed point. The nest's value is calculated the same way as in the case of planes and spheres. Finally, the estimated cylinder parameters are calculated (Figure 2).

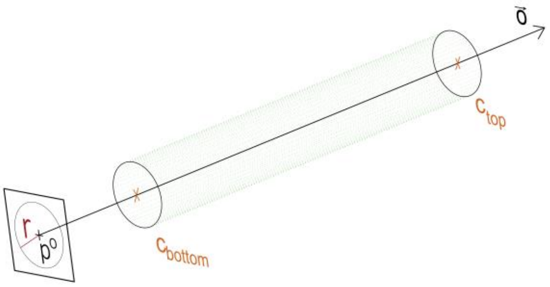


Figure 2 Cylinder parameters (Honti, 2022)

The estimation of these parameters is made in steps:

- Determination of the cylinder axis $\vec{\boldsymbol{o}}$, which is perpendicular to the normal vectors in neighbouring points
- Projection of these points into the plane, which is orthogonal to the cylinder axis
- Calculation of the circle parameters (coordinates of the centre and the radius) by fitting these points to the curve (minimal distance)
- Determination of the coordinates of the base centre points (bottom and top)

The estimation steps described above are applied iteratively for the inlier points. The set of points considered as inliers is updated at each iteration based on two criteria, mathematically formulated as follows:

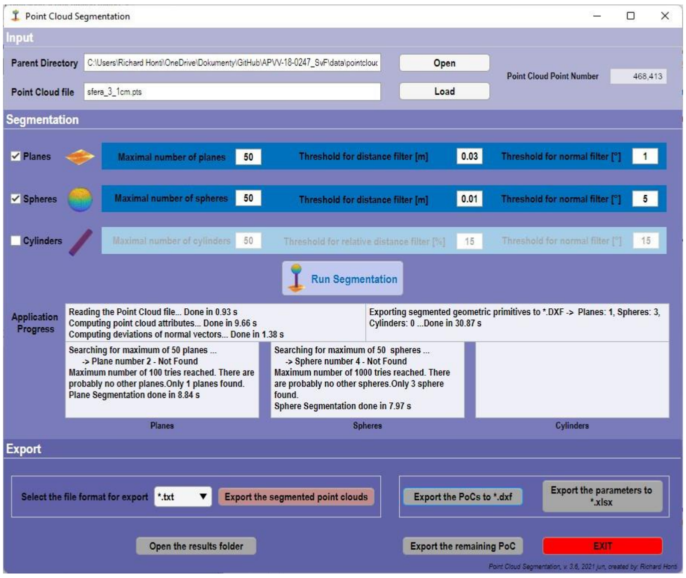
$$(|\Delta \boldsymbol{dist}_i| < r \cdot t_d[\%]) \wedge (|\Delta \boldsymbol{norm}_i| < t_n) \qquad (2)$$

where Δ***dist***$_i$ is the orthogonal distance between the selected point and the cylinder surface, and Δ***norm***$_i$ is the angle between the normal vector of the selected point and the vector perpendicular to the cylinder axis $\vec{\boldsymbol{o}}$ in the selected point.

The inliers are automatically updated in every iteration, and the outliers are removed from the estimations. After every iteration, the cylinder parameters are re-estimated using all the inliers that meet the specified conditions. The inlier updating is performed on the whole point cloud at once, similar to the sphere segmentation part. Then, the iterative re-estimation is performed until all the points of the detected cylinder have been selected. Similar to the plane and sphere segmentation part, the automatic stopping of the calculation is implemented in this part.

3. DEVELOPMENT OF THE SOFTWARE APPLICATION

To support the efficiency of the point cloud segmentation process, a standalone computational application (**PoCSegmentation**) was developed. The application's graphical user interface (GUI) was designed in MATLAB® software (Figure 3). Also, the calculation takes place in the MATLAB® software environment, so the Matlab Runtime is required to run the application, which is freely available. The dialogue window of the application (Figure 3) consists of three major sections. The top section is for importing the point cloud. The point cloud can be imported in several file formats, which are as follows: **.txt*, **.xyz*, **.pts*. **.pcd*, **.ply*, **.mat*.



Parent Directory: C:\Users\Richard Honti\OneDrive\Dokumenty\GitHub\APVV-18-0247_SvF\data\pointclouc

Point Cloud file: sfera_3_1cm.pts

Point Cloud Point Number: 468,413

☑ Planes — Maximal number of planes: 50 — Threshold for distance filter [m]: 0.03 — Threshold for normal filter [°]: 1

☑ Spheres — Maximal number of spheres: 50 — Threshold for distance filter [m]: 0.01 — Threshold for normal filter [°]: 5

□ Cylinders — Maximal number of cylinders: 50 — Threshold for relative distance filter [%]: 15 — Threshold for normal filter [°]: 15

```
Reading the Point Cloud file... Done in 0.93 s
Computing point cloud attributes... Done in 9.66 s
Computing deviations of normal vectors... Done in 1.38 s
```

```
Exporting segmented geometric primitives to *.DXF ->  Planes: 1, Spheres: 3,
Cylinders: 0 ...Done in 30.87 s
```

```
Searching for maximum of 50 planes ...
    -> Plane number 2 - Not Found
Maximum number of 100 tries reached. There are
probably no other planes.Only 1 planes found.
Plane Segmentation done in 8.84 s
```

```
Searching for maximum of 50  spheres ...
    -> Sphere number 4 - Not Found
Maximum number of 1000 tries reached. There
are probably no other spheres.Only 3 sphere
found.
Sphere Segmentation done in 7.97 s
```

Select the file format for export: *.txt

Figure 3 Dialog window of the software (Honti, 2022)

The middle part serves for the segmentation, where the types of geometric shapes can be chosen at the top, and the threshold parameters can be selected. The segmentation procedure is started by pressing the *Run Segmentation* button. At the bottom of this section, the application's progress is described, i.e., the individual processes executed step by step and the time required for its execution. Finally, the bottom part of the dialogue window is used to export the segmentation results.

The **PoCSegmentation** application offers various options to export the results. On the left, the segmented point clouds can be exported in several file formats (**.txt, *.pts, *.xyz, *.pcd, *.ply, *.mat*). In addition, it is also possible to export the segmented point clouds into *.DXF (Drawing Exchange Format) is a CAD data file format for vector graphics and can be imported into more than 25 applications from various software developers. The advantage of the application is that, in the DXF file, the individual segmented point clouds are divided into separate layers. Next, the parameters of the geometric shapes can be exported to an Excel file, and the remaining point cloud can also be exported to *.pts format.

4. RESULTS OF TESTS

Experimental testing was provided with the application, both using a scan of a model (laboratory conditions) and on several point clouds collected by scanning different types of artificial objects.

4.1 Model-based test

The laboratory conditions were represented by the model of a **double-cylinder-shaped test object** (Figure 4). The geometric parameters of the cylinder object were measured by a CMM (Coordinate Measuring Machine) measuring system with an accuracy of 0.1 mm. The measurement was performed with TLS Leica Scanstation 2 (average density 10 mm, the accuracy of a single measured point up to 2.5 mm, approximately 202 thousand points). Figure 4 shows the segmented planes and the segmented cylinders.

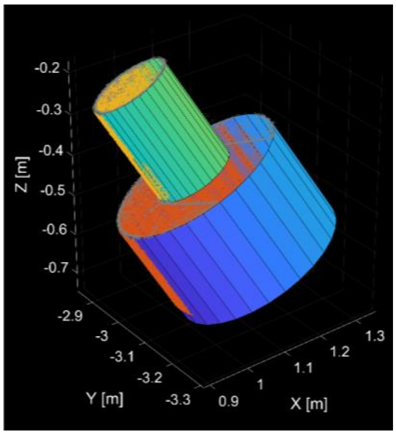


Figure 4 Segmented planes and cylinders (Honti, 2022)

The differences between the known geometric parameters (radius and height of the cylinders) and the estimated parameters from the processing were also calculated to verify the cylinder estimation (Table 1). The radius differences were 0.2 mm for the larger cylinder and 0.1 mm for the smaller cylinder. The height differences were 0.3 mm for the larger cylinder and 0.5 mm for the smaller cylinder.

Table 1

Cylinder	**Model parameters**		**Parameters determined by application**		**Absolute differences**	
	r_{mod} [m]	**h_{mod} [m]**	**r_{app} [m]**	**h_{app} [m]**	**Δr [mm]**	**Δh [mm]**
Large	0.200	0.250	0.202	0.253	0.2	0.3
Small	0.090	0.250	0.091	0.255	0.1	0.5

4.2 Test based on the real scene

The application was tested on a complex data set containing all types of geometric shapes. For this test, the point cloud of a part of the entire room of the **Pavol Országh Hviezdoslavov Theatre in Bratislava** was used, which contained more than 1.8 million points. Scanning was also performed with a Trimble TX5 3D laser scanner. The average point cloud density was 20 mm, and the accuracy in the spatial position of a single measured point was less than 5 mm. The initial point cloud from the two views is shown in Figure 5.

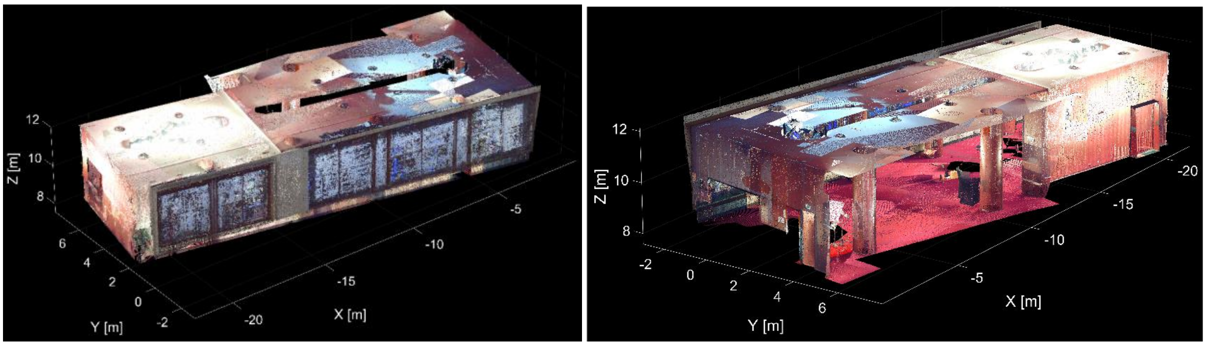


Figure 5 Initial point cloud of the theatre (Honti, 2022)

The scanned room contained 22 planar surfaces (walls, floor, etc.) and six cylinder-shaped columns. The algorithm identified and segmented all 22 planes and six cylinders from the point cloud. To verify the results of cylinder segmentation, the known (objective) parameters of individual cylinders were compared with the estimated parameters from processing the point cloud with the application developed. The cylindrical columns had a uniform radius (r_{real}) of 0.4 m and a height (h_{real}) of 3.9 m. The parameters of each of the columns were measured by a measuring tape at various positions, and the final parameters were calculated as average values.

The radiuses (r_{app}) and heights (h_{app}) obtained from the processing are shown in Table 2 with the calculated absolute differences.

Table 2

Cylinder	**Real parameters**		**Parameters determined by application**		**Absolute differences**		***s***
	r_{real} [m]	**h_{real} [m]**	**r_{app} [m]**	**h_{app} [m]**	**Δr [mm]**	**Δh [mm]**	**[mm]**
1	0.400	3.900	0.409	3.902	9	2	10
2			0.408	3.909	8	9	15
3			0.406	3.895	6	5	12
4			0.395	3.892	5	8	11
5			0.394	3.899	6	1	13
6			0.402	3.906	2	6	15

The maximal difference in radiuses was 9 mm, and in height was 9 mm. These differences include the imperfection of the construction of these columns, the effect of the environmental conditions, the systematic errors of the instrument, the measurement error, and the processing errors. The standard deviation in Table 2 describes the quality of the cylinder fitting to the segmented points, which was 10 to 15 mm.

Finally, using the proposed application, all the planes and cylinders that form the room's structural elements (walls, columns, etc.) were correctly and automatically detected and segmented from the point cloud.

4.3 Comparison with the standard RANSAC algorithm

A comparison with the standard RANSAC algorithm was performed on both datasets to demonstrate the potentiality and advantages of the proposed approach. First, the comparison is executed for each geometric shape type separately. Then, the geometric shape segmentation process is performed 50 times for each dataset with the same parameters, and the best results are compared. Based on the experiments presented, the proposed algorithm shows better results than the standard RANSAC algorithm. Table 2 brings an example of cylinder segmentation.

Table 3

	Point cloud - theatre		**Point cloud - model**	
	Proposed algorithm	**RANSAC algorithm**	**Proposed algorithm**	**RANSAC algorithm**
$cyl_{correct}$	6	6	2	2
$cyl_{uncorrect}$	0	7	0	3
$cyl_{undetected}$	0	0	0	0
time	5 min 22 s	6 min 5 s	0 min 30 s	0 min 25 s

On average, the segmentation quality was less than 50% in the case of the standard RANSAC algorithm. In the case of the proposed algorithm, it was mostly 100% (only one time, in the case of the first plane comparison, it was 92% since the roof of the building was divided into two planar surfaces, due to its undulation, and so it was not caused by the imperfection of the proposed algorithm). Furthermore, the standard deviation of the fitting is ten times lower on average using the proposed algorithm since the outlier, noise removal process is more thorough, and a normal-based filtration technique is added to the distance-based filtration. For more details, see the publication (Honti, 2022).

Moreover, several validation steps are executed to remove incorrectly detected shapes from the results. Furthermore, in the case of the proposed algorithm, there is no need to select the number of geometric shapes in the point cloud exactly since an approach was proposed to stop the calculations after the correctly segmented bodies. The proposed algorithm's advantage is the possibility to segment three shape types at once in a semi-automated way.

5. CONCLUSION

Data collection can be performed relatively quickly and in a short time today. However, manually processing the measured point clouds can be time-consuming and complicated. Therefore, a basic premise of the efficiency of using point clouds is a high degree of automation of the processing steps. For example, when creating a 3D model (or BIM) of an existing building, one of the basic steps is the identification and segmentation of the essential structural elements of the object. Usually, these basic elements are formed in the shape of basic geometric primitives (e.g., walls – planes; columns or piping network – cylinders; etc.). Therefore, automation of identification and segmentation of sphere objects can be helpful, for example, in the case of point cloud registration based on spherical targets. It also simplifies 3D model creation.

The paper presents the algorithm proposed for the automated identification and segmentation of geometric shapes from point clouds with the requirement of selecting a minimal number of input parameters. The algorithm can detect end-segment subsets of points belonging to planes, spheres, and cylinders from complex, noisy, unstructured point clouds. Inlier detection is performed using distance-based and normal-based filtering. Additionally, several validation steps were proposed to eliminate incorrect estimations.

The algorithm proposed was tested on several point clouds with various densities, complexity, and different levels of noise. Specifically, testing on two different point clouds was described. In both cases, the proposed algorithm correctly identified the geometric shapes regardless of size, number, or complexity. Moreover, one of the most significant advantages of the algorithm is that the results can be exported directly to DXF exchange format for further processing. Besides that, comparison between the proposed algorithm and the standard RANSAC algorithm was performed separately for the individual geometric shapes on both point clouds. On average, the segmentation quality was increased from 50% to 100% with the described algorithm.

REFERENCES

Bhanu, B.; Lee, S.; Ho, C.C.; Henderson, T. Range Data Processing: Representation of Surfaces by Edges. In Proceedings of Eighth International Conference on Pattern Recognition, Paris, France, 27–31 October 1986; pp. 236–238.

Biosca, J.M.; Lerma, J.L. Unsupervised robust planar segmentation of terrestrial laser scanner point clouds based on fuzzy clustering methods. *ISPRS J. Photogramm. Remote Sens.* **2008**, *63*, 84–98. https://doi.org/10.1016/j.isprsjprs.2007.07.010.

Charles, R.; Su, H.; Kaichun, M.; Guibas, L. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017; pp. 77–85. https://doi.org/10.1109/CVPR.2017.16.

Charles, R.Q.; Yi, L.; Su, H.; Guibas, L.J. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17), Long Beach, CA, USA, 4–9 December 2017; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 5105–5114.

Cheng, M.; Hui, L.; Xie, J.; Yang, J. SSPC-Net: Semi-supervised Semantic 3D Point Cloud Segmentation Network. In Proceedings of the AAAI Conference on Artificial Intelligence, 2–9 February 2021; Volum 35, pp. 1140–1147.

Fischler, M.A.; Bolles, R.C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **1981**, *24*, 381–395.

Grilli, E.; Menna, F.; Remondino, F. A Review of Point Clouds Segmentation and Classification Algorithms. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2017**, *42W3*, 339–344. https://doi.org/10.5194/isprs-archives-XLII-2-W3-339-2017.

Guo, Y.; Wang, H.; Hu, Q.; Liu, H.; Liu, L.; Bennamoun, M. Deep Learning for 3D Point Clouds: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 4338–4364. https://doi.org/10.1109/TPAMI.2020.3005434.

Honti, R.; Erdélyi, J.; Kopáčik, A. Semi-Automated Segmentation of Geometric Shapes from Point Clouds. *Remote Sens.* **2022**, *14*, 4591. https://doi.org/10.3390/rs14184591

Hough, P.V.C. Method and Means for Recognizing Complex Patterns. U.S. Patent 3069654, 18 December 1962.

Li, L.; Yang, F.; Zhu, H.; Li, D.; Li, Y.; Tang, L. An Improved RANSAC for 3D Point Cloud Plane Segmentation Based on Normal Distribution Transformation Cells. *Remote Sens.* **2017**, *9*, 433. https://doi.org/10.3390/rs9050433.

Luo, N.; Yu, H.; Huo, Z.; Liu, J.; Wang, Q.; Xu, Y.; Gao, Y. KVGCN: A KNN Searching and VLAD Combined Graph Convolutional Network for Point Cloud Segmentation. *Remote Sens.* **2021**, *13*, 1003. https://doi.org/10.3390/rs13051003.

Marendić, A.; Paar, R.; Tomić, H.; Roić, M.; Krkač, M. Deformation monitoring of Kostanjek landslide in Croatia using multiple sensor networks and UAV. In Proceedings of the INGEO 2017–7th International Conference on Engineering Surveying, Lisbon, Portugal, 18–20 October 2017.

Mi, Z.; Luo, Y.; Tao, W. SSRNet: Scalable 3D Surface Reconstruction Network. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 967–976. https://doi.org/10.1109/CVPR42600.2020.00105.

Murtiyoso, A.; Grussenmeyer, P. Point cloud segmentation and semantic annotation aided by gis data for heritage complexes, *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* **2019**, *XLII-2/W9*, 523–528. https://doi.org/10.5194/isprs-archives-XLII-2-W9-523-2019.

Nguyen, A.; Le, B. 3D Point Cloud Segmentation: A survey. In Proceedings of the 6th IEEE Conference on Robotics, Automation and Mechatronics (RAM), Manila, Singapore, 12–15 November 2013; pp. 225–230, ISBN 978-1-4799-1201-8. https://doi.org/10.1109/RAM.2013.6758588.

Nurunnabi, A.; Belton, D.; West, G. Robust Segmentation in Laser Scanning 3D Point Cloud Data. In Proceedings of International Conference on Digital Image Computing Techniques and Applications (DICTA), Fremantle, WA, Australia, 3–5 December 2012. https://doi.org/10.1109/DICTA.2012.6411672.

Pavelka, K.; Matoušková, E.; Pavelka, K.; Pacina, J. Spatial 3D documentation of historical mining remnants in forested area in the Erzgebirge/Krušnohoří mining region UNESCO site. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2021**, *XLVI-M-1-2021*, 523–529. https://doi.org/10.5194/isprs-archives-XLVI-M-1-2021-523-2021.

Pierdicca, R.; Paolanti, M.; Matrone, F.; Martini, M.; Morbidoni, C.; Malinverni, E.S.; Frontoni, E.; Lingua, A.M. Point Cloud Semantic Segmentation Using a Deep Learning Framework for Cultural Heritage. *Remote Sens.* **2020**, *12*, 1005. https://doi.org/10.3390/rs12061005.

Sappa, A.D.; Devy, M. Fast range image segmentation by an edge detection strategy. In Proceedings of Third International Conference on 3-D Digital Imaging and Modeling, Quebec City, QC, Canada, 28 May–1 June 2001; IEEE: New York, NY, USA, pp. 292–299.

Schnabel, R.; Wahl, R.; Klein, R. Efficient RANSAC for point-cloud shape detection. *Comput. Graph. Forum* **2007**, *26*, 214–226. https://doi.org/10.1111/j.1467-8659.2007.01016.x.

Strom, J.; Richardson, A.; Olson, E. Graph-based segmentation for colored 3D laser point clouds. In Proceedings of The 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, 18–22 October 2010; pp. 2131–2136. https://doi.org/10.1109/IROS.2010.5650459.

Štroner, M.; Urban, R.; Křemen. T.; Koska, B. Accurate Measurement of the Riverbed Model for Deformation Analysis using Laser Scanning Technology. *Geoinform. FCE CTU* **2018**, *17*, 81–92. https://doi.org/10.14311/gi.17.2.5.

Vosselman, G.; Gorte, B.G.H.; Sithole, G.; Rabbani, T. Recognising Structure in Laser Scanner Point Clouds. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2004**, *46 Pt 8/W2*, 33–38.

Vo, A.-V.; Truong-Hong, L.; Laefer, D.; Bertolotto, M. Octree-based region growing for point cloud segmentation. *ISPRS J. Photogramm. Remote Sens.* **2015**, *104*, 88–100. https://doi.org/10.1016/j.isprsjprs.2015.01.011.

Xu, B.; Chen, Z.; Zhu, Q.; Ge, X.; Huang, S.; Zhang, Y.; Liu, T.; Wu, D. Geometrical Segmentation of Multi-Shape Point Clouds Based on Adaptive Shape Prediction and Hybrid Voting RANSAC. *Remote Sens.* **2022**, *14*, 2024. https://doi.org/10.3390/rs14092024.

Yuan, H.; Sun, W.; Xiang, T. Line laser point cloud segmentation based on the combination of RANSAC and region growing. In Proceedings of the 2020 39th Chinese Control Conference (CCC), Shenyang, China, 27–29 July 2020; pp. 6324–6328. https://doi.org/10.23919/CCC50068.2020.9188506.

FUNDING

This work was supported by the Slovak Research and Development Agency under Contract no. APVV-18-0247.

BIOGRAPHICAL NOTES

Alojz Kopáčik, Ph.D. is a Professor at the Slovak University of Technology, in the Department of Surveying. He has been lecturing on Geodesy for CE, Underground and Mine Surveying and Engineering Surveying, Measurement systems in engineering surveying and Surveying for Civil Engineering (in English). His research is focused on the field of TLS applications, automated measuring systems, calibration, and standardisation in geodesy and cartography.
Past Chair of FIG C6, delegate national for FIG C2 (Education). President of the Slovak Chamber of Surveyors and Cartographers. Member of the board of several journals and WGs of FIG and IAG. Member of ISO TC172 SC6 Geodetic and surveying instruments. Chairman of the TC for standardisation TC89 Geodesy and cartography (Slovakia).

Ján Erdélyi, Ph.D., is an Associate Professor at the Department of Surveying at the Slovak University of Technology in Bratislava. He has lectured on Engineering Surveying and Engineering Surveys for Industry since 2009. He worked on the Habilitation on Use of

Terrestrial Laser Scanning in Construction and Industry in 2018 at the STU in Bratislava, and he performs research activities in the field of TLS, Deformation Analysis, and Building Information Modeling. In addition, he is a co-investigator of many research and commercial projects aimed at different areas of engineering surveying.

Richard Honti, Ph.D. is a Lecturer at the Slovak University of Technology in Bratislava, Department of Surveying. Lectures from Geodesy for Civil Engineers, Industrial Surveying, Surveying for Civil Engineering (in English) and Field Courses on Engineering Surveying. PhD at the Department of Surveying the SUT Bratislava from 2017-21. Publications in various journals and conference proceedings.
Research in the field of TLS applications and automated point cloud processing.

CONTACTS

Prof. Alojz Kopáčik, PhD.
Department of Surveying of the Slovak University of Technology
Radlinského 11
810 05 Bratislava
SLOVAKIA
Phone: +421 2 3288 8559
Email: alojz.kopacik@stuba.sk
Web site: https://www.svf.stuba.sk/en/kgde.html?page_id=3486

Assoc. Prof. Ján Erdélyi, PhD.
Department of Surveying of the Slovak University of Technology
Radlinského 11
810 05 Bratislava
SLOVAKIA
Phone: +421 2 3288 85390
Email: jan.erdelyi@stuba.sk
Web site: https://www.svf.stuba.sk/en/kgde.html?page_id=3486

Richard Honti, PhD.
Department of Surveying of the Slovak University of Technology
Radlinského 11
810 05 Bratislava
SLOVAKIA
Phone: +421 2 3288 8391
Email: richard.honti@stuba.sk
Web site: https://www.svf.stuba.sk/en/kgde.html?page_id=3486