CLASSIFYING LIDAR POINT CLOUD USING BENTLEY MICROSTATION TERRASOLID AND ARCGIS PRO SOFTWARE PRODUCTS

Anca Patricia GRĂDINARU, PhD. Student Eng., Technical University of Civil Engineering Bucharest, Faculty of Geodesy, Romania, anca.patricia2611@gmail.com
Ana Cornelia BADEA, Prof. PhD. Eng., Technical University of Civil Engineering Bucharest, Faculty of Geodesy, Romania, anacorneliabadea@gmail.com, ana.badea@utcb.ro
Petre Iuliu DRAGOMIR, Prof. PhD. Eng., Technical University of Civil Engineering Bucharest, Faculty of Geodesy, Romania, petreiuliu.dragomir@gmail.com

Abstract: *In order to obtain accurate 3D City Models, some of the most important steps are geospatial data acquisition and the processing of this data. Aerial laser scanning (LiDAR-Light Detection and Ranging) is one of the most used methods of data acquisition and the main concern is classifying the point cloud in order to assign a class to each point. The classes of interest for 3D city models are the ground, buildings and possibly the high vegetation class which may interfere with the buildings. In this article, we aimed to highlight the main steps that are necessary to classify LiDAR point clouds. We chose two software products, one developed by Bentley – Microstation Terrasolid Suite and the second by Esri – ArcGIS Pro.*

Keywords: *LiDAR, Point Cloud, Classification*

1. Introduction

A 3D city model is a 3D representation of an urban area focused especially on the representation of buildings [1]. Automatic point cloud classification is challenging due to the large number of object classes and the complex structures of the objects.

Terrasolid Suite is a Bentley Microstation software used for point clouds and images processing, composed of four main modules: TerraScan, TerraModeler, TerraMatch and TerraPhoto and it provides multiple tools, including point cloud classification tools [11].

In this study we are intending to use the Bentley Microstation TerraScan module and ArcGIS Pro to classify a LiDAR point cloud, focusing on the ground, buildings and high vegetation classes.

2. The State of the Art

Recent research in the field is based in particular on machine learning classification, which involves the manual creation of some features and automatic classification using algorithms such as Support Vector Machine (SVM), Adaboost, Random Forest, Markov Random Field or Conditional Random Field [8].

Traditionally, in previous studies, point cloud classification is achieved by manually calculating some features and implementing different algorithms such as SVM to classify segments based on features related to radiometry, geometry, topology, or AdaBoost to classify 3D LiDAR data into four categories: buildings, trees, low vegetation and roads. These models, however, ignore the contextual information of point clouds and mark each point individually, resulting in classification inconsistency.

To solve this problem, the CRF (Conditional Random Field) method, which can include contextual information and the relationships between objects, was introduced to classify the point cloud based on geometric and intensity features. Also, recent researches discuss the very efficient use of graphical convolutional neural networks in point cloud classification, based on advanced deep learning methods [13].

Analyses regarding various methods of point cloud classification were conducted by Duran et al [7], who used the Cloud Compare software to calculate geometric features for both LiDAR point cloud and photogrammetric point cloud and compared the classification results of various machine learning algorithms (LR-Logistic Regression, LDA-Linear Discriminant Analysis, K-NN-K-Nearest Neighbours, DTC-Decision Tree Classifier, GNB-Gauss Naïve Bayes, MLP-Multilayer Perceptron, ADB-Adaboost, RF-Random Forest, SVM-Support Vector Machines), Cai et al [2], who compared object classification methods of airborne LiDAR point clouds and remote sensing images such as RF, DT and SVM and classified the points into four categories: trees, houses and buildings, low-growing vegetation and imprevious surfaces, Morsy and Shaker [9], who applied a classification method based on RF algorithm to classify a 3D point cloud collected by a TLS (Terrestrial Laser Scanner) System and Diab et al [6], who summarised the main deep learning classification models.

3. Materials and Methods

The area of study is located in Baia Mare city, Romania (Figure 1).

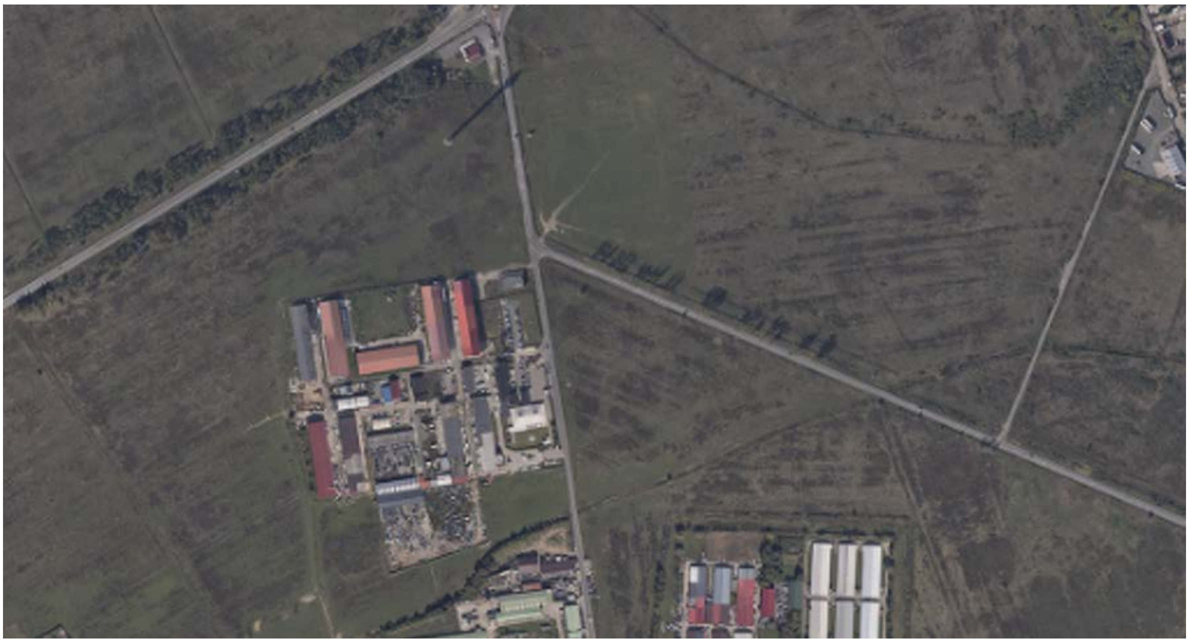
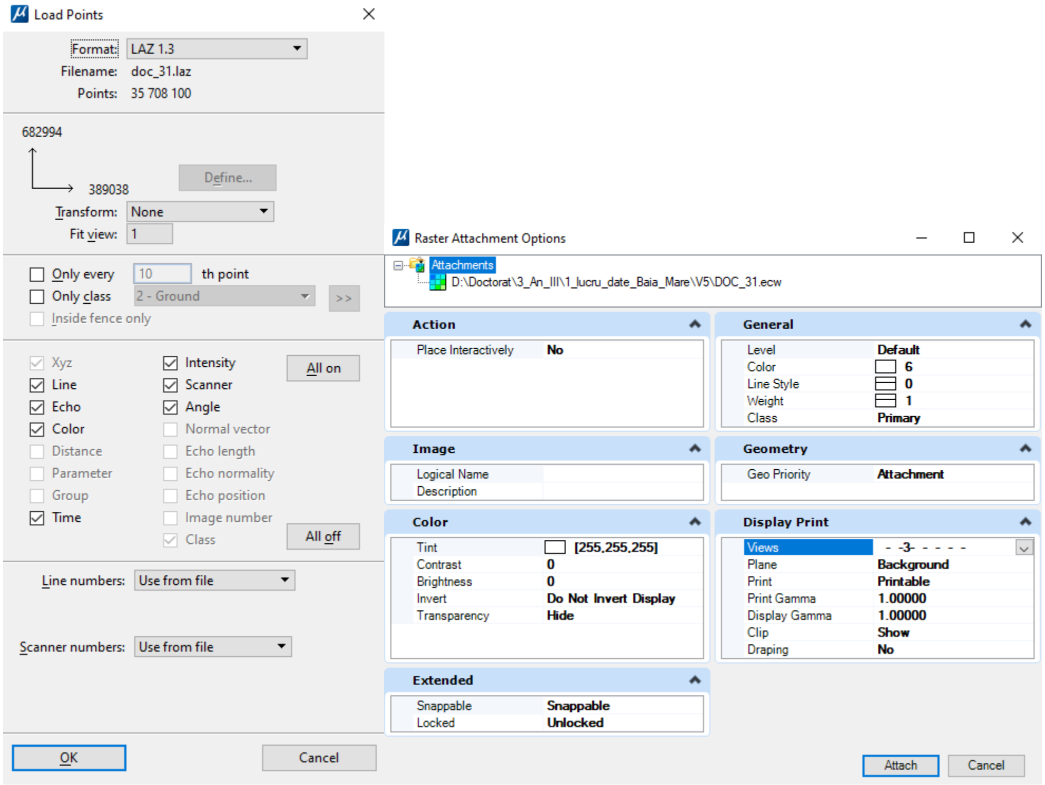


Figure 1. Area of Study

The first steps were to import a *.laz format file, containing 35.708.100 points in Bentley Microstation and to attach the image associated with the area of the point cloud (Figure 2).



Load Points

Format:	LAZ 1.3
Filename:	doc_31.laz
Points:	35 708 100

682994
389038
Define...

Transform:	None
Fit view:	1

☐ Only every 10 th point
☐ Only class 2 - Ground >>
☐ Inside fence only

☑ Xyz	☑ Intensity	All on
☑ Line	☑ Scanner	
☑ Echo	☑ Angle	
☑ Color	☐ Normal vector	
☐ Distance	☐ Echo length	
☐ Parameter	☐ Echo normality	
☐ Group	☐ Echo position	
☑ Time	☐ Image number	
	☑ Class	All off

Line numbers: Use from file

Scanner numbers: Use from file

OK Cancel

Raster Attachment Options

Attachments
D:\Doctorat\3_An_III\1_lucru_date_Baia_Mare\V5\DOC_31.ecw

Action

Place Interactively	No

General

Level	Default
Color	6
Line Style	0
Weight	1
Class	Primary

Image

Logical Name	
Description	

Geometry

Geo Priority	Attachment

Color

Tint	[255,255,255]
Contrast	0
Brightness	0
Invert	Do Not Invert Display
Transparency	Hide

Display Print

Views	- -3- - - - -
Plane	Background
Print	Printable
Print Gamma	1.00000
Display Gamma	1.00000
Clip	Show
Draping	No

Extended

Snappable	Snappable
Locked	Unlocked

Attach Cancel

Figure 2. Importing Data

In order to set up an efficient working space, we used three synchronized views, one for the top view of the point cloud, one for the lateral view and for vertical sections views and one for the image view.

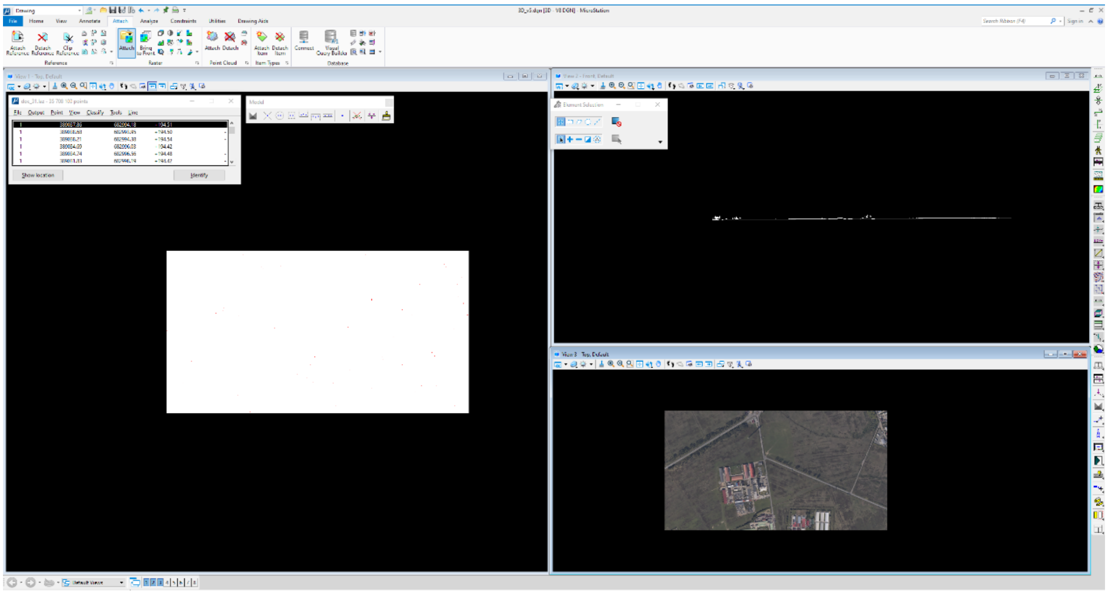
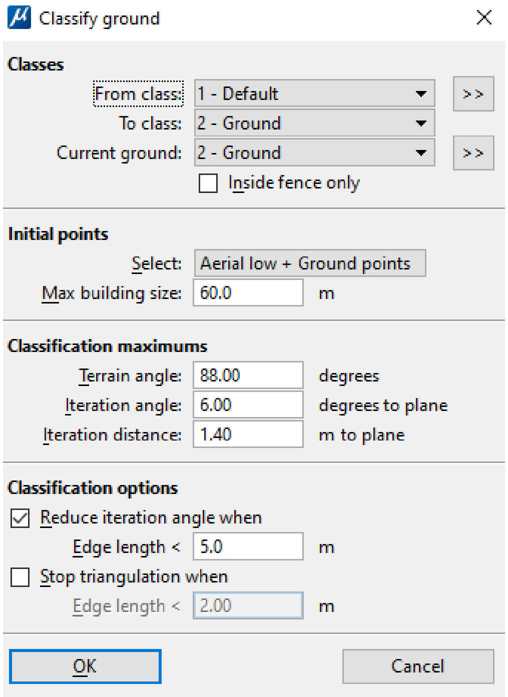


Figure 3. Capture from Bentley Microstation – 3 Views Workspace

The first classification we wanted to perform was the Ground points classification, as the Ground class is necessary for any other classification. The Ground routine classifies the points by creating a triangulated surface model iteratively and is sensitive to low error points in the point cloud, which is why a Low points classification routine had to be used before [10].

After setting the Ground classification parameters (Figure 4), we ran the classification routine and obtained a number of 3.505.724 ground points (Figure 5).



Classify ground

Classes

From class: 1 - Default
To class: 2 - Ground
Current ground: 2 - Ground
□ Inside fence only

Initial points

Select: Aerial low + Ground points
Max building size: 60.0 m

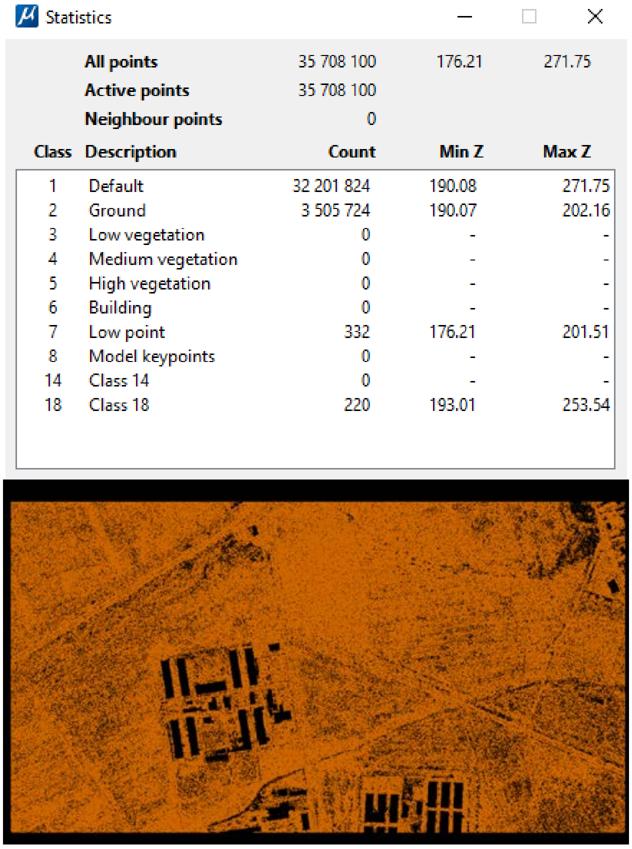
Classification maximums

Terrain angle: 88.00 degrees
Iteration angle: 6.00 degrees to plane
Iteration distance: 1.40 m to plane

Classification options

☑ Reduce iteration angle when
Edge length < 5.0 m
□ Stop triangulation when
Edge length < 2.00 m

Figure 4. Setting the Parameters for the Ground Points Classification



Statistics

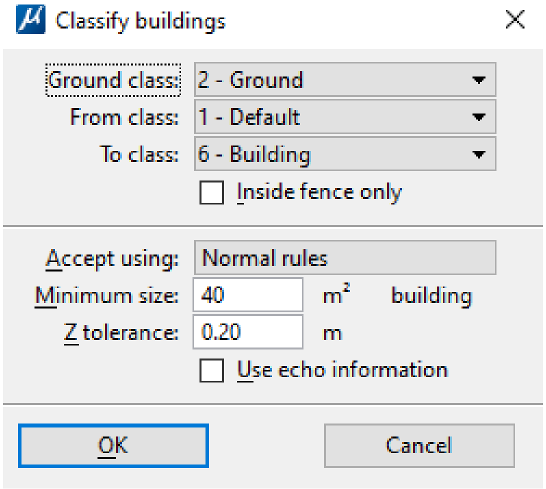
	All points	35 708 100	176.21	271.75
	Active points	35 708 100		
	Neighbour points	0		

Class	Description	Count	Min Z	Max Z
1	Default	32 201 824	190.08	271.75
2	Ground	3 505 724	190.07	202.16
3	Low vegetation	0	-	-
4	Medium vegetation	0	-	-
5	High vegetation	0	-	-
6	Building	0	-	-
7	Low point	332	176.21	201.51
8	Model keypoints	0	-	-
14	Class 14	0	-	-
18	Class 18	220	193.01	253.54

Figure 5. Results from the Ground Points Classification

Next classification we performed was the Building points classification. For this classification, a prior classification of the ground points is mandatory. The Buildings routine classifies points that form a planar surface and represent the building roofs and it starts from empty areas on the ground class, finding points on planar surfaces above them [10].

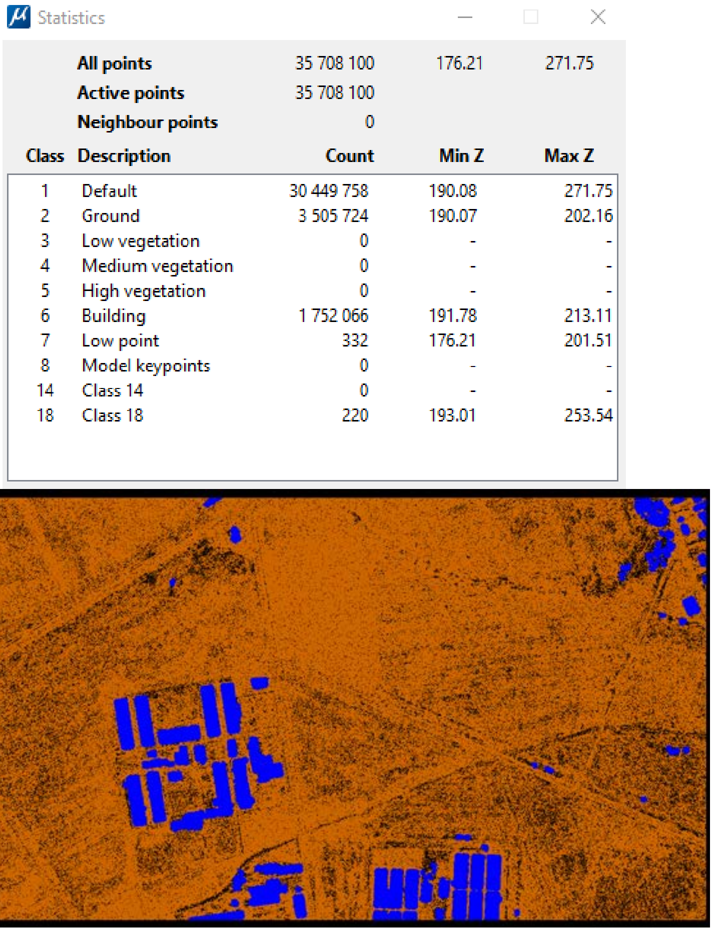
After setting the classification parameters (Figure 6) we ran the classification routine and obtained 1.752.066 points in the buildings class (Figure 7).


Classify buildings

Ground class: 2 - Ground
From class: 1 - Default
To class: 6 - Building
□ Inside fence only

Accept using: Normal rules
Minimum size: 40 m² building
Z tolerance: 0.20 m
□ Use echo information

Figure 6. Buildings Classification Parameters


Statistics

	All points	35 708 100	176.21	271.75
	Active points	35 708 100		
	Neighbour points	0		
Class	**Description**	**Count**	**Min Z**	**Max Z**
1	Default	30 449 758	190.08	271.75
2	Ground	3 505 724	190.07	202.16
3	Low vegetation	0	-	-
4	Medium vegetation	0	-	-
5	High vegetation	0	-	-
6	Building	1 752 066	191.78	213.11
7	Low point	332	176.21	201.51
8	Model keypoints	0	-	-
14	Class 14	0	-	-
18	Class 18	220	193.01	253.54

Figure 7. Buildings Classification Results

One of the main issues we have encountered was the fact that the software classified points around the buildings in the same class as the buildings (Figure 8). This issue had to be resolved manually.

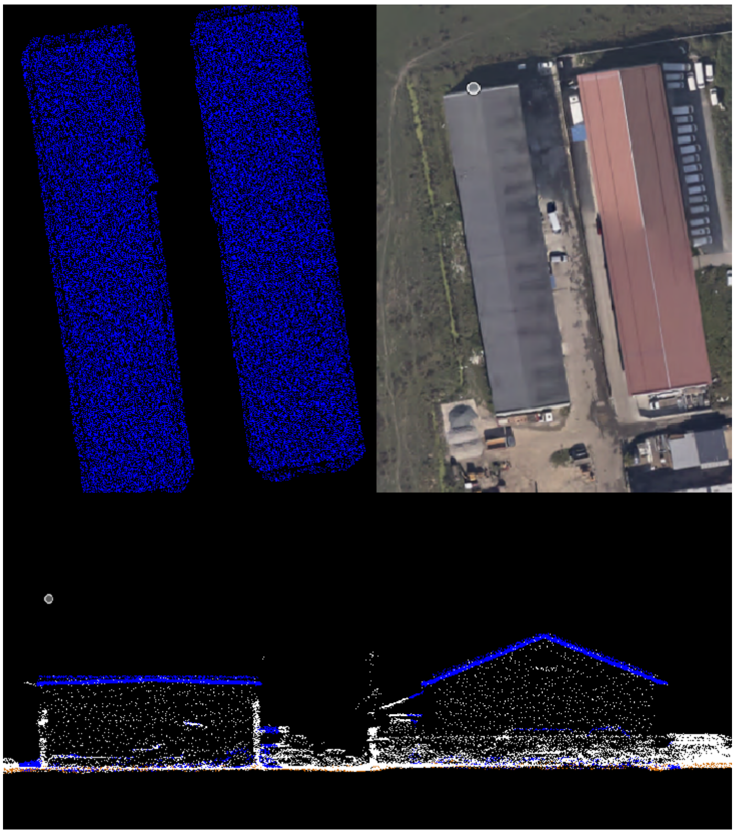


Figure 8. Building Classification Errors

Another issue was that the routine was unable to identify the buildings` walls (Figure 9). This was also addressed manually, by using the "classify above line" tool.

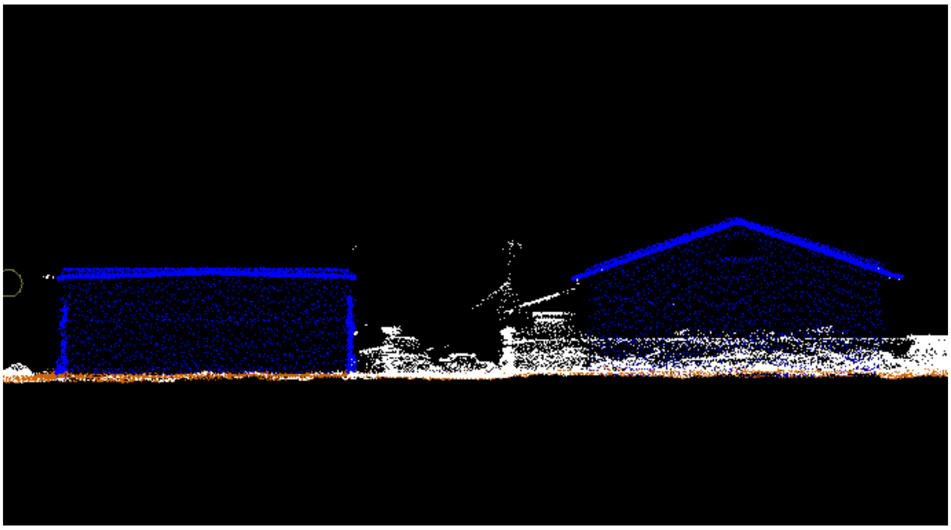


Figure 9. Manually Classified Building Walls

Some vegetation elements were mistakenly classified as buildings (Figure 10) and had to be manually moved back to the default class.

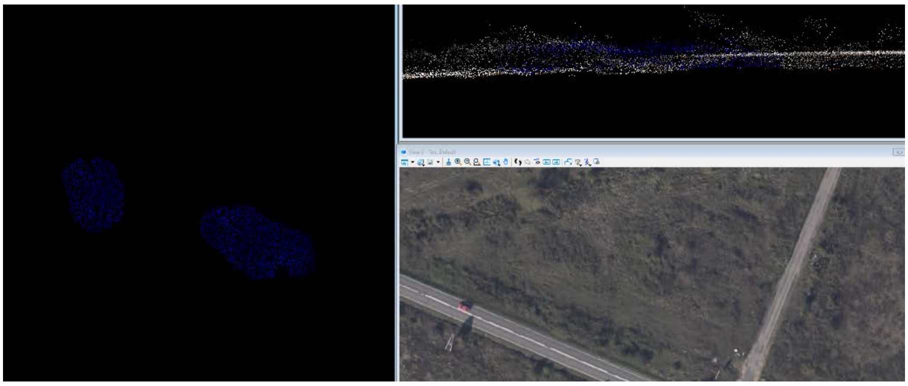


Figure 10. Vegetation Elements Classified as Buildings

In case the building's rooftop had any see-through or empty spots, points were collected through them and were classified as buildings (Figure 11). These also had to be manually moved to the default class.

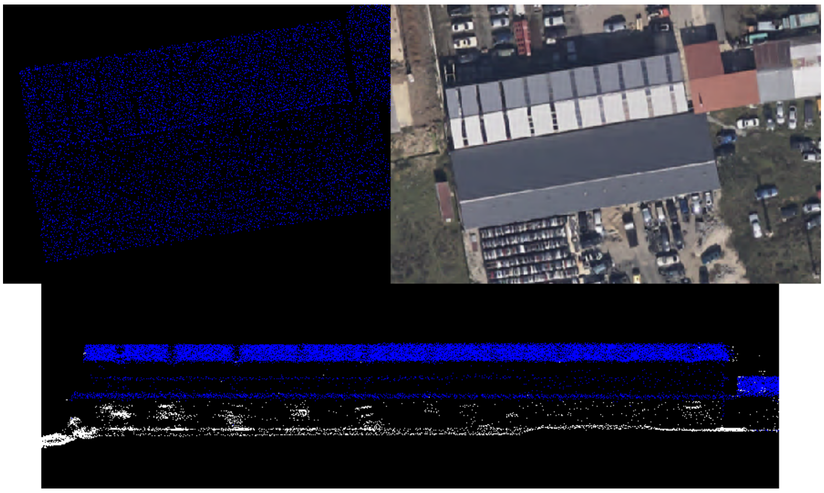


Figure 10. Building with See-through Portions of the Roof

Another important issue was that some buildings were partially covered by vegetation (Figure 11) and points from the roof were missing. This issue may be resolved by choosing to collect data during winter, when there is almost no vegetation.

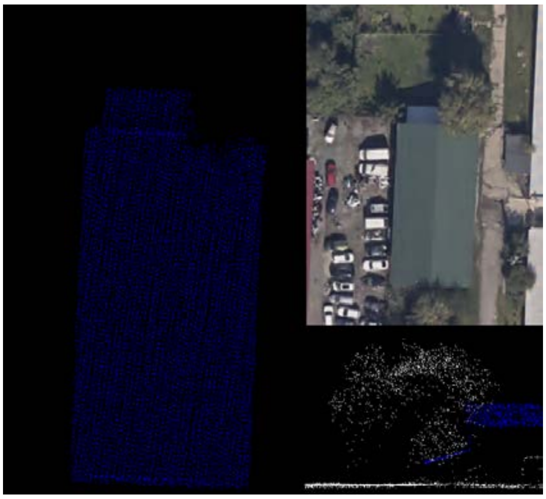
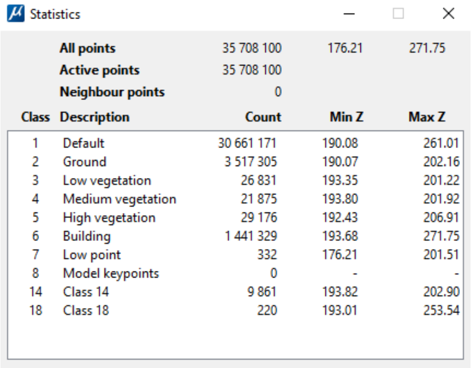


Figure 11. Building Covered by Vegetation

After manually correcting the classification results, we obtained a number of 1.441.329 points in the building class (Figure 12).



		Count	Min Z	Max Z
	All points	35 708 100	176.21	271.75
	Active points	35 708 100		
	Neighbour points	0		
Class	Description	Count	Min Z	Max Z
1	Default	30 661 171	190.08	261.01
2	Ground	3 517 305	190.07	202.16
3	Low vegetation	26 831	193.35	201.22
4	Medium vegetation	21 875	193.80	201.92
5	High vegetation	29 176	192.43	206.91
6	Building	1 441 329	193.68	271.75
7	Low point	332	176.21	201.51
8	Model keypoints	0	-	-
14	Class 14	9 861	193.82	202.90
18	Class 18	220	193.01	253.54

Figure 12. Building Classification Results after Manually Correcting the Classification

By comparing the results in Figure 7 with the final building classification results in Figure 13, we can notice that in the latter the buildings have more regular shapes.

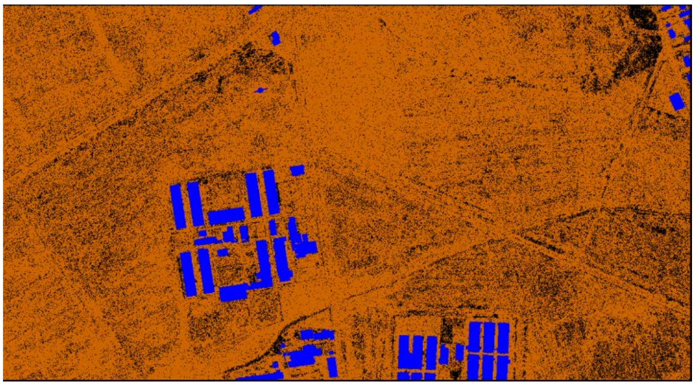
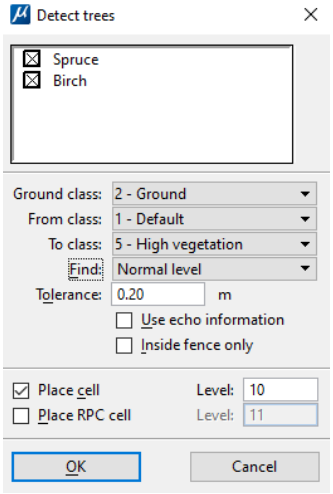


Figure 13. Final Results of the Buildings Classification

Another classification tool we used was the “Detect trees” routine. After setting the parameters (Figure 14), we ran the classification tool and obtained a number of 1.369.291 points in the High vegetation class.



☒ Spruce
☒ Birch

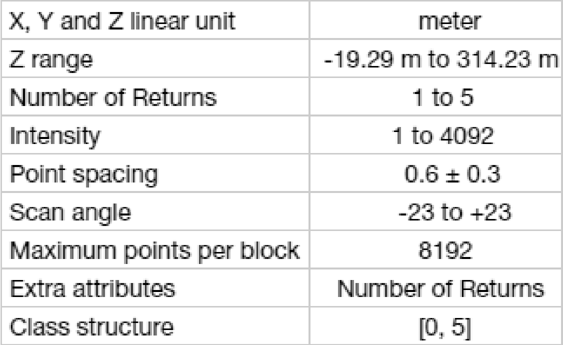
Ground class: 2 - Ground
From class: 1 - Default
To class: 5 - High vegetation
Find: Normal level
Tolerance: 0.20 m
□ Use echo information
□ Inside fence only

☑ Place cell Level: 10
□ Place RPC cell Level: 11

Figure 14. Setting the Parameters for the “Detect trees” Routine

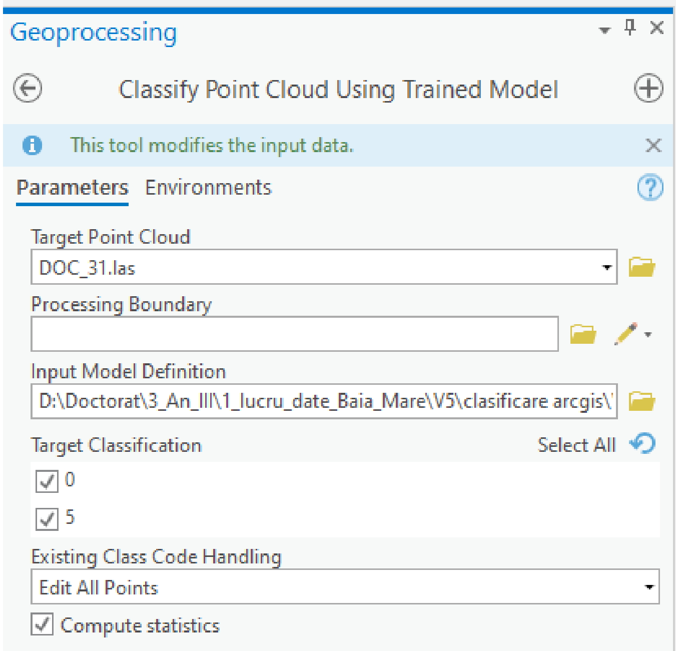
In order to classify the LiDAR point cloud using ArcGIS Pro, we used classification tools such as *Classify Point Cloud Using Trained Model*, for the High Vegetation class, *Classify LAS Ground* for the Ground class and *Classify LAS Building* for the Building class and a *.LAS file containing the 35.708.100 points.

The Classify Point Cloud Using Trained Model tool classifies a point cloud using a PointCNN classification model and requires an input model definition [5]. In this case, we used the Tree Point Classification model from ArcGIS Living Atlas which has the training data shown in Figure 15, set the classification parameters as shown in Figure 16 and obtained a number of 1.408.183 points in the High Vegetation class.



X, Y and Z linear unit	meter
Z range	-19.29 m to 314.23 m
Number of Returns	1 to 5
Intensity	1 to 4092
Point spacing	0.6 ± 0.3
Scan angle	-23 to +23
Maximum points per block	8192
Extra attributes	Number of Returns
Class structure	[0, 5]

Figure 15. Tree Point Classification Model – Training Data [12]



Geoprocessing

Classify Point Cloud Using Trained Model

This tool modifies the input data.

Parameters Environments

Target Point Cloud
DOC_31.las

Processing Boundary

Input Model Definition
D:\Doctorat\3_An_III\1_lucru_date_Baia_Mare\V5\clasificare arcgis\

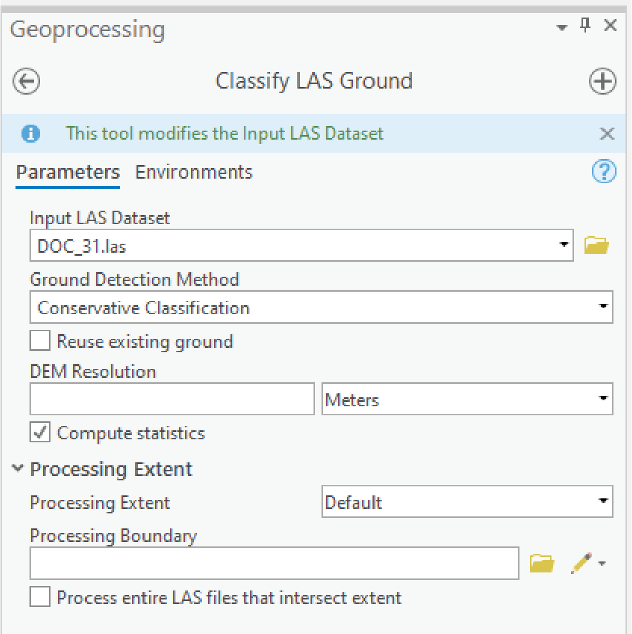
Target Classification Select All
☑ 0
☑ 5

Existing Class Code Handling
Edit All Points

☑ Compute statistics

Figure 16. Setting Parameters for the Classify Point Cloud Using Trained Model Tool

For the *Classify LAS Ground* tool, which classifies points with class code values of 0 (Never Classified), 1 (Unassigned) and 2 (Ground) [4], we set the parameters shown in Figure 17 and obtained a number of 16.066.673 points in the Ground class.



Geoprocessing

Classify LAS Ground

This tool modifies the Input LAS Dataset

Parameters Environments

Input LAS Dataset: DOC_31.las

Ground Detection Method: Conservative Classification

□ Reuse existing ground

DEM Resolution: Meters

☑ Compute statistics

Processing Extent

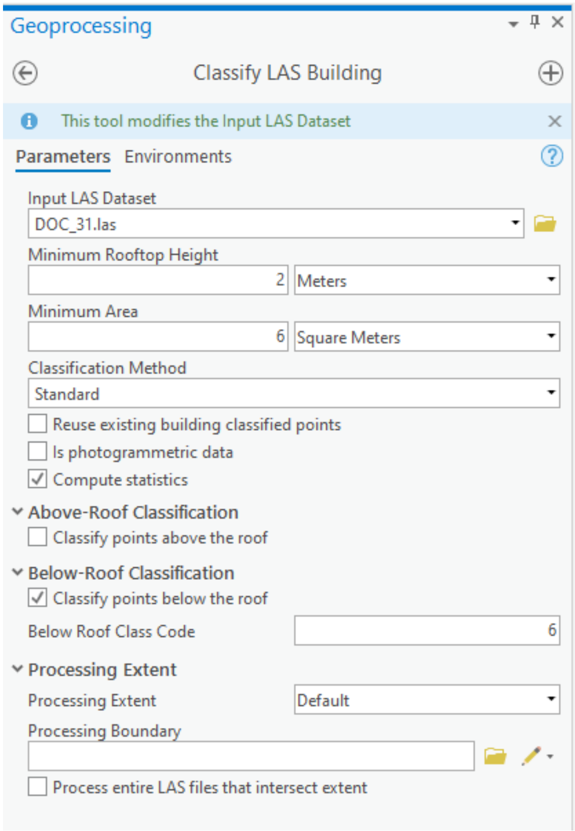
Processing Extent: Default

Processing Boundary

□ Process entire LAS files that intersect extent

Figure 17. Setting the Parameters for the Classify LAS Ground Tool

For the Classify LAS Building tool, in order to classify the walls points as Buildings, we checked the *Classify points below the roof* box when setting the parameters shown in Figure 18. With this tool, LAS points with class code values of 0, 1, and 6 (Building) are evaluated to determine if they fit the characteristics of building rooftops [3].



Geoprocessing

Classify LAS Building

This tool modifies the Input LAS Dataset

Parameters Environments

Input LAS Dataset: DOC_31.las

Minimum Rooftop Height: 2 Meters

Minimum Area: 6 Square Meters

Classification Method: Standard

□ Reuse existing building classified points

□ Is photogrammetric data

☑ Compute statistics

Above-Roof Classification

□ Classify points above the roof

Below-Roof Classification

☑ Classify points below the roof

Below Roof Class Code: 6

Processing Extent

Processing Extent: Default

Processing Boundary

□ Process entire LAS files that intersect extent

Figure 18. Setting the Parameters for the Classify LAS Building Tool

The issues we encountered when classifying the LiDAR point cloud using ArcGIS Pro tools include wrongly classified tall, narrow buildings as high vegetation (Figure 19) and other elements, such as cars, classified as High Vegetation (Figure 20).

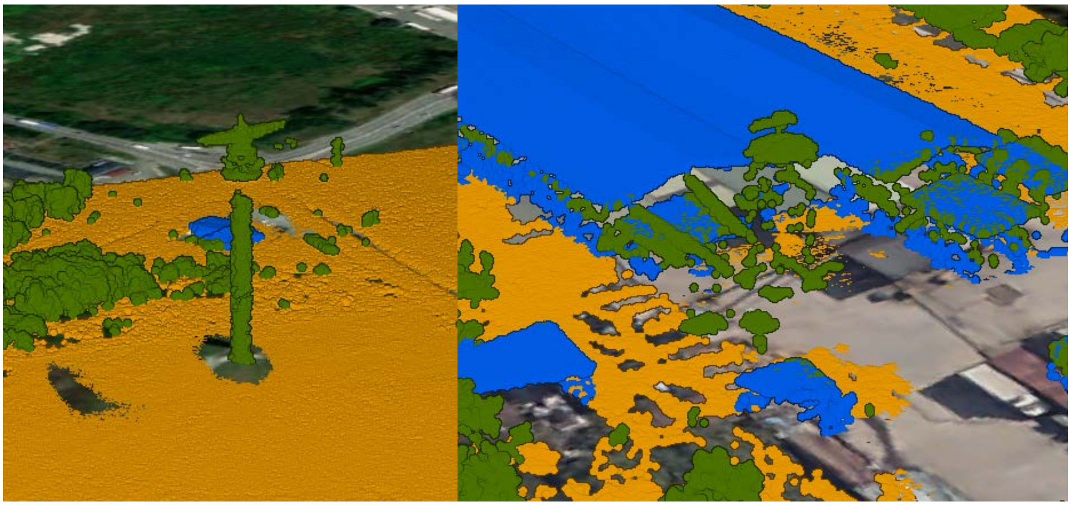


Figure 19. Tall and Narrow Buildings Classified as High Vegetation

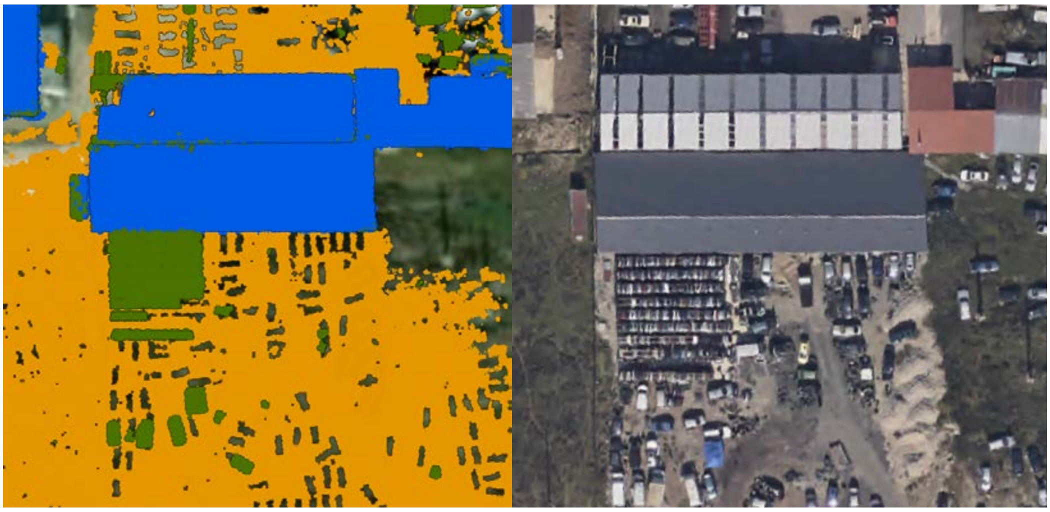
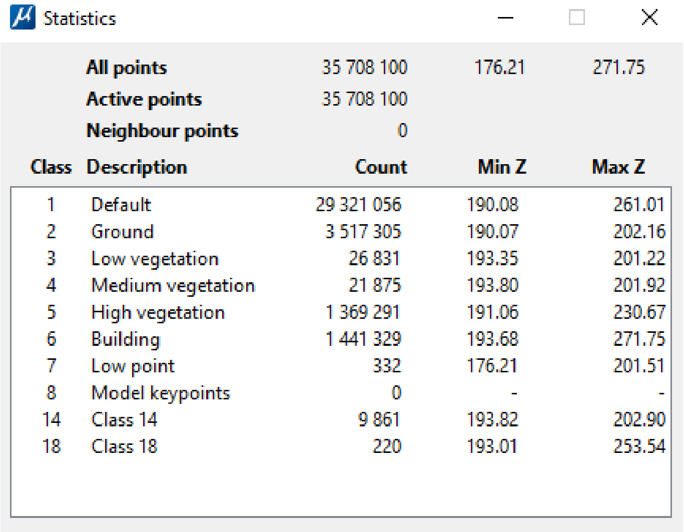


Figure 20. Cars Classified as Vegetation

4. Results and Discussion

Using the Bentley Microstation Terrasolid Suite, after classifying the Low points, we were able to do the Ground points classification, followed by the Building points classification and the High vegetation classification. The automatic point cloud classification was not enough and additional manual interventions had to be done. The results of the classification can be seen in Figure 21 and Figure 22.

Using the ArcGIS Pro Software, we utilised a deep learning model from ArcGIS Living Atlas in order to classify the High Vegetation points, and ArcGIS Pro implemented tools such as Classify LAS Ground and Classify LAS Building. These models and tools proved to need additional manual intervention as well. The results of the classification can be seen in Figure 23 and Figure 24.



	All points	35 708 100	176.21	271.75
	Active points	35 708 100		
	Neighbour points	0		
Class	**Description**	**Count**	**Min Z**	**Max Z**
1	Default	29 321 056	190.08	261.01
2	Ground	3 517 305	190.07	202.16
3	Low vegetation	26 831	193.35	201.22
4	Medium vegetation	21 875	193.80	201.92
5	High vegetation	1 369 291	191.06	230.67
6	Building	1 441 329	193.68	271.75
7	Low point	332	176.21	201.51
8	Model keypoints	0	-	-
14	Class 14	9 861	193.82	202.90
18	Class 18	220	193.01	253.54

Figure 21. Final Results of the Point Cloud Classification Using the Terrasolid Suite

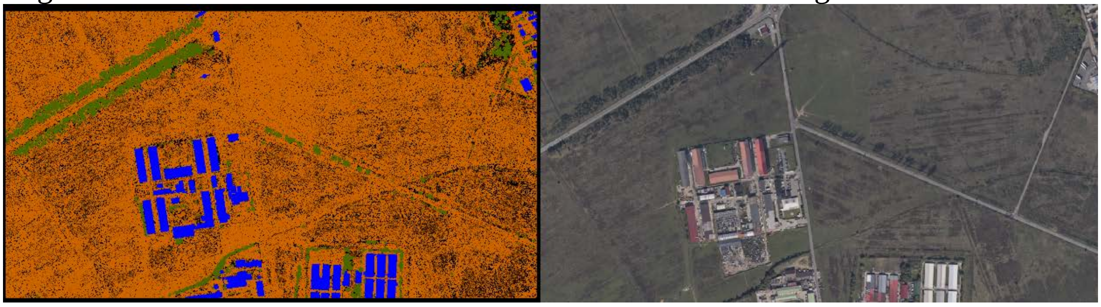
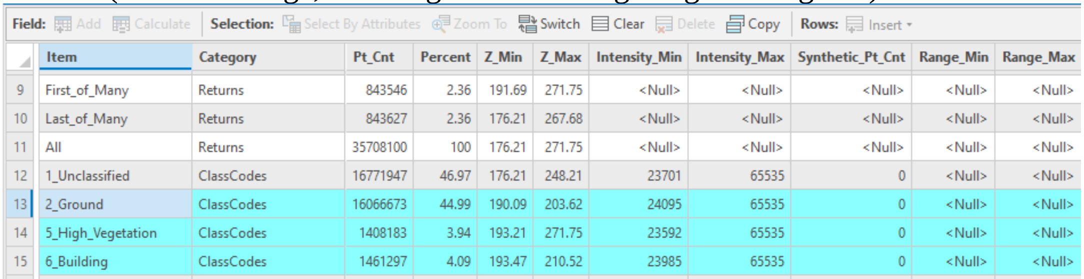


Figure 22. Final Results of the Point Cloud Classification Using the Terrasolid Suite (Ground-orange, Buildings-blue and High vegetation-green)


Field: Add Calculate Selection: Select By Attributes Zoom To Switch Clear Delete Copy Rows: Insert

	Item	Category	Pt_Cnt	Percent	Z_Min	Z_Max	Intensity_Min	Intensity_Max	Synthetic_Pt_Cnt	Range_Min	Range_Max
9	First_of_Many	Returns	843546	2.36	191.69	271.75	<Null>	<Null>	<Null>	<Null>	<Null>
10	Last_of_Many	Returns	843627	2.36	176.21	267.68	<Null>	<Null>	<Null>	<Null>	<Null>
11	All	Returns	35708100	100	176.21	271.75	<Null>	<Null>	<Null>	<Null>	<Null>
12	1_Unclassified	ClassCodes	16771947	46.97	176.21	248.21	23701	65535	0	<Null>	<Null>
13	2_Ground	ClassCodes	16066673	44.99	190.09	203.62	24095	65535	0	<Null>	<Null>
14	5_High_Vegetation	ClassCodes	1408183	3.94	193.21	271.75	23592	65535	0	<Null>	<Null>
15	6_Building	ClassCodes	1461297	4.09	193.47	210.52	23985	65535	0	<Null>	<Null>

Figure 23. Final Results of the Point Cloud Classification Using ArcGIS Pro

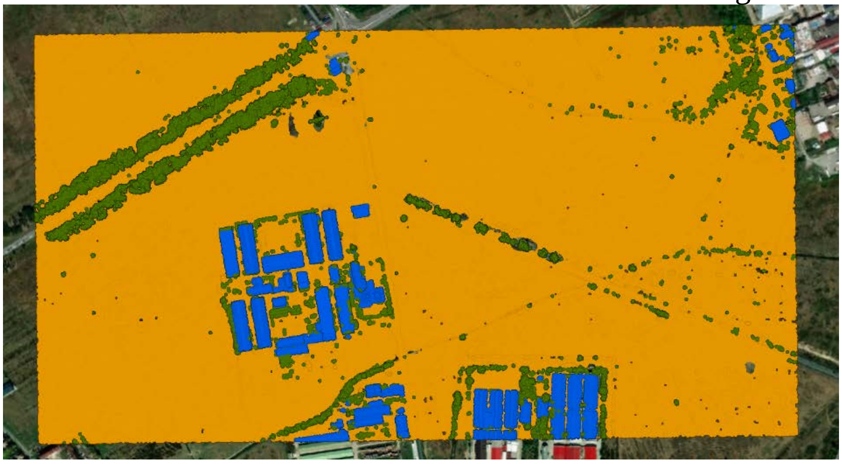


Figure 24. Final Results of the Point Cloud Classification Using ArcGIS Pro (Ground-Orange, Buildings-Blue and High Vegetation-Green)

5. Conclusions

Class/Software	Number of Points Obtained	
	Bentley Microstation Terrasolid Suite	ArcGIS PRO
Ground	3.517.305	16.066.673
Building	1.441.329	1.461.297
High Vegetation	1.369.291	1.408.183

Table 1. Results Comparison

The comparison between the results of both methods can be seen in Table 1.

Using Bentley's Microstation Terrasolid software product to classify a LiDAR point cloud proved to be an efficient process. Setting the classification parameters is not only related to the characteristics of the acquisitioned data but is also related to the characteristics of the computer on which the classification is performed. We have also come to the conclusion that automatic classification should always be accompanied by manual interventions where needed. On the other hand, using ArcGIS Pro needed less manual intervention, as building walls were classified automatically as well, but it also proved to be less efficient in terms of manual intervention, as it is not a point cloud dedicated software.

6. References

1. *Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., & Çöltekin, A. (2015). Applications of 3D City Models: State of the Art Review. ISPRS International Journal of Geo-Information, 4(4), Article 4. https://doi.org/10.3390/ijgi4042842*
2. *Cai, H., Wang, Y., Lin, Y., Li, S., Wang, M., & Teng, F. (2022). Systematic Comparison of Objects Classification Methods Based on ALS and Optical Remote Sensing Images in Urban Areas. Electronics, 11(19), Article 19. https://doi.org/10.3390/electronics11193041*
3. *Classify LAS Building (3D Analyst)—ArcGIS Pro | Documentation. (n.d.). Retrieved December 10, 2022, from https://pro.arcgis.com/en/pro-app/latest/tool-reference/3d-analyst/classify-las-building.htm*
4. *Classify LAS Ground (3D Analyst)—ArcGIS Pro | Documentation. (n.d.). Retrieved December 10, 2022, from https://pro.arcgis.com/en/pro-app/2.9/tool-reference/3d-analyst/classify-las-ground.htm*
5. *Classify Point Cloud Using Trained Model (3D Analyst)—ArcGIS Pro | Documentation. (n.d.). Retrieved December 10, 2022, from https://pro.arcgis.com/en/pro-app/latest/tool-reference/3d-analyst/classify-point-cloud-using-trained-model.htm*
6. *Diab, A., Kashef, R., & Shaker, A. (2022). Deep Learning for LiDAR Point Cloud Classification in Remote Sensing. Sensors, 22(20), Article 20. https://doi.org/10.3390/s22207868*
7. *Duran, Z., Ozcan, K., & Atik, M. E. (2021). Classification of Photogrammetric and Airborne LiDAR Point Clouds Using Machine Learning Algorithms. Drones, 5(4), Article 4. https://doi.org/10.3390/drones5040104*
8. *Grădinaru, A. P. (2021). Raport de cercetare științifică 1—Studiul metodelor de achiziție a datelor spațiale necesare reprezentării spațiale a imobilelor.*
9. *Morsy, S., & Shaker, A. (2022). Evaluation of LiDAR-Derived Features Relevance and Training Data Minimization for 3D Point Cloud Classification. Remote Sensing, 14(23), Article 23. https://doi.org/10.3390/rs14235934*
10. *Soininen, A. (2022). TerraScan USER GUIDE.*

11. *Terrasolid—Software For Point Cloud and Image Processing. (n.d.). Terrasolid. Retrieved December 3, 2022, from https://terrasolid.com/*
12. *Tree Point Classification—Overview. (n.d.). Retrieved December 10, 2022, from https://www.arcgis.com/home/item.html?id=58d77b24469d4f30b5f68973deb65599*
13. *Wen, C., Li, X., Yao, X., Peng, L., & Chi, T. (2021). Airborne LiDAR point cloud classification with global-local graph attention convolution neural network. ISPRS Journal of Photogrammetry and Remote Sensing, 173, 181–194. https://doi.org/10.1016/j.isprsjprs.2021.01.007*

7. Acknowledgements

This study was conducted within the Geodetic Engineering Measurements and Spatial Data Infrastructures Research Centre, Faculty of Geodesy, Technical University of Civil Engineering Bucharest

The LiDAR point cloud was collected within S.C. Cornel&Cornel Topoexim S.R.L. Company.

This study was conducted using Esri software licenses provided by the Doctoral School of the Technical University of Civil Engineering Bucharest.