

# SELF-LEARNING ONTOLOGY FOR INSTANCE SEGMENTATION OF 3D INDOOR POINT CLOUD

F. Poux<sup>1,\*</sup>, J.J. Ponciano<sup>2</sup>

<sup>1</sup> Geomatics Unit, University of Liège (ULiège), Belgium; fpoux@uliege.be

<sup>2</sup> i3mainz, University of Applied Sciences Mainz, Germany; jean-jacques.ponciano@hs-mainz.de

Technical Commission TCII - Photogrammetry

**KEY WORDS:** 3D point cloud; voxel; feature extraction; instance segmentation; classification; 3D semantics; ontology; deep learning

## ABSTRACT:

Automation in point cloud data processing is central for efficient knowledge discovery. In this paper, we propose an instance segmentation framework for indoor buildings datasets. The process is built on an unsupervised segmentation followed by an ontology-based classification reinforced by self-learning. We use both shape-based features that only leverages the raw X, Y, Z attributes as well as relationship and topology between voxel entities to obtain a 3D structural connectivity feature describing the point cloud. These are then used through a planar-based unsupervised segmentation to create relevant clusters constituting the input of the ontology of classification. Guided by semantic descriptions, the object characteristics are modelled in an ontology through OWL2 and SPARQL to permit structural elements classification in an interoperable fashion. The process benefits from a self-learning procedure that improves the object description iteratively in a fully autonomous fashion. Finally, we benchmark the approach against several deep-learning methods on the S3DIS dataset. We highlight full automation, good performances, easy-integration and a precision of 99.99% for planar-dominant classes outperforming state-of-the-art deep learning.

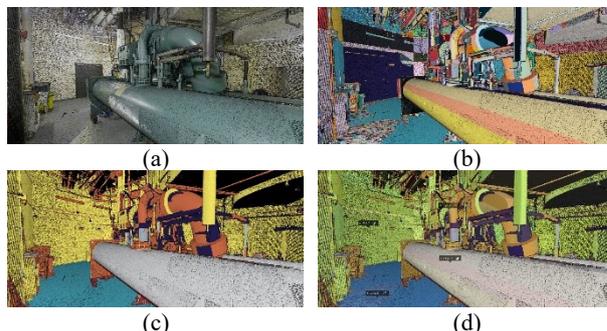
## 1. INTRODUCTION

Extracting knowledge from raw point cloud data is actively driving academic and industrial research. There is a great need for automated processes that can speed up and make existing frameworks faster and more reliable (Poux and Billen, 2019a). It often integrates a classification step to extract any relevant information regarding one application domain. However, one classification approach cannot efficiently satisfy all the domains as the semantic concepts that are attached to objects and the location can vary, depending on uses (e.g. considering a chair as an object, or its legs). Therefore, ensuring that such information is transferable to benefit other applications could provide a great opening on point cloud data usage. Yet, this is a non-trivial task that necessitates highly interoperable reasoning and a flexible way to handle data, relationships, and semantics. Our method considers the Gestalt's theory (Koffka, 2013), which states that the whole is greater than the sum of its parts, and that relationships between the parts can yield new properties/features. We want to leverage the human visual system predisposition to group sets of elements.

In this paper, we aim at providing an instance segmentation module to extract instances for each class through a full workflow from voxel partitioning to semantic-guided classification.

The module acts as a standalone within a Smart Point Cloud Infrastructure (Poux and Billen, 2019b)—a set-up where point data is the core of decision-making processes—and it handles point clouds with heterogeneous characteristics. As such, we investigate an objective solution for versatile 3D point cloud semantic representation transparent enough to be usable on different point clouds and within different application domains such as Architecture, Engineering, and Construction (AEC), Building Information Modelling (BIM), Facility Management (FM). This orients our research to learning architectures while strongly considering limitations of relying too heavily on point

cloud training datasets. As such, we study the usage of formalized ontologies, by creating multi-level object descriptions that can guide a classification process. To explore new ways in Geo-Artificial Intelligence, we experiment self-learning processes that can adapt in full autonomy to the data specificity without requiring any training data.



**Figure 1** The classification process and naming conventions. (a) coloured point cloud; (b) segmentation result; (c) semantic segmentation; (d) instance segmentation.

To this end, we first propose an unsupervised segmentation extending (Poux and Billen, 2019a) approach by extracting pertinent clusters that retain both shape and relationship information. Then, we study their fit for instance segmentation tasks using an ontology of classification. We provide an additional layer of interoperability through an initial approximative semantic definition reinforced by a self-learning process that adapts the object description to the point cloud data.

The article is structured as following. In Section 2, we review related work that constitute the base for building up our approach. In section 3 we present our approach constituted of a Voxel-based graph representation, a Feature Extraction, an Unsupervised

\* Corresponding author

Segmentation base of a Semantic model of objects and rule-based reasoning Automatic Classification and self-learning.

Finally, we benchmark performances and results against state of the art deep-learning methods. The experiments were conducted on the full S3DIS (Armeni et al., 2016) indoor dataset, but it is generalizable to outdoor environments with man-made objects/characteristics.

## 2. RELATED WORKS

The first challenge in pure segmentation frameworks is to obtain group of points that can describe the organization of the data by a relevant clustering with enough detachment. The work of Weber et al. provides the first approach of using relationships while conserving the point-based flexibility (Weber et al., 2010). They propose an over-segmentation algorithm using ‘supervoxels’, an analogue of the superpixel approach for 2D methods. Based on a local k-means clustering, they try and group the voxels with similar feature signatures (39-dimensional vector) to obtain segments. The work is interesting because it is one of the earliest to try and propose a voxel-clustering with the aim of proposing a generalist decomposition of point cloud data in segments. Son et Kim use such a structure in (Son and Kim, 2017) for indoor point cloud data segmentation. They aim at generating the as-built BIMs from laser-scan data obtained during the construction phase. Their approach consists of three steps: region-of-interest detection to distinguish the 3D points that are part of the structural elements to be modelled, scene segmentation to partition the 3D points into meaningful parts comprising different types of elements while using local concave and convex properties between structural elements, and volumetric representation. The approach clearly shows the dominance of planar features in man-made environments.

Another very pertinent work is (Wang et al., 2017), which proposes a SigVox descriptor. The paper first categorizes object recognition task following the approach of: (1) model-fitting based (starts with segmenting and clustering point cloud, followed by fitting point segments); (2) semantic methods (based on a set of rule-based prior knowledge); and, (3) shape-based methods (shape featuring from implicit and explicit point clusters). They use a 3D ‘EGI’ descriptor to differentiate voxels that only extract specific values from a Principal Component Analysis (PCA) (Liu and Ramani, 2009). The approach proves useful for MLS point clouds, grouping points in object candidates, following the number. Another voxel-based segmentation approach is given in (Xu et al., 2018, 2017) while using a probabilistic connectivity model. The authors use a voxel structure, in which they extract local contextual pairwise-connectivity. It uses geometric ‘cues’ in a local Euclidean neighbourhood to study the possible similarity between voxels. This approach is similar to (Zhu et al., 2017), where the authors classify a 2.5D aerial LiDAR point cloud multi-level semantic relationships description (point homogeneity, supervoxel adjacency, class-knowledge constraints). They use a feature set, among others, composed of the elevation above ground, normal vectors, variances, and eigen-based features. Another analogous approach can be found in (Wang et al., 2016) for building point detection from vehicle-borne LiDAR data based on voxel group and horizontal hollow analysis. Authors present a framework for automatic building point extraction, which includes three main steps: voxel group-based shape recognition, category-oriented merging, and building point identification by horizontal hollow ratio analysis. This article proposes a concept of ‘voxel group’, where each group is composed of several voxels that belong to one single class-dependent object. Subsequently, the shapes of point clouds in each voxel group are recognized and this shape information is utilized to merge the voxel group. This article

efficiently leverages a sensory characteristic of vehicle-borne LiDAR building data but specializes the approach in consequence.

The references (Ben-Shabat et al., 2018, 2017) are built upon a graph-based over-segmentation methodology that is composed of a local 3D variation extraction, a graph construction, descriptor computation, and edge-wise assignment, followed by sequential subgraph criteria-based merging. The used descriptors are mainly RGB, location and normal vectors on top of the fast point feature histogram (Rusu et al., 2009). While the approach is domain-related, it offers some additional insight regarding the power of relational approaches between local point patches for the task of semantic segmentation. However, as shown in (Nguyen et al., 2018), using a multi-scale voxel representation of 3D space is very beneficial, even in complexity reduction of terrestrial lidar data. The authors propose a combination of point and voxel generated features to segment 3D point clouds into homogeneous groups in order to study the surface changes and vegetation cover. The results suggest that the combination of point and voxel features represent the dataset well, which shows the benefit of dual representations. The work of (Ni et al., 2017) uses Random Forests for aerial Lidar point cloud segmentation, which aims at extracting planar, smooth, and rough surfaces, being classified using semantic rules. This is interesting to answer specific domains through ontology formalization.

The work of (Ben Hmida et al., 2012) proposes to use the OWL ontology language (Antoniou and Harmelen, 2004) and the Semantic Web Rule Language (SWRL), presented in (Horrocks et al., 2004), for the detection of objects in the 3D point cloud. This approach aims at detecting railway objects (e.g. mast, signals) to feed a GIS system or an Industry Foundation Classes (IFC) file. The approach consists of (1) detecting geometries through SWRL built-ins that process the point cloud according to the object description in the ontology; (2) characterizing the topology between geometries through SWRL built-ins that analyse two geometries; and, (3) classifying objects through SWRL rules according to identified geometries and their topology. It has the advantage to benefit from the expert knowledge of the railway domain to guide the detection process, but it specializes the approach to this domain. A pertinent work for object detection in 3D using an ontology in different specific contexts is presented in (Dietenbeck et al., 2017). For each specific context, this approach proposes to build a multi-layer ontology on top of a basic knowledge layer that represents 3D objects features through their geometry, topology, and possible attributes. The authors use the ontology to generate a decision tree that allows for performing the segmentation and annotation of the point cloud simultaneously. This approach has the advantage to be applicable in different contexts and to use expert knowledge for a given domain, even if experts have no computer sciences skills. Knowledge-based approaches mainly use knowledge about object attributes, data features, and algorithms to enhance the detection process. They can solve many ambiguity problems by combining knowledge of different object attributes.

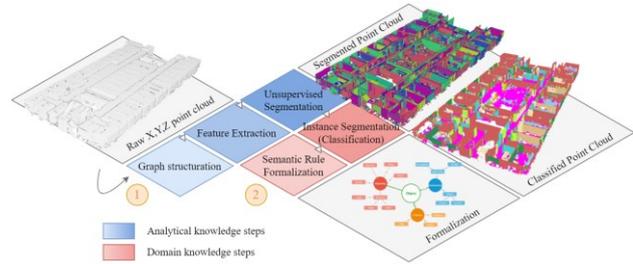
These methodologies contrast with deep learning approaches, as they try to solve the semantic segmentation problem by first understanding which set of features/relations will be useful to obtain the relevant results. The following methodologies directly start with the data and will learn by themselves how to combine the initial attributes (X, Y, Z, R, G, B...) into efficient features for the task at hand. Following PointNet (Qi et al., 2017a) and PointNet++ (Qi et al., 2017b), which are considered as a baseline approach in the community, other work apply deep learning to point set input or voxel representations. The end-to-end framework SEGCloud (Tchapmi et al., 2017) combines a 3D-FCNN, trilinear interpolation, and CRF to provide class labels for

3D point clouds. Their approach is mainly performance-oriented when compared to state-of-the-art methods that are based on neural networks, random forests, and graphical models. Interestingly, they use a trilinear interpolation, which adds an extra boost in performance, enabling segmentation in the original 3D points space from the voxel representation. Landrieu and Simonovsky provide another promising approach for large scale Point Cloud semantic segmentation with Superpoint graphs (Landrieu and Simonovsky, 2018). In the article, the authors propose a deep learning-based framework for semantic segmentation of point clouds. They initially postulate that the organization of 3D point clouds can be efficiently captured by a structure (Superpoint graph), which is derived from a partition of the scanned scene into geometrically homogeneous elements (segments). Their goal is to offer a compact representation of the contextual relationships between object parts to exploit through convolutional network. In essence, the approach is similar to (F. Poux et al., 2017; Poux and Billen, 2019b), through a graph-based representation. Finally, the works of Engelmann et al. in (Engelmann et al., 2018a, 2018b) provide very interesting performances by including the spatial context into the PointNet neural network architecture (Engelmann et al., 2018a) or providing an efficient feature learning and neighbourhood selection strategy (Engelmann et al., 2018b). These works are very inspiring, and they have the potential to become de-facto methodologies for a wide variety of applications through transfer learning. As such, they are an interesting basis for benchmarking semantic segmentation approaches.

In this condensed state-of-the-art review of pertinent related work, we highlighted three different directions that will drive our methodology. First, it is important that we identify the key points in a point cloud that can retain a relevant connotation to domain-related objects. Secondly, we noted that, for gravity-based scenes, these elements have a space continuity and often feature homogeneity best captured through segments. Third, specifically, man-made scenes retain a high proportion of planar surfaces that can host other elements (floor, ceiling, wall ...) (Poux et al., 2016a) and thus the use of ontologies can efficiently describe semantic concepts.

### 3. METHODOLOGY

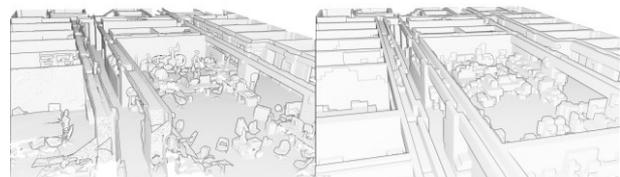
We propose a point cloud instance segmentation to extract semantic clusters (Connected Elements (F. Poux et al., 2017)), which are then specified through application-dependent classes. Our automatic procedure is composed of two independent architectures (1 and 2 as illustrated in **Figure 2** Architecture overview. A raw point cloud goes through three steps that strengthen segments using analytical knowledge, that then serves a domain knowledge injection for instance segmentation.) and is described in the five following sub-sections. In Section 3.1, we describe the voxel-based graph representation. In Section 3.2, we cover feature extraction processes both for low-level and relationship descriptor abstraction. Subsequently, in Section 3.3 we provide a connected-component labelling system using extracted feature sets for the constitution of Connected Elements (Poux and Billen, 2019b). In parallel, we formalize a set of semantic rules (Section 3.4) used in a self-learning process for the ontology-based classification (Section 3.5), routine to obtain a fully labelled point cloud data benchmarked in Section 4.



**Figure 2** Architecture overview. A raw point cloud goes through three steps that strengthen segments using analytical knowledge, that then serves a domain knowledge injection for instance segmentation.

#### 3.1 Voxel-based graph representation

Our approach proposes to integrate different generalization levels in both feature space and spatial space. First, we establish an octree-derived voxel grid over the point cloud, and we store points at the leaf level. As stated in (Poux et al., 2016b; Quan et al., 2018; Truong-Hong et al., 2012), an octree involves recursively subdividing an initial bounding-box into smaller voxels until a depth level is reached. Various termination criteria may be used: the minimal voxel size, predefined maximum depth tree, or a maximum number of sample points within a voxel. In the proposed algorithm, a maximum depth tree is used to avoid the computations necessitating domain knowledge early on. The grid is constructed following the initial spatial frame system of the point cloud to account for complex scenarios where point repartition does not precisely follow the axes. The cubic volume, defined by a voxel entity, provides us with the advantage of fast yet uniform space division (**Figure 3**), and we hence obtain an octree-based voxel structure at a specific depth level. The constituted voxel grid discards empty voxels to only retain points-filled voxels. However, for higher end applications such as pathfinding, the voxel-grid can be used as a negative to look for empty spaces. Subsequently, we construct a directed graph  $\mathcal{G}$  with nodes representing non-empty voxel at a specific octree level.



**Figure 3** On the left the S3DIS dataset point cloud, on the right the extracted voxel structure for a defined octree level.

#### 3.2 Feature Extraction

The first group of low-level features is mainly derived from  $\Sigma$ , our data covariance matrix of points within each voxel for the low memory footprint and fast calculation, which, in our case, we define as:

$$\Sigma = \frac{1}{m-1} \sum_{i=1}^m (X_i - \bar{X})(X_i - \bar{X})^T \quad (1)$$

where  $\bar{X}$  is the mean vector  $\bar{X} = \sum_{i=1}^m p_i$ , and  $p_i$  the  $i^{\text{th}}$  point.

From this high yielding matrix, we derive eigen values and eigen vectors through Singular Value Decomposition (De Lathauwer et al., 2003) to increase the computing efficiency, which firstly correspond to modelling our voxel containment by a plane, showing to largely improve performances. We follow a Principal Component Analysis (PCA) to describe three principal axes describing the point sample dispersion. Thus, we rely heavily on

eigen vectors and eigen values as a feature descriptor at this point. Therefore, their determination needs to be robust. This is why we use a variant of the Robust PCA approach presented in the article (Poux et al., 2018) to avoid miscalculation. We sort eigenvalues  $\lambda_1, \lambda_2, \lambda_3$ , such as  $\lambda_1 > \lambda_2 > \lambda_3$ , where linked eigen vectors  $\vec{v}_1, \vec{v}_2, \vec{v}_3$ , respectively, represent the principal direction, its orthogonal direction, and the estimated plane normal. These indicators, as reviewed in Section 2, are interesting for deriving several eigen-based features (Feng and Guo, 2018), from which we use the omnivariance, planarity and verticality for their good informative description as seen in (Florent Poux et al., 2017; Poux and Billen, 2019a).

There are very few works that deal with explicit relationship feature extraction within point clouds. The complexity and exponential computation to extract relevant information at the point-level mostly justify this. Thus, the second set of proposed feature set is determined at several octree levels. First, we extract a 26-connectivity graph for each leaf voxel, which appoints every neighbour for every voxel. These connectivity's are primarily classified regarding their touch-topology (Clementini and Di Felice, 1997), which either is vertex.touch, edge.touch, or face.touch. Each processed voxel is complemented through new relational features to complement this characterization of voxel-to-voxel topology. Immediate neighbouring voxels are initially studied to extract  $F_g$  (geometrical difference) while using the log Euclidean Riemannian metric, which is a measure of the similarity between adjacent voxels covariance matrices:

$$F_g = \left\| \log \Sigma_{v_i} - \log \Sigma_{v_j} \right\|_F \quad (2)$$

where  $\log(\cdot)$  is the matrix logarithm operator and  $\| \cdot \|_F$  is the Frobenius norm.

Third, we extract four different planarity-based relationships between voxels as presented in (Poux and Billen, 2019a), for Pure Horizontal relationship, Pure Vertical relationship, Mixed relationship and Neighbouring relationship. If two voxels do not hold one of these former constraining relationships but are neighbours, then the associated nodes are connected by an undirected edge without tags. Finally, the number of relationships per voxel is accounted as the edge weights pondered by the type of voxel-to-voxel topology.

This is translated into a multi-set graph representation to give a flexible featuring possibility to the initial point cloud. As such, extended vicinity is then a possible seed/host of new relationships that permit a topology view of the organization of voxels within the point cloud (e.g. Figure 4).

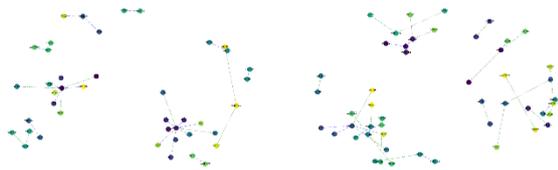


Figure 4 Graph representation within a voxel sample of the point cloud.

These relationships are represented in different groups to extract different features completing the relationship feature set. Graphs are automatically generated through full voxel samples regarding the Category tags.

### 3.3 Unsupervised Segmentation

Based on the feature sets, we create a connected-component workflow that is driven by planar patches. Connected-component labelling is one of the most important processes for image analysis, image understanding, pattern recognition, and computer

vision, and it is reviewed in (He et al., 2017). Being mostly applied for 2D data, we extend it to our 3D octree structure for efficient processing and parallelization compatibility. We study the predominance of planar surfaces in man-made environments and the feature-related descriptor, which provides segmentation benefits. The designed feature representations that are described in Section 3.2 are used as a mean to segment the gridded point cloud into groups of voxels that share a conceptual similarity. These groups are categorized within four different entities: Primary Elements (PE), Secondary elements (SE), transition elements (TE), and remaining elements (RE), as illustrated in Figure 5.

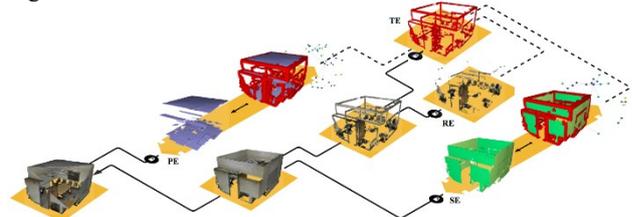


Figure 5 Elements detection and categorization. A point cloud is searched for Primary Elements (PE); the rest is searched for Secondary elements (SE). The remaining from this step is searched for transition elements (TE), leaving remaining elements (RE). TE permits extracting graphs through SF2 analysis with PE, SE, and RE. source (Poux and Billen, 2019a)

We then use bounding-boxes generalization of segments for our instance segmentation workflow.

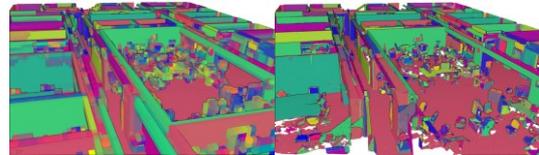


Figure 6 Voxel-based segmentation (left) and point-level refinement.

### 3.4 Semantic model of objects and rule-based reasoning

The shape and relationships between segments represented as a feature list is highly valuable for distinguishing them. On the contrary, local features shared by several segments highlight their belonging to a common group or class. Therefore, the principle of classification consists of gathering segments under a class that best represents them. The process uses an ontology and automatic reasoning to identify classes best representing each segment. This ontology contains semantic models of the different classes to which the segments can belong. Each of these classes is formalized as a "semantic object". The object modelling is composed of the definition of geometric features, the definition of its relationships with other objects, and the definition of remaining features. Geometric features mainly gather information about the shape, the orientation, and the dimensions of an object. The relationships between objects are mainly mathematical relationships (e.g. perpendicular, parallel) and spatial relationships (e.g. touches, contains). The spatial relationship "contains" allows for describing an object as a composition of other objects. Such a relationship is typically used to describe the model of a room, which is composed of walls, a floor, and a ceiling. Finally, the other object features are mainly information about appearance (e.g. colour) and texture (e.g. material, roughness).

Let us take the example of the wall modelling to illustrate the different components of an object modelling. A wall can be initially characterized geometrically as a plane having a horizontal normal, a height of at least two meters, and having a length or width greater than three meters. It has two types of

relationships with a floor: it is on the floor and perpendicular to it. Finally, a wall can have a colour, low roughness, and a mat or reflective material. Such semantic description in OWL2 under the Manchester syntax is described in listing 1.

```
wall:
hasGeometry some (Plane and (hasOrientation only
HorizontalNormal))
and isPerpendicular min 1 Floor
and on min 1 Floor
and (hasHeight exactly 1 xsd:double[>
"2.0"^^xsd:double])
and ((haswidth exactly 1 xsd:double[>=
"3.0"^^xsd:double]) or (hasLength exactly 1
xsd:double[>= "3.0"^^xsd:double]))
and hasColour some Colour
and hasMaterial some (Mat or Reflective)
and hasRoughness some LowRoughness
```

**Listing 1** Example of Wall description in OWL2 under the Manchester syntax.

Each region obtained after the segmentation process is integrated into the ontology as a “segment” with their essential geometric characteristics such as their centroid, their orientation, their dimensions (height, length, width) as well as features described in section 3.2. The segment modelling in OWL2 under the Manchester syntax is described in listing 2.

```
Segment:
(hasHeight exactly 1 xsd:double)
and (hasLength exactly 1 xsd:double)
and (haswidth exactly 1 xsd:double)
and hasCentroid exactly 1 ((hasX exactly 1 xsd:double)
and (hasY exactly 1 xsd:double) and( hasZ exactly 1
xsd:double))
and hasOrientation exactly 1 vector that ((hasX exactly
1 xsd:double) and (hasY exactly 1 xsd:double) and( hasZ
exactly 1 xsd:double))
and hasTangentVectors exactly 1 Vector
and hasBiTangentVectors exactly 1 Vector
```

**Listing 2** Segment description in OWL2 under the Manchester syntax.

As illustrated in listing 2, features extracted during the segmentation process that compose the segment modelling are not best fitting with the object modelling phase. However, the features extracted from each segment provide implicit information that complement their explicit description.

Therefore, after adding segments description obtained from the segmentation, we apply first rule-based reasoning to explicit further information from the base feature set. This rule-based reasoning uses SPARQL with some built-ins to compute new features from other features. For example, the dimension of a segment is deduced from the minimum and maximum points of its bounding box.

Then, the second reasoning on the ontology allows identifying the object corresponding to each segment, used for classification of segments without any training data (J. J. Ponciano et al., 2019).

### 3.5 Automatic Classification

The use of OWL2 to formalize knowledge allows for classifying segments through logical reasoning. Constraints that specify characteristics permit to better define objects. Thus, a segment is classified when it satisfies the constraints of an object. The fit of object constraints means that a segment has all characteristics of a candidate object (class). This explicit classification is carried out by translating the description logics (mainly the “class construct”) of objects into a rule of inference through SPARQL construct queries (J.-J. Ponciano et al., 2019). SPARQL queries provide great flexibility and robustness to work on ontologies composed of millions of triples. For example, Listing 3 presents a translation of the wall’s description logic from Listing 1.

```
CONSTRUCT {?s rdf:type wall}
WHERE{
?s rdf:type Segment .
?s hasGeometry Plane.
?s hasOrientation ?n .
?n rdf:type Horizontal .
?s hasHeight ?h .
FILTER ( ?h > 2 ) .
?s haswidth ?w .
?s hasLength ?d .
FILTER ( ?w >= 3 || ?d >=3 )}
```

**Listing 3** Example of the automatically generated SPARQL construct query for Wall classification.

This automatic classification process consists of two main steps. First, we apply rules that semantically describe each class. The description of these rules depends on expert’s knowledge and its adaptation to the data (device knowledge see (Poux et al., 2016a)). These dependencies often lead to insufficient characterization for a precise classification due to divergences between the expected representation of the classes in the data and the obtained representations. For example, the occlusion of one object by another may cause divergences between the obtained geometry and the expected geometry of the object. Therefore, the second step of automatic classification consists of automatically adapting the semantic rules to the specificities of the data.

This adaptation is performed by an ontology-based self-learning first introduced in (J.-J. Ponciano et al., 2019). This AI learning process uses the results of the first classification step (performed by the application of semantic rules) as a basis for learning to formulate new and more robust rules. The learning consists of analysing the common properties between each of the segments classified with the same object type, to formulate new hypothetical rules. When the properties relate to numerical values (e.g. segment’s size), the learning process calculates the confidence interval (Kalinowski and Fidler, 2010) on the set of values that the regions of a studied class possess. The Equation (3) expresses the confidence interval with  $\bar{x}$  the values mean,  $\delta$  the standard deviation,  $\eta$  the number of values, and  $t_\alpha$  the confidence coefficient.

$$I_c = \left[ \bar{x} - t_\alpha \frac{\delta}{\sqrt{\eta}}; \bar{x} + t_\alpha \frac{\delta}{\sqrt{\eta}} \right] \quad (3)$$

This interval provides significant flexibility for the new rules. Each newly generated rule is then tested on a “fork” of the ontology to determine its validity. The assignment of a validity (expressed as a percentage) to each rule is based on the well-classified class elements expressed through the sum of points of well-classified regions (SPC), the number of newly classified elements expressed through the sum of points of the new regions (SPN), and the number of misclassified elements expressed through the sum of points of misclassified regions (SPM). Equation (4) shows the computation of the validity percentage of a rule according to the different sums of points obtained from the results of the rule application.

$$validity = \frac{SPC + SPN}{SPN + SPM + SPC} \quad (4)$$

The rule providing the highest validity rate is then added to the set of rules, and the classification process is repeated. The automatic classification process is repeated until the rule system becomes idempotent, i.e. until no new rules are generated by the semantic learning process.

## 4. RESULTS

### 4.1 Metrics

Existing literature has suggested several quantitative metrics for assessing the semantic segmentation and classification outcomes. We define the metrics regarding the following terms:

True Positive (TP): Observation is positive and predicted positive; False Negative (FN): Observation is positive but is predicted negative; True Negative (TN): Observation is negative and is predicted to be negative; False Positive (FP): Observation is negative but is predicted positive. Subsequently, the following metrics are used:

$$\overline{IoU} = \frac{TP}{FP + FN + TP} \quad F_{1-score} = \frac{2TP}{2TP + FP + FN} \quad (5)$$

$$precision = \frac{TP}{TP + FP} \quad recall = \frac{TP}{TP + FN} \quad (6)$$

The precision is the ability of the classifier not to label as positive a sample that is negative, the recall is intuitively the ability of the classifier to find all the positive samples. The F1-score can be interpreted as a weighted harmonic mean of the precision and recall, thus giving a good measure of how well the classifier performs. Indeed, global accuracy metrics are not appropriate evaluation measures when class frequencies are unbalanced, which is the case in most scenarios, both in real indoor and outdoor scenes since they are biased by the dominant classes. In general, the Intersection-Over-Union (IoU) metric tends to penalize the single instances of bad classification more than the F1-score, even when they can both agree that this one instance is bad. Thus, the IoU metric tends to have a "squaring" effect on the errors relative to the F1-score. Henceforth, the F1-score in our experiments gives an indication on the average performance of our proposed classifier, while the IoU score measures the worst-case performance.

We did not use any training data and our autonomous approach treats clusters as a set of independent bounding-boxes (Figure 7).

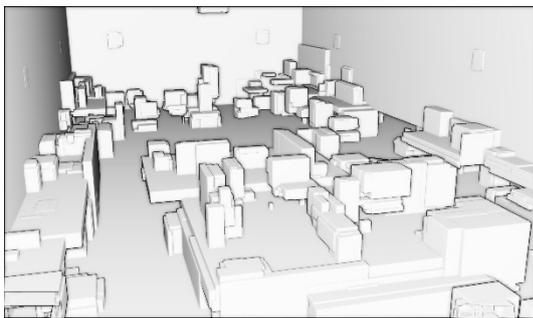


Figure 7 Shape features translated as a set of bounding boxes for each segment fed to the self-learning process

We report results for five major classes, 3 structural (Ceiling, Floor, Walls) as well as 2 furniture (Table and chairs) in Table 1.

Table 1: Benchmark results of our semantic segmentation approach against best-performing deep-learning methods.

$\overline{IoU}$	Ceiling	Floor	Wall	Table	Chair	mIoU 13 c.
PointNet <sup>1</sup>	88.0	88.7	69.3	54.1	42.0	47.6
MS+CU(2) <sup>2</sup>	88.6	95.8	67.3	51.9	45.1	47.8
SegCloud <sup>3</sup>	90.1	96.1	69.9	75.9	70.4	48.9
G+RCU <sup>4</sup>	90.3	92.1	67.9	58.1	47.4	49.7
SPG <sup>5</sup>	92.2	95.0	72.0	65.1	69.5	54.1
KWYND <sup>6</sup>	92.1	90.4	78.5	64.0	61.6	58.3

VB <sup>7</sup>	85.4	92.4	65.2	27.8	23.7	42.2
<b>Ours</b>	<b>80</b>	<b>89</b>	<b>58</b>	<b>41</b>	<b>53</b>	<b>49.9</b>

Compared methods are <sup>1</sup>(Qi et al., 2017a); <sup>2</sup>(Engelmann et al., 2018a); <sup>3</sup>(Tchapmi et al., 2017); <sup>4</sup>(Engelmann et al., 2018a); <sup>5</sup>(Landrieu and Simonovsky, 2018); <sup>6</sup>(Engelmann et al., 2018b); <sup>7</sup>(Poux and Billen, 2019a).

We note that the use of our semantic-guided classification approach compared to the approach used in (Poux and Billen, 2019a) obtains worst results on the structural classes (ceiling, floors, and walls) but better results on the table and chairs furniture. Generally, our approach permits to obtain a non-weighted IoU of 49.9 averaged over 13 classes (Ceiling, Floor, Wall, Beam, Column, Window, Door, Table, Chair, Sofa, Bookcase, Board, and Stairs) on the full S3DIS dataset, compared to an average score of 42.2 for (Poux and Billen, 2019a). This permit the approach to get overall scores slightly above G+RCU but under SPG and KWYND, thus in the top 3-tier of benchmarked deep learning approaches. Generally, it scores lower for planar dominant classes but shows a lesser variance across classes.

To get better insights on its performances, we present in Table 2 the associated precision and recall scores over the full S3DIS dataset.

Table 2 Per class metrics for the S3DIS dataset using our approach.

	Ceil.	Floor	Wall	Door	Table	Chair	Tot.
Prec.	1	1	1	1	1	1	1
Rec.	0.80	0.89	0.58	0.28	0.41	0.53	0.58
F1-s.	0.88	0.94	0.74	0.43	0.57	0.69	0.71

We note that the precision is maximal for each of the studied classes. Such precision is obtained thanks to the segmentation strategy and the self-learning ontology-based process that allows for adapting and refining the rules used to classify the regions, according to the data used. An illustration for Area 1 at the feature generalization level (bounding-boxes) is illustrated in Figure 8.

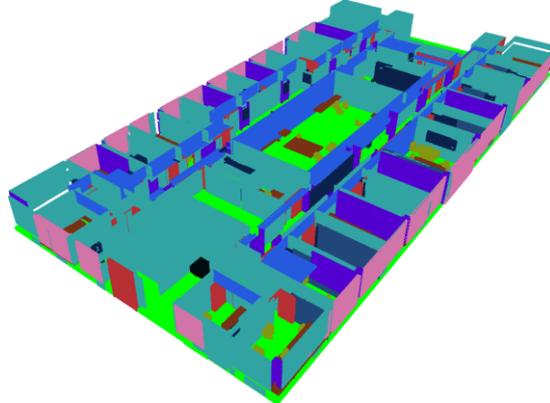
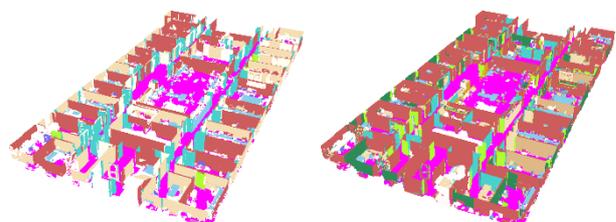
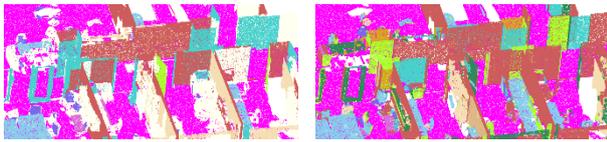


Figure 8 Illustration of the results on the bounding-boxes generalization of each segments for 13 classes of the Area 1 in the S3DIS dataset

These are then back-projected to the segments and constituting points to get predictions at the point level as seen in Figure 9.



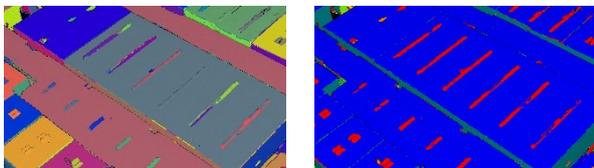


**Figure 9** Classification visual results for furniture and structural elements (left), and the self-learning refinement results for structural elements (right)

The recall values shown in **Table 2** vary between 0.89 (floor) and 0.28 (Doors) mainly due to the “over-classification” challenge of regions (a region is predicted to belong to several classes). This is explained by a current description of the classes too similar. This resemblance is due to a lack of contrasting criteria for accurately defining the classes separation. In future work, we will study the addition of other characteristics such as relational links between regions, to allow a more distinct definition of each class.

## 5. CONCLUSION AND FUTURE WORKS

In this article, we provide a framework for automatic object extraction within 3D point clouds using an unsupervised segmentation and an ontology-based classification reinforced by self-learning. Results are illustrated in Figure 10.



**Figure 10** Segment instances for each class (left) and their associated class (right)

This framework groups points in a voxel-based space, where each voxel is studied by analytic featurizing and similarity analysis to define the semantic clusters that retain highly representative signatures. This process is conducted regarding an initial connected component from multi-composed graph representations after automatically detecting different planar-dominant elements leveraging their prevalence in man-made environments. These clusters are then integrated into an ontology containing the knowledge about the different classes the clusters may have. They are then classified through the application of logical reasoning on the ontology to provide a first set of classification. This set is then used in a self-learning process based on the ontology to automatically adapt the knowledge of the classes defined in the ontology to the specificities of the processed data. This adaptation is performed by analysing the extracted characteristics common to each cluster firstly classified in the same category.

The result obtained by this detection framework in the S3DIS dataset produces a 99.99% accuracy for the planar-dominant classes, surpassing the best deep-learning approach studied in literature. While our dedicated approach was tested on the S3DIS dataset, it can easily be adapted to other point clouds that provide an additional research direction. The approach will be tested against indoor and outdoor point clouds from different sensors and the classification can be adapted to account for various well-established classes. As such, a large effort is currently undergoing to create accurate labelled datasets for AEC and outdoor 3D mapping applications, to be shared as open data.

Our goal is to provide a powerful framework that should be able to adapt to different levels of generalization. As such, our unsupervised segmentation approach combined with automatic classification using an ontology-based self-learning process

allows for an automatic point cloud labelling that is easy to integrate in workflows. Future work will also dive in optimizing and refining its performances for better results. Our focus is driven by a general global/local contextualization of digital 3D environments, where we aim at providing a flexible infrastructure that should be able to scale up to different generalization levels.

## REFERENCES

- Antoniou, G., Harmelen, F. Van, 2004. A semantic web primer.
- Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S., 2016. 3D Semantic Parsing of Large-Scale Indoor Spaces, in: Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, United States, pp. 1534–1543. <https://doi.org/10.1109/CVPR.2016.170>
- Ben-Shabat, Y., Avraham, T., Lindenbaum, M., Fischer, A., 2018. Graph based over-segmentation methods for 3D point clouds. *Computer Vision and Image Understanding* 174, 12–23. <https://doi.org/10.1016/j.cviu.2018.06.004>
- Ben-Shabat, Y., Lindenbaum, M., Fischer, A., 2017. 3D Point Cloud Classification and Segmentation using 3D Modified Fisher Vector Representation for Convolutional Neural Networks. *arXiv.org*.
- Ben Hmida, H., Boochs, F., Cruz, C., Nicolle, C., 2012. Knowledge Base Approach for 3D Objects Detection in Point Clouds Using 3D Processing and Specialists Knowledge. *International Journal on Advances in Intelligent Systems* 5, 1–14.
- Clementini, E., Di Felice, P., 1997. Approximate topological relations. *International Journal of Approximate Reasoning* 16, 173–204. [https://doi.org/10.1016/S0888-613X\(96\)00127-2](https://doi.org/10.1016/S0888-613X(96)00127-2)
- De Lathauwer, L., De Moor, B., Vandewalle, J., 2003. A Multilinear Singular Value Decomposition. *SIAM Journal on Matrix Analysis and Applications* 21, 1253–1278. <https://doi.org/10.1137/s0895479896305696>
- Dietenbeck, T., Torkhani, F., Othmani, A., Attene, M., Favreau, J.-M.M., 2017. Multi-layer ontologies for integrated 3D shape segmentation and annotation, in: *Studies in Computational Intelligence*. Springer, Cham, pp. 181–206. [https://doi.org/10.1007/978-3-319-45763-5\\_10](https://doi.org/10.1007/978-3-319-45763-5_10)
- Engelmann, F., Kontogianni, T., Hermans, A., Leibe, B., 2018a. Exploring Spatial Context for 3D Semantic Segmentation of Point Clouds, in: *International Conference on Computer Vision (ICCV)*. IEEE, Istanbul, Turkey, pp. 716–724. <https://doi.org/10.1109/ICCVW.2017.90>
- Engelmann, F., Kontogianni, T., Schult, J., Leibe, B., 2018b. Know What Your Neighbors Do: 3D Semantic Segmentation of Point Clouds, in: *European Conference on Computer Vision (ECCV)*. Munich, Germany.
- Feng, C.C., Guo, Z., 2018. Automating parameter learning for classifying terrestrial LiDAR point cloud using 2D land cover maps. *Remote Sensing* 10, 1192. <https://doi.org/10.3390/rs10081192>
- He, L., Ren, X., Gao, Q., Zhao, X., Yao, B., Chao, Y., 2017. The connected-component labeling problem: A review of state-of-the-art algorithms. *Pattern Recognition* 70, 25–43. <https://doi.org/10.1016/J.PATCOG.2017.04.018>
- Horrocks, I., Patel-Schneider, P., Boley, H., Tabet, S., Grosz, B., Dean, M., 2004. SWRL: A Semantic Web Rule Language Combining OWL and RuleML [WWW Document]. W3C members. URL [http://www.academia.edu/download/30680504/SWRL\\_A\\_Semantic\\_Web\\_Rule\\_Language\\_Combining\\_OWL\\_and\\_RuleM....pdf](http://www.academia.edu/download/30680504/SWRL_A_Semantic_Web_Rule_Language_Combining_OWL_and_RuleM....pdf) (accessed 5.2.20).
- Kalinowski, P., Fidler, F., 2010. Interpreting Significance: The

- Differences Between Statistical Significance, Effect Size, and Practical Importance. *Newborn and Infant Nursing Reviews* 10, 50–54. <https://doi.org/10.1053/j.nainr.2009.12.007>
- Koffka, K., 2013. *Principles of Gestalt psychology*. Routledge.
- Landrieu, L., Simonovsky, M., 2018. Large-Scale Point Cloud Semantic Segmentation with Superpoint Graphs, in: *Conference on Computer Vision and Pattern Recognition (CVPR)*. Salt Lake City, United States, pp. 4558–4567. <https://doi.org/10.1109/CVPR.2018.00479>
- Liu, Y.-S., Ramani, K., 2009. Robust principal axes determination for point-based shapes using least median of squares. *Computer aided design* 41, 293–305. <https://doi.org/10.1016/j.cad.2008.10.012>
- Nguyen, C., Starek, M.J., Tissot, P., Gibeau, J., 2018. Unsupervised clustering method for complexity reduction of terrestrial lidar data in marshes. *Remote Sensing* 10, 133. <https://doi.org/10.3390/rs10010133>
- Ni, H., Lin, X., Zhang, J., Ni, H., Lin, X., Zhang, J., 2017. Classification of ALS Point Cloud with Improved Point Cloud Segmentation and Random Forests. *Remote Sensing* 9, 288. <https://doi.org/10.3390/rs9030288>
- Ponciano, J.-J., Trémeau, A., Boochs, F., 2019. Automatic Detection of Objects in 3D Point Clouds Based on Exclusively Semantic Guided Processes. *ISPRS International Journal of Geo-Information* 8, 442. <https://doi.org/10.3390/ijgi8100442>
- Ponciano, J.J., Karmacharya, A., Wefers, S., Atorf, P., Boochs, F., 2019. Connected Semantic Concepts as a Base for Optimal Recording and Computer-Based Modelling of Cultural Heritage Objects, in: *RILEM Bookseries*. Springer Netherlands, pp. 297–304. [https://doi.org/10.1007/978-3-319-99441-3\\_31](https://doi.org/10.1007/978-3-319-99441-3_31)
- Poux, F., Billen, R., 2019a. Voxel-Based 3D Point Cloud Semantic Segmentation: Unsupervised Geometric and Relationship Featuring vs Deep Learning Methods. *ISPRS International Journal of Geo-Information* 8, 213. <https://doi.org/10.3390/ijgi8050213>
- Poux, F., Billen, R., 2019b. A Smart Point Cloud Infrastructure for intelligent environments, in: Lindenbergh, R., Belen, R. (Eds.), *Laser Scanning: An Emerging Technology in Structural Engineering*, ISPRS Book Series. Taylor & Francis Group/CRC Press, United States. <https://doi.org/in-generation>
- Poux, F., Hallot, P., Neuville, R., Billen, R., 2016a. SMART POINT CLOUD: DEFINITION AND REMAINING CHALLENGES. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences IV-2/W1*, 119–127. <https://doi.org/10.5194/isprs-annals-IV-2-W1-119-2016>
- Poux, F., Neuville, R., Hallot, P., Billen, R., 2017. MODEL FOR SEMANTICALLY RICH POINT CLOUD DATA. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences IV-4/W5*, 107–115. <https://doi.org/10.5194/isprs-annals-IV-4-W5-107-2017>
- Poux, F., Neuville, R., Hallot, P., Billen, R., 2016b. Point clouds as an efficient multiscale layered spatial representation, in: Vincent, T., Biljecki, F. (Eds.), *Eurographics Workshop on Urban Data Modelling and Visualisation*. The Eurographics Association, Liège, Belgium. <https://doi.org/10.2312/udmv.20161417>
- Poux, F., Neuville, R., Nys, G.-A., Billen, R., 2018. 3D Point Cloud Semantic Modelling: Integrated Framework for Indoor Spaces and Furniture. *Remote Sensing* 10, 1412. <https://doi.org/10.3390/rs10091412>
- Poux, Florent, Neuville, R., Van Wersch, L., Nys, G.-A., Billen, R., 2017. 3D Point Clouds in Archaeology: Advances in Acquisition, Processing and Knowledge Integration Applied to Quasi-Planar Objects. *Geosciences* 7, 96. <https://doi.org/10.3390/geosciences7040096>
- Qi, C.R., Su, H., Mo, K., Guibas, L.J., 2017a. PointNet: Deep learning on point sets for 3D classification and segmentation, in: *Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, Hawaii, United States, pp. 77–85. <https://doi.org/10.1109/CVPR.2017.16>
- Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017b. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space, in: *Conference on Neural Information Processing Systems (NIPS)*. Long Beach, United States.
- Quan, S., Ma, J., Hu, F., Fang, B., Ma, T., 2018. Local voxelized structure for 3D binary feature representation and robust registration of point clouds from low-cost sensors. *Information Sciences* 444, 153–171. <https://doi.org/10.1016/J.IINS.2018.02.070>
- Rusu, R.B., Blodow, N., Beetz, M., 2009. Fast Point Feature Histograms (FPFH) for 3D registration, in: *International Conference on Robotics and Automation (ICRA)*. IEEE, Kobe, Japan, pp. 3212–3217. <https://doi.org/10.1109/ROBOT.2009.5152473>
- Son, H., Kim, C., 2017. Semantic as-built 3D modeling of structural elements of buildings based on local concavity and convexity. *Advanced Engineering Informatics* 34, 114–124. <https://doi.org/10.1016/j.aei.2017.10.001>
- Tchapmi, L.P., Choy, C.B., Armeni, I., Gwak, J., Savarese, S., 2017. SEGCloud: Semantic Segmentation of 3D Point Clouds, in: *International Conference on 3D Vision (3DV)*. Qingdao, China.
- Truong-Hong, L., Laefer, D.F., Hinks, T., Carr, H., 2012. Flying Voxel Method with Delaunay Triangulation Criterion for Façade/Feature Detection for Computation. *Journal of Computing in Civil Engineering* 26, 691–707. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000188](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000188)
- Wang, J., Lindenbergh, R., Menenti, M., 2017. SigVox – A 3D feature matching algorithm for automatic street object recognition in mobile laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing* 128, 111–129. <https://doi.org/10.1016/j.isprsjprs.2017.03.012>
- Wang, Y., Cheng, L., Chen, Y., Wu, Y., Li, M., 2016. Building point detection from vehicle-borne LiDAR data based on voxel group and horizontal hollow analysis. *Remote Sensing* 8, 419. <https://doi.org/10.3390/rs8050419>
- Weber, C., Hahmann, S., Hagen, H., 2010. Sharp feature detection in point clouds, in: *International Conference on Shape Modeling and Applications*. IEEE, Washington, United States, pp. 175–186. <https://doi.org/10.1109/SMI.2010.32>
- Xu, Y., Hoegner, L., Tutas, S., Stilla, U., 2017. Voxel- and Graph-Based Point Cloud Segmentation of 3D Scenes Using Perceptual Grouping Laws, in: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. ISPRS, Hannover, Germany, pp. 43–50. <https://doi.org/10.5194/isprs-annals-IV-1-W1-43-2017>
- Xu, Y., Tutas, S., Hoegner, L., Stilla, U., 2018. Voxel-based segmentation of 3D point clouds from construction sites using a probabilistic connectivity model. *Pattern Recognition Letters* 102. <https://doi.org/10.1016/j.patrec.2017.12.016>
- Zhu, Q., Li, Y., Hu, H., Wu, B., 2017. Robust point cloud classification based on multi-level semantic relationships for urban scenes. *ISPRS Journal of Photogrammetry and Remote Sensing* 129, 86–102. <https://doi.org/10.1016/j.isprsjprs.2017.04.022>