

DETECTION OF BUILDING ROOFS AND FACADES FROM AERIAL LASER SCANNING DATA USING DEEP LEARNING

F. Pirotti ^{1,2*}, C. Zanchetta ³, M. Previtali ⁴, S. Della Torre ⁴

¹ CIRGEO - Interdepartmental Research Center of Geomatics, University of Padova, Viale dell'Università 16, Legnaro (PD), Italy - francesco.pirotti@unipd.it

² TESAF Department, University of Padova, Viale dell'Università 16, Legnaro (PD), Italy

³ DICEA Department, University of Padova, Via Francesco Marzolo, 9, 35131 Padova, Italy – carlo.zanchetta@unipd.it

⁴ Department of Architecture, Built Environment and Construction Engineering, Politecnico di Milano, 20133 Milan, Italy – (mattia.previtali, stefano.dellatorre)@polimi.it

Commission IV, WG IV/10

KEY WORDS: Deep learning, semantic classification, Tensorflow, laser scanning, lidar

ABSTRACT:

In this work we test the power of prediction of deep learning for detection of buildings from aerial laser scanner point cloud information. Automatic extraction of built features from remote sensing data is of extreme interest for many applications. In particular latest paradigms of 3D mapping of buildings, such as CityGML and BIM, can benefit from an initial determination of building geometries. In this work we used a LiDAR dataset of urban environment from the ISPRS benchmark on urban object detection. The dataset is labelled with eight classes, two were used for this investigation: roof and facades. The objective is to test how TensorFlow neural network for deep learning can predict these two classes. Results show that for “roof” and “facades” semantic classes respectively, recall is 84% and 76% and precision is 72% and 63%. The number and distribution of correct points well represent the geometry, thus allowing to use them as support for CityGML and BIM modelling. Further tuning of the hidden layers of the DL model will likely improve results and will be tested in future investigations.

1. INTRODUCTION

In the last years 3D models and point clouds became more and more popular in several application fields, among the other: urban planning (Boyko and Funkhouser, 2011), facility management (Wang et al., 2015), cultural heritage (Tucci et al., 2016), indoor navigation (Díaz Vilariño et al., 2016) just to name few of them. The popularity of point clouds as a primary source of information is connected with the recent technological developments that are making available more reliable, accurate and easy to use instruments (Adán et al., 2019). In addition, nowadays a set of pre-processing steps like noise reduction, data registration and alignment are mainly relying on automated procedures allowing a large accessibility to scanning technology also to non-expert users (Wujanz et al., 2019). Both Airborne laser scanning – ALS and dense image matching, especially when using oblique images (Rupnik et al., 2017) are the two main sources of massive point clouds. However, despite their broad availability, the further steps of 3D data analysis are mainly relying on manual operations; e.g. outlier removal (Pirotti et al., 2018), segmentation, classification, semantic enrichment for example for modelling 3D city models (CityGML) and Building Infrastructure Modelling (BIM) from LiDAR (Barazzetti et al., 2010; Yao et al., 2018). For this reason, there is still a relevant need of methods, preferably automatic, to provide 3D data with meaningful attributes that characterize and provide significance. Indeed, a 3D point cloud is probably the simplest data structure that can be used to capture the shape, size and position of an object into a 3D space. By integrating different sensors, it is possible to associate to each point further characteristics like colours (combining a camera to instrument), temperature (by using a thermal camera) or multi-spectral information (Costanzo et al., 2015), etc. However, semantic and topological information

cannot be directly integrated into a point cloud. This is the reason why automated analysis of 3D point clouds has become a topic of major interest. Two main steps are identified as crucial in this task: data segmentation and data classification. Segmentation (Rabbani et al., 2016) can be defined as the subdivision of the point cloud into a set of coherent elements according to a specific feature, e.g., geometrical, radiometric, etc. Instead, classification (Grilli et al., 2017) is the assignment of a specific class (label) to each point according to a specified criterion, e.g., functional, positional, etc. Solving these two problems is made more complex due to: (i) characteristics of the acquired data and (ii) inter-class variability. The first is associated with irregular sampling and density, noise of the data, registration errors, etc. The latter is associated to the large variety of objects into the same class (e.g., different shape and colour of objects under the class “building roof”). For this reason, multiple works are focusing on optimizing the problem for specific research studies considering the specific needs of an application field, e.g., cultural heritage (Fiorillo et al., 2013), autonomous navigation, urban planning (Rodríguez-Cuenca et al., 2015), etc.

The two key elements of any classification are: discriminative features and proper classifiers. Given a 3D point and its local neighborhood, geometric features may be derived from the spatial arrangement of 3D points within its neighbourhood. Features may include distances, angles and angular variations, eigenvalues of the 3D structure tensor in the local neighbourhood, height and local plane characteristics, height characteristics and curvature properties (Weinmann et al., 2015, Weinmann et al., 2017). However, when deriving such features the proper selection of 3D neighbourhoods is of primary importance to obtain an appropriate description of the local 3D structure. In addition, typical effects in point cloud data like

noise, loss of sharp features and outliers worsen the problem since those factors may significantly affect features quality. In particular, we can distinguish between single scale approaches, where the scale for features definition is the same for each point in the 3D point cloud, or multi-scale features, where features are derived at multiple scales. Based on the derived feature vectors, classification is typically conducted in a supervised way.

Classification can be carried out either at individual point or exploiting larger context. Individual point classification may be carried out very efficiently, but there it is more prone to be affected by point cloud noise and features variability, especially when using single scale approaches, and it may end out in a noisy classification result. Contextual classification accounts for the fact that the class labels of neighbouring 3D points tend to be correlated. The exploitation of larger context is a trend of scene classification. For example, by plane segmentation (Xu et al., 2014) or second or higher order Markov random field (MRF) or conditional random fields (CRF) (Lafarge and Mallet, 2012, Niemeyer et al., 2014, Niemeyer et al., 2016).

Although existing methods have shown good results in ALS filtration and classification, they still need much human labour to generate fully realisable results. In this work we want to exploit existing Deep Learning (DL) algorithms for point cloud classification. The first part of the method consist in extracting predictors from geometric features at each point considering nn nearest neighbours. The cardinality of nn is not fixed, but determined to maximize geometric consistency. The vectors of predictors at each point are used for training. Labels are then assigned to each point in the point cloud. The classification is then carried out using DL a class of machine learning algorithms that use multi-layer neural networks

The remaining of the paper is organized as follows: next section provides an overview of related work, then materials and methods are presented and defined, and finally results are discussed in light of suggestions for improvements and future work.

2. RELATED WORK

As previously anticipated classification methods can be categorized as single point- or contextual-based (Gerke and Xiao, 2014). This section presents a brief review of previous works related to this subject.

Point-based methods generally extract point-wise features locally from the neighbourhood defined by a sphere or cylinder, and then supervised or unsupervised classifiers are used.

Classification is carried out by using methods such as Random Forests (RFs) (Breiman, 2001) and support vector machines (SVMs) (Mountrakis et al., 2011). Points are generally labelled individually in the feature space without considering relationships, which often leads to discontinuities in the classification results. To overcome this drawback object-based methods firstly subdivide the point cloud into segments sharing homogeneous properties (e.g. geometry, radiometric, curvature, etc.) or to partition the partition the point cloud into regular-shaped clusters known as voxels (Aijazi et al., 2013) or supervoxels (Lim and Suter, 2009). Starting from this clusters more discriminative features are extracted and classification is then carried out on the identified segments using cluster features.

A recent trend in the classification of point clouds is to exploit contextual information. For example, some point based methods incorporates contextual information by adopting Markov Random Field (MRF), Associative and non-Associative Markov

Networks (Munoz et al., 2009) Conditional Random Fields (CRF) (Lafarge and Mallet, 2012, Niemeyer et al., 2014, Niemeyer et al., 2016), Simplified Markov Random Fields (Lu and Rasmussen, 2012). However, the adoption of contexture information only involves a small local region around the selected points and considers only point-level interactions Niemeyer et al. (2016) consider longer interactions through higher order Potts model while Hackel et al. (2016) use multiscale representation of point clouds. Deep convolutional networks – ConvNets (LeCun et al., 1989) have shown a great success at solving recognition and classification tasks for 2D image dataset. However, the adoption of ConvNets to 3D models is not-trivial. The most straightforward way to make ConvNets applicable to 3D data, is to rasterize 3D models onto uniform voxel grids. Such approach however leads to excessively large memory footprints, slow processing times and low data resolution. A volumetric representation is associated with a quantization tradeoff: choosing a coarse grid leads to quantization artifacts and to substantial loss of information, whereas choosing a fine grid significantly increases the number of voxels, which are mostly empty but still induce a high computational cost. Several groups investigated application of ConvNets to the rasterizations of 3D models on uniform 3D grids.

3. MATERIALS AND METHODS

The data used for testing in this investigation is a dataset from the ISPRS benchmark on urban object detection and 3D building reconstruction (Rottensteiner et al., 2014). It consists, among other things, in a laser scanner dataset of the city of Vaihingen (Germany). Points are labelled with eight classes, among which there are two building elements: roofs and facades.

The objective of this investigation is to assess how well two semantic classes, roofs and facades can be detected using Deep Learning(DL). DL is a class of machine learning algorithms that use multi-layer neural networks. TensorFlow, a framework library for DL models created by Google's artificial intelligence team and released in 2015 with an open source license, will be used for this task.

The first part of the methods consist in extracting predictors from geometric features at each point considering a set of nm nearest neighbours. The nm value is not fixed, but determined to maximize geometric consistency, as per indications in (Weinmann et al., 2015, 2017). This is done by selecting the number of nm which result in the lowest Shannon entropy index value, calculated using normalized eigenvalues of the 3D covariance matrix:

$$nm = \min \left\{ \sum_i^3 \lambda_i \cdot \ln(\lambda_i) \right\} \quad (1)$$

where $nm_{min} = 15$ and $nm_{max} = 100$ similar to previous investigations. The 3D structure tensor - 3D covariance matrix - is calculated from the nm points.

$$\text{cov}(i, j) = \frac{\sum_{n=1}^m (i_n - \bar{i})(j_n - \bar{j})}{N-1} \quad (2)$$

$$i = j = \{x, y, z\}$$

$$C_m = \begin{bmatrix} \text{cov}(x, x) & \text{cov}(x, y) & \text{cov}(x, z) \\ \text{cov}(y, x) & \text{cov}(y, y) & \text{cov}(y, z) \\ \text{cov}(z, x) & \text{cov}(z, y) & \text{cov}(z, z) \end{bmatrix}$$

$$C_m \in \mathbb{R}^{3 \times 3} \quad (3)$$

Twenty-three predictors are extracted from the 3D structure tensors both using 3D and 2D features some of which are described in (Weinmann et al., 2015, 2017). Intensity information was also used. training the DL model. Labels are then assigned to each point in the point cloud, defined as “roofs”, “facades” and “other”. Output values are tested for accuracy using K-fold validation with K=3.

Deep learning is done using convolutional neural networks, that are currently used for image classification and are here applied through the TensorFlow framework for classification of unstructured points from LiDAR. The difference between images and the tested point cloud is that descriptors in images are reflectance values, whereas in point clouds specific descriptors are extracted from neighbouring geometric features, as described in the previous section. Extraction of descriptors from points using neighbours can be considered similar to convolution, as neighbours are considered and used to create a unique descriptive value. The input layer is created with the descriptors. The framework consists in a convolutional layer, pooling layers and dense layers which perform the actual classification. The final result of classification is a actually a probability value between zero and one, which is the output of the softmax activation function that normalizes output node values.

4. RESULTS AND DISCUSSION

Results show that for “roof” and “facades” semantic classes respectively, accuracy is 84% and 76%. Figure 1 below shows additional details on a roof. It is visible that misclassified points are on the border of the roof and on isolated points. The number and distribution of correct points well represent the geometry, thus allowing to use them as support for CityGML and BIM modelling. Further tuning of the hidden layers of the DL model will likely improve results.

The results are shown in figure 1 where it is clear that points that belong to buildings (roofs and facades) are mostly regular and correct, but further analyses are necessary to explain the efficiency of the algorithm. For further modelling BIM and CityGML representations, the geometries of the building are, of course, important. Therefore the classification results should be analysed in terms of correctly labelled points, but in particular false positives are important to detect and quantify. False negatives are not as important if not numerous. This is because false positives create wrong geometries in a scenario of automatic geometry creation, whereas false negatives can be interpolated by correct points, if there are enough in the neighbourhood.

It is trivial to say that similar geometries can create false positives. This is particularly true in the case of the façade class. As can be seen from table 1 below. Roofs are obviously easier to detect by a trained neural network than facades. Further visual

analysis shows that trees and low vegetation near walls can be misinterpreted as facades. The low overall number of façade points results in a higher relative number of false positives.

| | Roof | Façade |
|------------------|------|--------|
| Precision | 0.72 | 0.63 |
| Recall | 0.84 | 0.76 |

Table 1. Precision and recall of roof and façade classes



Figure 1. Real and predicted classes left and middle respectively: blue=facades, green=rooftop, grey=others. Right: detail of rooftop with misclassified points in blue and red.

As seen in a close-up in figure 2, there are many false positives also of roof class that belong to ground objects. These errors can be partly corrected by a pre-processing step that extracts an accurate ground model. There are tested methods for extracting ground points from a LiDAR point cloud, mostly using progressive iterative triangulation of lower points in a user-defined neighbourhood. An accurate terrain model can be used to automatically remove points that are closer than 2 m from the terrain, thus removing many of the above mentioned false positives, and improve results.

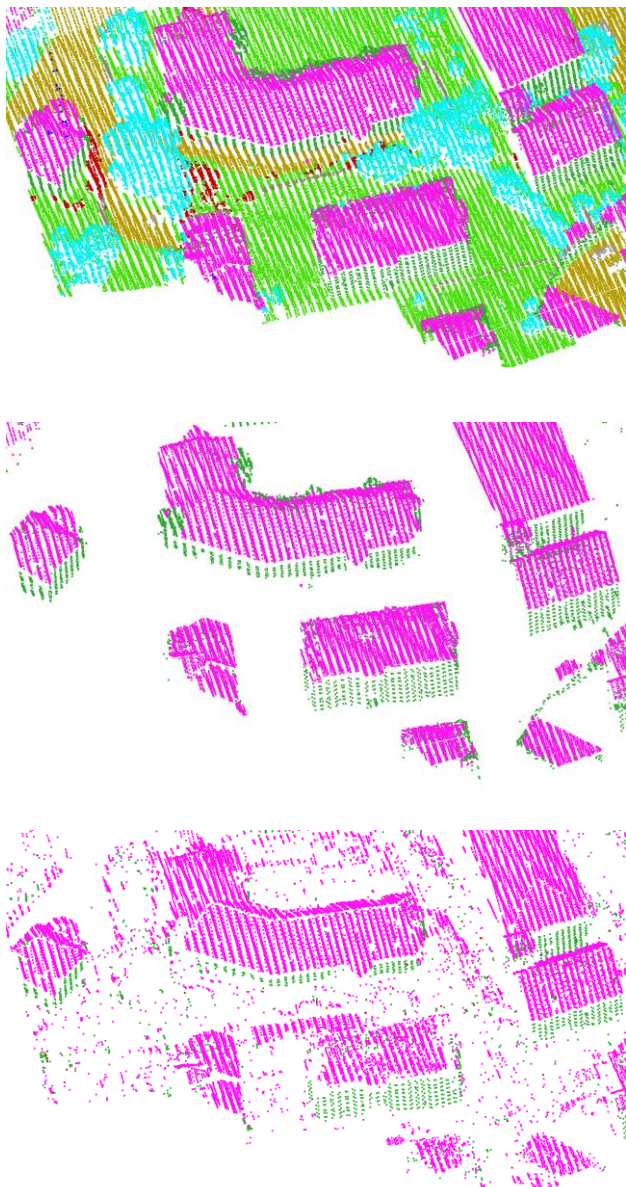


Figure 2. From top to bottom respectively detail of original points, real roof/façade classes and predicted roof and façade classes.

The original benchmark dataset contains eight labelled classes, it is therefore possible to do a more in-depth analysis of commission errors considering from which other classes false positives to roofs and facades came from. Figure 3 shows that some classes were wrongly attributed to the two target classes. As mentioned, figure 3 shows that trees are the features that mostly get assigned incorrectly to facades, most likely due to urban vegetation growing close to buildings. The number of neighbours that create descriptors range from twenty to one hundred and even if the process aims at minimizing the entropy, points belonging to facades get mixed with points from trees, causing a point set, during the training phase, that leads to inaccurate descriptors of the geometry of a façade.

In general figure 3 shows that vegetation is cause of false positives, also in the case of low vegetation that gets misinterpreted as building elements (both roof and façade). As the training was done with a random subset of points from the two classes, an improvement can likely be seen by selecting manually “ideal” roof and façade examples for training. In other

words, find buildings at a distance from vegetation, in order for the training step to extract “clean” predictors and avoid noise from vegetation.

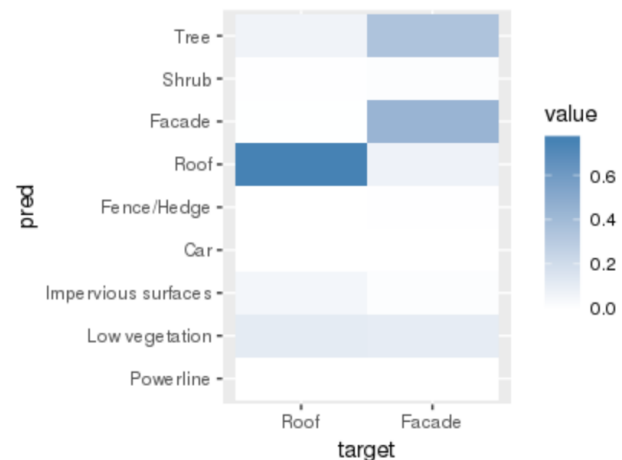


Figure 3. Commission errors per each of the eight labels available in the benchmark dataset.

Further work is needed to investigate on improvements from tuning the hyperparameters of the hidden layer in the TensorFlow framework. This step can be done by iterating different parameters and checking accuracy metrics over independent subsets with respect to the training set. It is worth noting that the presented work has been all done with open-source software for LiDAR processing (Pirotti, 2019), R for defining descriptors and applying TensorFlow, CloudCompare for further visual analysis.

5. CONCLUSIONS

The investigation presented shows that it is possible to extract information regarding building geometries from a LiDAR survey with a trained neural network framework. Tensorflow was used in this case, thus applying a deep learning paradigm, i.e. using a convolutional neural network. Training the model requires first to extract descriptors from the point set. This is a convolutional-type operation, as neighbours are considered and geometric descriptors of point subsets are used. Once the model is trained and validated, accuracy metrics are extracted, showing for “roof” and “facades” semantic classes respectively a recall value of 84% and 76% and precision values of 72% and 63%. This model can be improved as highlighted in discussion, and then it can be applied to other point cloud datasets of similar scenarios.

ACKNOWLEDGEMENTS

This research is supported in part by the Italian Ministry of Education through the project of national interest (PRIN) Urban-Geo Big Data (project code: 20159CNLW8).

REFERENCES

- Adán, A., Quintana, B., Prieto, S. A., 2019. Autonomous Mobile Scanning Systems for the Digitization of Buildings: A Review. *Remote Sensing*, 11(3), 306.
- Aijazi, A., Checchin, P., Trassoudaine, L., 2013. Segmentation based classification of 3D urban point clouds: A super-voxel

- based approach with evaluation. *Remote Sensing*, 5(4), 1624-1650.
- Barazzetti, L., Brovelli, M.A., Valentini, L., 2010. LiDAR digital building models for true orthophoto generation. *Applied Geomatics*, doi:10.1007/s12518-010-0034-9
- Boyko, A. and Funkhouser, T., 2011. Extracting roads from dense point clouds in large scale urban environment. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(6), S2-S12.
- Breiman, L. (2001). Random forests. *Machine learning*, 45(1), 5-32.
- Costanzo, A., Minasi, M., Casula, G., Musacchio, M., Buongiorno, M., 2015. Combined use of terrestrial laser scanning and IR thermography applied to a historical building. *Sensors*, 15(1), 194-213.
- Díaz Vilariño, L., Boguslawski, P., Khoshelham, K., Lorenzo, H., Mahdjoubi, L., 2016. Indoor navigation from point clouds: 3D modelling and obstacle detection. *International archives of photogrammetry, remote sensing and spatial information sciences*, 275-281.
- Fiorillo, F., Fernández-Palacios, B. J., Remondino, F., Barba, S., 2013. 3d Surveying and modelling of the Archaeological Area of Paestum, Italy. *Virtual Archaeology Review*, 4(8), 55-60.
- Gerke, M., and Xiao, J., 2014. Fusion of airborne laserscanning point clouds and images for supervised and unsupervised scene classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, 87, 78-92.
- Grilli, E., Menna, F., Remondino, F., 2017. A review of point clouds segmentation and classification algorithms. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42, 339.
- Hackel, T., Wegner, J. D., Schindler, K., 2016. FAST SEMANTIC SEGMENTATION OF 3D POINT CLOUDS WITH STRONGLY VARYING DENSITY. *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences*, 3(3).
- Lafarge, F., and Mallet, C., 2012. Creating large-scale city models from 3D-point clouds: a robust approach with hybrid representation. *International journal of computer vision*, 99(1), 69-85.
- LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., Jackel, L. D., 1989. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4), 541-551.
- Lim, E. H., and Suter, D., 2009. 3D terrestrial LIDAR classifications with super-voxels and multi-scale conditional random fields. *Computer-Aided Design*, 41(10), 701-710.
- Lu, Y., and Rasmussen, C., 2012. Simplified Markov random fields for efficient semantic labeling of 3D point clouds. In 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (pp. 2690-2697). IEEE.
- Mountrakis, G., Im, J., Ogole, C., 2011. Support vector machines in remote sensing: A review. *ISPRS Journal of Photogrammetry and Remote Sensing*, 66(3), 247-259.
- Munoz, D., Bagnell, J. A., Vandapel, N., Hebert, M., 2009. Contextual classification with functional max-margin markov networks. In 2009 *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 975-982). IEEE.
- Niemeyer, J., Rottensteiner, F., Soergel, U., 2014. Contextual classification of lidar data and building object detection in urban areas. *ISPRS journal of photogrammetry and remote sensing*, 87, 152-165.
- Niemeyer, J., Rottensteiner, F., Sörgel, U., Heipke, C., 2016. Hierarchical higher order crf for the classification of airborne lidar point clouds in urban areas. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. 41 655-662.
- Pirotti, F., Ravanelli, R., Fissore, F., Masiero, A., 2018. Implementation and assessment of two density-based outlier detection methods over large spatial point clouds. *Open Geospatial Data, Software and Standards*, 3, 14. doi:10.1186/s40965-018-0056-5
- Pirotti, F., 2019. The Role of Open Software and Standards in the Realm of Laser Scanning Technology. *Open Geospatial Data, Software and Standards*, in print.
- Rabbani, T., Van Den Heuvel, F., Vosselmann, G., 2006. Segmentation of point clouds using smoothness constraint. *International archives of photogrammetry, remote sensing and spatial information sciences*, 36(5), 248-253.
- Rottensteiner, F., Sohn, G., Gerke, M., Wegner, J.D., Breitkopf, U., Jung, J., 2014. Results of the ISPRS benchmark on urban object detection and 3D building reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 93, 256-271. doi:10.1016/j.isprsjprs.2013.10.004
- Rodríguez-Cuenca, B., García-Cortés, S., Ordóñez, C., Alonso, M., 2015. Automatic detection and classification of pole-like objects in urban point cloud data using an anomaly detection algorithm. *Remote Sensing*, 7(10), 12680-12703.
- Rupnik, E., Daakir, M., Pierrot Deseilligny, M., 2017. MicMac – a free, open-source solution for photogrammetry. *Open Geospatial Data, Software and Standards*, 2, 14. doi:10.1186/s40965-017-0027-2
- Tucci, G., Bonora, V., Fiorini, L., Conti, A., 2016. The florence baptistry: 3-D Survey as a knowledge tool for historical and structural investigations. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 41, pp. 977-984. DOI: 10.5194/isprsarchives-XLI-B5-977-2016
- Wang, C., Cho, Y. K., Kim, C., 2015. Automatic BIM component extraction from point clouds of existing buildings for sustainability applications. *Automation in Construction*, 56, 1-13.
- Weinmann, M., Jutzi, B., Hinz, S., Mallet, C., 2015. Semantic point cloud interpretation based on optimal neighbourhoods, relevant features and efficient classifiers. *ISPRS Journal of Photogrammetry and Remote Sensing*, 105, 286-304.
- Weinmann, M., Weinmann, M., Mallet, C., Brédif, M., 2017. A classification-segmentation framework for the detection of

individual trees in dense MMS point cloud data acquired in urban areas. *Remote sensing*, 9(3), 277.

Wujanz, D., Barazzetti, L., Previtali, M., Scaioni, M., 2019. A Comparative study among three registration algorithms: performance, quality assurance and accuracy. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 422, 779-786.

Xu, S., Vosselman, G., Elberink, S. O., 2014. Multiple-entity based classification of airborne laser scanning data in urban areas. *ISPRS Journal of photogrammetry and remote sensing*, 88, 1-15.

Yao, Z., Nagel, C., Kunde, F., Hudra, G., Willkomm, P., Donaubaer, A., Adolphi, T., Kolbe, T.H., 2018. 3DCityDB - a 3D geodatabase solution for the management, analysis, and visualization of semantic 3D city models based on CityGML. *Open Geospatial Data, Software and Standards*, doi:10.1186/s40965-018-0046-7