# Integrating Ontologies and Object Compositional Hierarchies for Dynamic Semantic Segmentation of Point Clouds

Matteo Codiglione[1,2], Samuele Facenda[1], Fabio Remondino[1]

[1] 3D Optical Metrology (3DOM) unit, Bruno Kessler Foundation (FBK), Trento, Italy – (mcodiglione, sfacenda, remondino)@fbk.eu
[2] University of Trento, Department of Information Engineering and Computer Science (DISI), Trento, Italy

**Keywords:** Semantic Segmentation, 3D Point Clouds, Ontology, Taxonomy, Classification

**Abstract**

While there is a continuous progress on the 3D classification of point clouds, no effort has been seen in in maximizing the amount of semantic information which can be extracted from a single classification operation. This line of research is indeed complementary to the traditional semantic segmentation, enhancing its native outputs with increased flexibility and semantic structuring. Following this direction, this paper proposes a *Dynamic Semantic Segmentation* (DSS), an approach that lets a single classified 3D point cloud be visualised at multiple semantic granularities instead of at a fixed one. The method consists of three interlocking ideas. First, each point is allowed to carry several distinct labels at once, supporting its belonging to multiple object instances. Second, the spatial overlaps between these co-occurring labels are exploited to establish part-whole (mereological) relationships between object instances, yielding a compositional reading of the scene. Third, the labels are organised into a taxonomy and extended to their ancestors, yielding a conceptual reading. In this way, the semantic information needed at every scale is present simultaneously rather than committed to at classification time. The proposed approach builds upon the 3DGraph format and the 3DOnt framework, using an RDF backbone ontology to encode the taxonomy. Results show the potential of the method and its replicability to other scales and scenarios. Because all labels are pre-computed, each segmentation is produced within few seconds on a point cloud with some million points, adding expressive power over the static counterpart at no cost. Further information and visual results about the 3DOnt framework and DSS are available at: https://3dom.fbk.eu/projects/3DOnt.

## 1. Introduction and Motivation

3D point clouds encode fine-grained geometric representations of real-world assets and this geometry is usually enriched through semantic segmentation, conventionally consisting in a many-to-one mapping that assigns each point a single label (Sarker et al., 2024). This single-labelling approach fixes the semantic granularity once and for all, yet real applications might benefit from enhanced semantic flexibility. Smart-city analysis alone spans tasks from the macro-scale (e.g. vegetation-to-built-area ratios) to the micro-scale (e.g. management of urban lighting infrastructure), each calling for a different reading of the same scene. Since classification is a costly step - slow if manual, technically demanding if automated - re-classifying for every granularity is extremely sub-optimal. In this context, the possibility to *a posteriori* extract multiple granularities from a single classification step would be a clear gain. This work delivers this possibility - under the name of *Dynamic Semantic Segmentation* (DSS) - by decoupling the moments of *classification* and of *segmentation* (Section 4) and using the former's output to build semantic structures from which multiple segmentations can be assembled on demand at the required granularities. The approach is built with three intwined features:

1. ***Multi-Labelling*** (Section 3.1): each point is allowed to carry multiple labels simultaneously.
2. ***Mereological Structure*** (Section 3.2): the spatial overlaps among these co-occurring labels are then used to infer part-whole relationships between objects, giving the scene a compositional, mereological structure.
3. ***Taxonomical Structure*** (Section 3.3): in parallel, the labels are arranged in a taxonomy and propagated to their ancestors, forming a hierarchical structure of concepts.

The mereological and taxonomical structures form two orthogonal axes which can be independently traversed at different degrees of generality. Semantic segmentation becomes a parametric function *S(Mr, Tr)* over such structures, generated from a single classification step.

Taken in their combination within the introduced DSS approach, the three abovementioned features turn a classified point cloud into a queryable, multi-resolution semantic resource. Building on the 3DGraph point cloud format (Codiglione et al., 2024a; 2024b), we include the approach in the 3DOnt framework and demonstrate it on the YTU3D urban-scale dataset (Figure 1a) and on the Temple of Neptune building-scale one (Figure 1b).

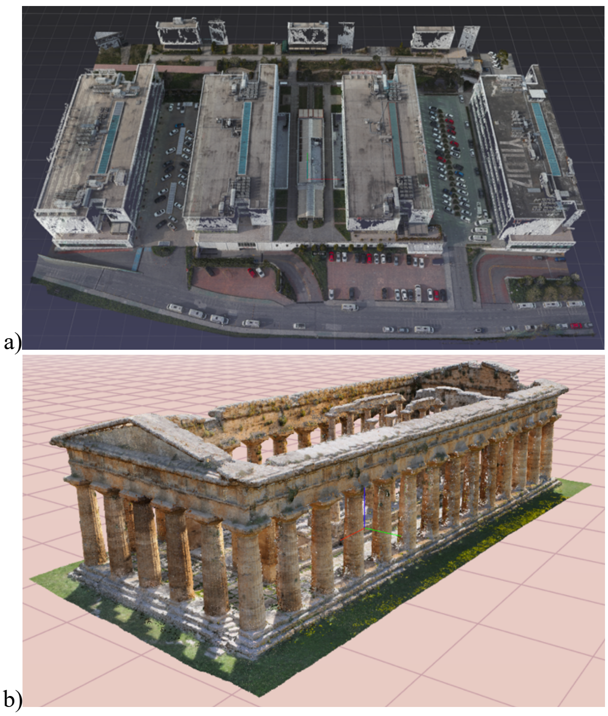
a)
b)


Figure 1: RGB point cloud of the Yildiz Technical University - YTU (Turkey)[1](a) and the Temple of Neptune in Paestum (Italy) (b).

[1] https://github.com/3DOM-FBK/YTU3D

## 2. Related Works

Semantic segmentation of 3D point clouds is a mature field, which has seen significant research efforts. In its dominant formulation, semantic segmentation encodes a many-to-one point-to-label mapping in a 3D point cloud dataset, with the aim of enriching its informational content beyond sheer geometries (Armeni et al., 2016; Hackel et al., 2017; Behley et al., 2019; Zhang et al., 2019; Xie et al., 2020; Guo et al., 2021; Liu et al., 2026; Song et al., 2026). Automatically assigning semantic labels to points is itself an active and demanding line of research, from the seminal point-based networks to large urban-scale benchmarks (Qi et al., 2017a; Qi et al., 2017b; Hu et al., 2021; Liu et al., 2026; Song et al., 2026). Nonetheless, the intrinsic rigidity of this established formulation of the task, which leads to a single fixed semantic granularity, hinders its practical usability. Two complementary responses to this limitation are relevant here. On the one side, layered and interoperable semantic city models organise urban concepts at multiple levels of detail, following a line of researcher paved by the work of Gröger et al. (2012). On the other, ontologies have been used to drive point cloud analysis and queries, both for indoor scenes (Stojanovic et al., 2020) and, more recently, in a domain-agnostic fashion through the 3DGraph format, which represent point clouds as RDF knowledge graphs (Codiglione et al., 2024a; 2024b). Our work builds directly on this latter line, adopting the 3DGraph format as a starting point. Then, a further body of theory underpins the approach. Part-whole (mereological) relations have an established formal treatment in cognitive science and logic (Winston et al., 1987), allowing for a structured representation of objects constitutional hierarchies within a 3D scene. The present contribution can be read as bringing mereology on top of an ontology-backed point cloud format, with the aim to enable Dynamic Semantic Segmentation (DSS; Section 4).

## 3. Theoretical Framework

### 3.1 Multi-Labelling

Conventional semantic segmentation encodes a many-to-one mapping as each 3D point carries exactly one label among the multiple available ones. The hereby introduced Multi-label classification generalises this to a many-to-many mapping, in which a single point may simultaneously belong to several leaf classes (e.g. a point may be at once *Solar Panel*, *Roof* and *Building*). This is achieved by providing, for each class, a binary classification ($A, \neg A$ for each class $A$) scalar field. A many-to-many labelling can be obtained in two ways:

- ***Native Multi-Label Classification***: independently performing, for each considered class, a binary classification of every point as belonging or not to that class.
- ***Prior-Based Mereological Class Aggregation***: starting from a single-labelled dataset, the existing labels are aggregated under explicit priors about which kinds of object instances can be considered as together constituting a composite one of a different kind. Practically, this consists in producing new binary scalar fields for the composite classes as the logical unions of the ones dedicated to the classes identified as constituting the former. This route, followed in the present work, is detailed in Section 5.2.

### 3.2 Mereological Structure

Mereological structuring leverages spatial overlaps between labels (allowed by multi-labelling) to establish compositional (part-of) relationships between *objects*. This structuring starts by executing, for each label, a connected-component-based instance segmentation on the subset of points presenting such a label. In this way, clusters of labels become *objects*. Then, object $O_1$ is said to be part of object $O_2$ if at least a certain ratio $\phi$ (e.g., 0.95) of the points constituting $O_1$ also constitutes $O_2$, which is to say, being $P(O_n)$ the set of points constituting $O_n$:

$$\frac{|P(O_1) \cap P(O_2)|}{|P(O_1)|} > \phi \tag{1}$$

Chains of objects, each part of the next, are thereby formed. These chains define the Mereological Resolution *Mr*: an upward movement is part → whole oriented, a downward movement whole → part oriented. The length of the longest chain in the scene fixes the range of *Mr*. The triviality of these chains depends on the labelling choice of Section 3.1. Under native multi-label classification, the part-whole links are inferred dynamically from region overlap and carry no class constraint, thus being able to capture unexpected spatial overlaps between instances (e.g., a solar panel on the roof of a truck) or to avoid inadequate assumptions (e.g., an unexpectedly detached window frame, for instance, would not be counted as part of a façade). Under prior-based aggregation, spatial overlaps between instances directly and rigidly reflect the aggregation priors (Section 3.1) and are therefore statically determined.

### 3.3 Semantic Taxonomical Structure

Whereas mereological structure relates *objects*, semantic taxonomical structure organises the *classes* conceptually. The associated semantic labels are arranged into a single rooted tree ordered by subsumption (subclass relationship): leaf classes are the fine-grained categories corresponding to the labels produced during classification, while internal nodes are higher-level abstractions that do not need to exist in the original label set. Moving up the tree is a conceptual generalisation - a set of sibling classes is replaced by their shared ancestor - and moving down is a conceptual specialisation that restores the finer distinctions. This tree defines the taxonomical resolution *Tr*: lowering *Tr* climbs the tree toward more general classes, raising it descends toward more specific ones. The range of *Tr* is fixed by the maximum depth (number of nodes met while going from a leaf to the root) of taxonomical leaves. The taxonomy answers only the question "*what kind of thing is this?*"; it says nothing about how things are physically composed, which is the concern of the mereological structure.

## 4. Dynamic Semantic Segmentation (DSS)

The three notions above let us decouple two moments that the conventional pipeline treats as one. There, classification (the annotation step) and segmentation (the determination of which label must be associated to each point for visualization) coincide and the output is a single fixed view. We instead keep classification as a one-off multi-label annotation and turn segmentation into a parametric function *S(Mr, Tr)*.

The mereological resolution *Mr* controls part-whole granularity - how deep the segmentation descends into the compositional chains of the scene - and the taxonomical resolution *Tr* controls conceptual granularity - how general or specific the displayed classes are. Each classified dataset is therefore associated not with one segmentation but with a two-dimensional space of them: every *(Mr, Tr)* pair yields a distinct segmentation, with its own legend and its own granularity. The extent of this space is set by the depth of the taxonomy (for *Tr*) and by the longest mereological chain in the scene (for *Mr*). We define this approach as Dynamic Semantic Segmentation (DSS).

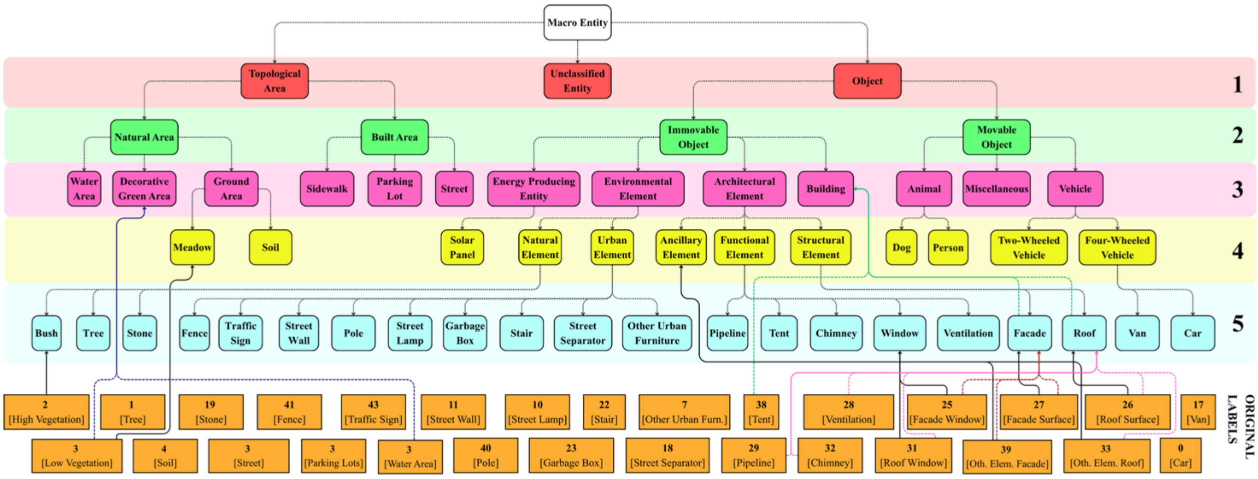



Figure 2: The original labels (orange) together with the taxonomy. Only non-trivial mappings have been highlighted (continuous black arrows). Dotted lines represent subclass relationships, coloured dashed lines show prior-based mereological aggregation choices.

## 5. Data and Methodology

### 5.1 Data

We primarily use the YTU3D urban dataset (Bayrak et al., 2023; Figure 1) to validate the proposed approach of DSS. Data are represented as a 3DGraph (Codiglione et al., 2024a, 2024b), a format that stores a point cloud as an RDF knowledge graph instantiated over a dedicated backbone ontology, with the semantic taxonomy encoded in the latter. YTU3D is natively single-labelled: each point carries a single integer code in its classification field. The proposed approach, however, needs a many-to-many labelling. The next sections describe how we obtain such many-to-many labelling, following the second of the possibilities described in Section 3.1.
The second dataset is the Temple of Neptune (Grilli and Remondino, 2019) and is employed to complement the validation of the proposed procedure under two distinct regards: (i) to provide a building-scale example alongside the urban-scale one; (ii) to show that the effectiveness of the approach is maintained even in a scene where taxonomy is not articulated, only leveraging mereological structures.

### 5.2 From Single-Label to Multi-Label by Class Aggregation

Because YTU3D is single-labelled, native multi-labelling (Section 3.1) is not available and the many-to-many mapping must be reconstructed from the existing labels. We do this by prior-based class aggregation, a twofold process: first a taxonomical mapping (Section 5.2.1), which places each raw code in the semantic taxonomy and then a mereological class aggregation (Section 5.2.2), which declares the compositional wholes the scene may contain. Together they produce, from a single-label source, the multi-labelled spatially overlapped representation the method consumes.

**5.2.1 Taxonomical Mapping**. Firstly, we map each raw label of YTU3D to the leaves of the customized taxonomy described in Figure 2 (e.g. 0 -> Car, 26 -> Roof, 30 -> Solar Panel) Figure 2 presents non-trivial mapping links. Beyond renaming labels, this process does three things:

1. ***Expliciting hidden semantic content***. Labels contained in a non-taxonomical organization carry no relational information on their own. Mapping them to a taxonomy helps in definition (e.g., Low Vegetation and High Vegetation are now mapped to *Meadow* and *Bush* respectively, where the former is a subclass of *Topological Surface* and the latter of *Object*).
2. ***Merging redundancies***. Some concepts were split across two codes (labels) only because the source tagged them by location. Facade Window (25) and Roof Window (31) both denote a window and map to a single *Window* leaf; The facade-vs-roof distinction is extrinsic to what the object is, thus it is ignored by the taxonomy. Nonetheless, this information is not discarded, rather, it is more correctly encoded in the mereological structure, which deals with spatial locations rather than objects identities and is more suitable to model these extrinsic features.
3. ***Extending label scope***. The original Facade Surface and Roof Surface codes were defined negatively - "the part of the facade/roof that is not one of its elements". Mapping them simply to Facade and Roof removes that exclusion, so each leaf denotes the full concept; the surface-vs-elements distinction is then re-expressed, where it belongs, as a part-whole relation in the mereology.

Only the leaf classes are considered during classification; every internal node is an abstraction introduced solely within the taxonomy and these internal nodes are exactly what *Tr* traverses.

**5.2.2 Mereological Class Aggregation.** Secondly, we define priors about mereological relationships. This consists in selecting labels *a,b,c,…n* with instances that will always be said as being parts of instances of class *A*. Concretely, this consists in generating a new scalar field for class *A* and in producing its values from a logical union of the fields associated to *a,b,c,…n*. This aggregation applies to original labels, not to taxonomical leaves, to recover and leverage the extrinsic spatial information ignored during taxonomical mapping (Section 5.2.1). Dashed lines in Figure 2 (different colours for different aggregated classes) show aggregation choices. Four aggregation classes are considered:

- ***Decorative Green Area*** (blue dashed lines)
- ***Roof*** (pink dashed lines);
- ***Façade*** (red dashed lines)
- ***Building*** (green dashed lines, which in the case of *Roof* and *Facade* ideally aggregate pink and red lines involved for facade and roof aggregation, which are not repeated for the sake of visual clarity. The fact that roofs and facades are themselves composite objects illustrates nesting possibilities of mereological structures)

In particular, this step is not yet mereological structuring: the latter only occurs later, once instance segmentation and overlap-based linking operate on multi-label data. The discussed step is merely a preprocess that manufactures multi-labelling by generating overlapping binary scalar fields (one uint8 mask per class), so that a point may carry several masks at once (e.g., a solar-panel point is flagged *Solar Panel*, *Roof* and *Building* together due to aggregation choices about the first being aggregated into the others and the second being aggregated into the latter). Crucially, the aggregation choices are free to traverse the taxonomy in any direction: a compositional whole may group taxonomic siblings (*Roof* and *Facade* into *Building*), classes lying in different branches (*Meadow*, under *Ground Area* and *Water Area* into *Decorative Green Area*), or a class that descends from a wholly unrelated node (*Solar Panel*, under *Energy Producing Element*, into the architectural wholes *Roof* and *Building*). Because membership in a whole is decided on spatial-functional grounds rather than by conceptual subsumption, the *part-of* organisation is left independent of the *is-a* one - which is exactly what allows *Mr* and *Tr* to be navigated independently. After this step is executed, 3DOnt-based processes (i.e., instance, segmentation, mereological structuring, 3DGraph generation) can finally take place.

### 5.3 Parametric Implementation of the DSS

The parametric implementation of the proposed DSS is kept lightweight by the organisation of the 3DOnt framework: *S(Mr, Tr)* is split into its mereological and taxonomical parts, computed independently. A first SPARQL query returns, for each point, all the labels it carries. This includes all the labels of the leaf classes to which it belongs (which together represent the mereological chain which the point participates) and all their super classes (given the transitivity of subclass relationships). The latter are excluded and *Mr* is then used to select, per point, one label among the remaining ones, the one at the desired position of the mereological chain. If *M*r is larger than the length of the chain, the smallest object is considered. A second query then builds a dictionary having leaf classes as keys and ordered lists of ancestors (leaf included) as values. For each point, *Mr*-selected label is used to find the right ancestor list, which is then traversed using *Tr* as index to find the final label to be shown for each point. At lower *Tr*, this returns a more general ancestor, while at maximum *Tr* this acts as the identity. The requested taxonomical resolution is capped at the depth actually available along each point's branch: if *Tr* asks for a level deeper than the deepest class present for a point, the mapping saturates at that class rather than inventing a distinction. Figure 2 displays depth layers with the numbers on the right. This process is applied to every point with linear complexity, adding negligible overhead over standard query processing.

## 6. Results

Visual examples for different positions in the *(Mr, Tr)* space are shown in Figure 3, with legends in Table 1 (Appendix). Maximum resolution on both axes (Mereological resolution - Mr and Taxonomical resolution - Tr) reproduces the original single-label classification (Figure 3a). Lowering *Mr* alone yields a spatially coarser segmentation (Figure 3b; Mr=2, Tr=5), while lowering *Tr* alone yields a conceptually more general one (Figure 3e); the latter also loses spatial resolution, since adjacent sibling labels merge under a shared abstraction. This loss of spatial resolution is not trivially the same seen in Figure 3b. Lowering both parameters gives the simplest segmentation (Figure 3c; Mr=1, Tr=1), with the most general classes and the coarsest objects. In this dataset setting *Tr* to its minimum overrides the choice of *Mr*, because above a certain level all classes whose instances take part in part-whole chains fall under the same side of the first conceptual division (*Topological Area* vs *Object*). Figure 3e (Mr=3, Tr=3) shows the non-triviality of the method: lowering *Tr* makes the solar panels emerge from the rest of the building as the only *Energy Producing Elements* - a higher-level abstraction with no counterpart in the original labels. Figures 3d (Mr=1, Tr=5), 3f (Mr=3, Tr=2) and 3h (Mr=1, Tr=4) provide four further *(Mr, Tr)* pairs.

### 6.1 Computational Cost

As every label is pre-computed and materialised as a scalar field at pre-processing time, the segmentation queries perform no classification at runtime. Furthermore, since semantic information is encoded in the 3DGraph in the form of RDF properties and since the 3DGraph is stored in an indexed triple store, SPARQL queries can be efficiently executed. On the portion of YTU3D used here - roughly four million points - a complete segmentation for a given *(Mr, Tr)* pair is produced in about 1-2 seconds, further reducible by caching the taxonomical mapping and batching the per-point assignment. On the hand, under the static pipeline, each additional view would require repeating classification - hours of manual annotation - for every new granularity. The proposed method thus turns each new semantic view from a fresh classification effort into a near-instantaneous query over an unchanged dataset.

### 6.2 Aggregation Decisions Enabled by Dual-Resolution Control

By traversing the (Mr, Tr) space, the proposed method lets a domain expert take qualitatively distinct kinds of semantic insights on one and the same classified dataset, illustrated below by three exemplary pairs of possible semantic segmentations.
The first example (Figure 4) shows a conceptual abstraction along the taxonomical axis, whether to keep fine-grained sibling classes apart or to merge them under a shared ancestor. The result compares a segmentation in which *Car* and *Van* are distinguished and their points are shown with different colours (Figure 4a) from one, obtained at a lower Tr, where both objects collapse into the generic *Vehicle* abstraction and are rendered uniformly (Figure 4b). The choice between a vehicle-type inventory and a generic traffic-occupancy view is thereby deferred to query time rather than fixed during classification.
The second example (Figure 5) shows a compositional aggregation along the mereological axis. Figure 5 compares a segmentation in which *Meadow* and *Water Area* appear as distinct parts (Figure 5b) from one in which, at a lower Mr, they are included by the compositional whole *Decorative Green Area* and rendered uniformly (Figure 5c). Different tasks on the same dataset could indeed make better use of one or the other segmentation output. For an estimation of the urban vegetation volume to calculate $CO_2$ absorption, Figure 5b would be more appropriate, whereas results in Figure 5c could better support analyses for an urban planning centred on citizen quality of life. This example does not concern a taxonomical generalisation: a meadow is not a kind of decorative green area, on the other hand meadow and water jointly constitute the green area as a designed landscape unit. Such a relation cannot be expressed by a taxonomy alone and capturing it is precisely the contribution of the mereological axis.
The third example (Figure 6) shows a particular category standing out while its spatial neighbours are absorbed into a coarse abstraction. Figure 6b compares a maximal-resolution view, in which the roof and the solar panels upon it are individually visible, with a view obtained by lowering *Tr* and where the roof and the other components collapse into the *Architectural Element*

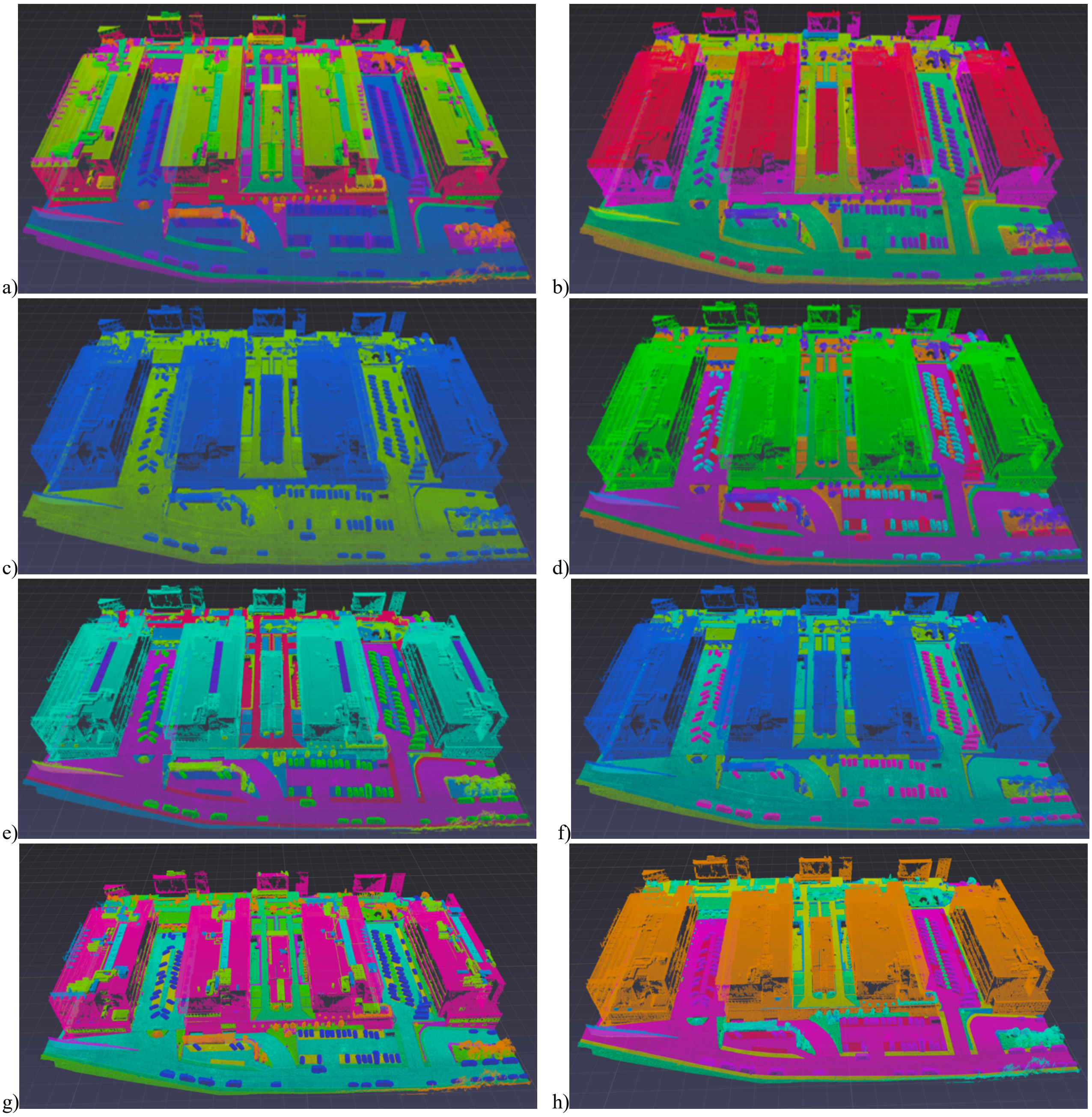



Figure 3: Examples of different DSS for the classified YTU3D dataset. Labels in Table 1 in the Appendix.

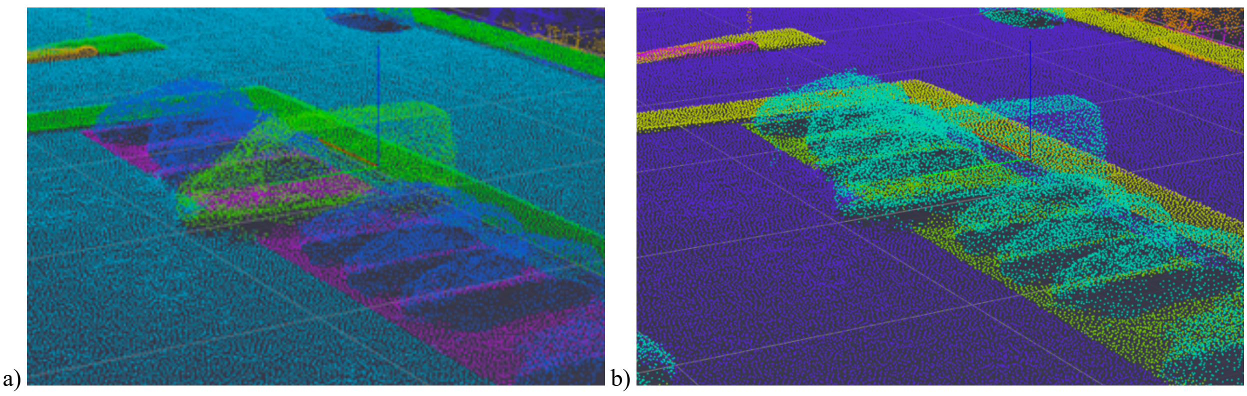
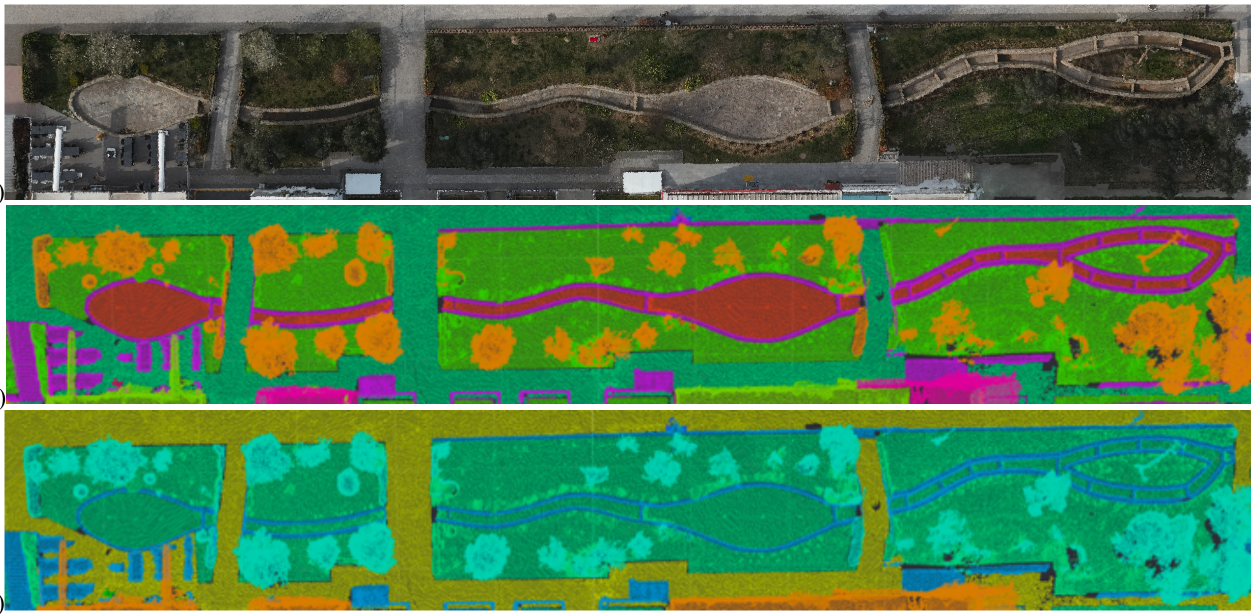
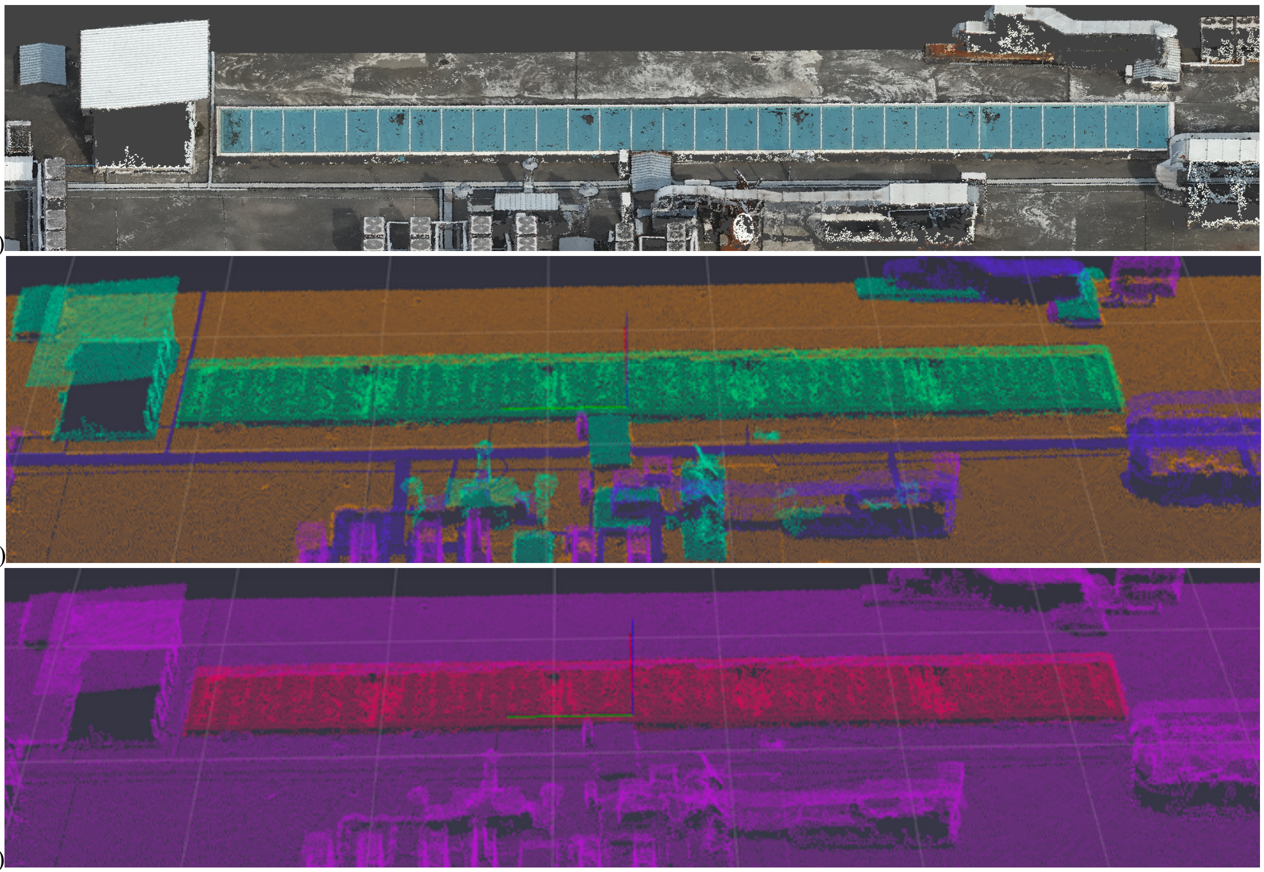



Figure 4: Conceptual abstraction along the taxonomical axis: *Car* (blue) and *Van* (green) distinguished as separate classes (a); *Car* and *Van are* both included under the generic *Vehicle* (light blue) abstraction at a lower taxonomical resolution.



a)
b)
c)

Figure 5: Selected view within the YTU3D dataset: RGB view (a); *Meadow* (green) and *Water Area* (red) shown as distinct parts (b). *Meadow* and *Water Area* both subsumed under the compositional whole *Decorative Green Area* (turquoise) under a lower *Mereological Resolution* (c).



a)
b)
c)

Figure 6: RGB view on the YTU dataset (a); roof surface and its elements separated at max *Tr* (b); solar panels (red) singled out as the only *Energy Producing Element* - whereas other instances belong to *Architectural* Element and are merged together - at a lower *Tr* (c).

abstraction while the solar panels remain singled out (Figure 6c). This distinction could support different tasks and analyses. Figure 6c could better serve for urban-scale analyses about energetic sustainability. In particular, the panels do not disappear into the building mass because, although they are mereologically part of the roof, they taxonomically belong to a different branch of the tree. This behaviour, anticipated with Figure 2e, is a direct consequence of the non-trivial decoupling between part-whole membership and conceptual subsumption.

### 6.3 Building-Scale and Mereology-Only Validation

Figure 7 shows visual results obtained on the 3DGraph of the Temple of Neptune (Italy). Crucially, this building-scale dataset is not associated with a high-order taxonomy, making *Mr* the only relevant parameter to vary. Nonetheless, results still highlight immediate differences between the visual outputs which can be obtained by traversing different *Mr* values. In particular, decreasing from *Mr* from 4 (Figure 7a) to 3 (Figure 7b) merges echini and abaci into capitals, then moving to *Mr*=2 (Figure 7c) unifies shafts and capitals into columns and aggregates friezes, cornices, architraves and tympani into ceilings and stone slabs and crepidomas into floorings. Finally, *Mr*=1 (Figure 7d) merges all architectural components into buildings.

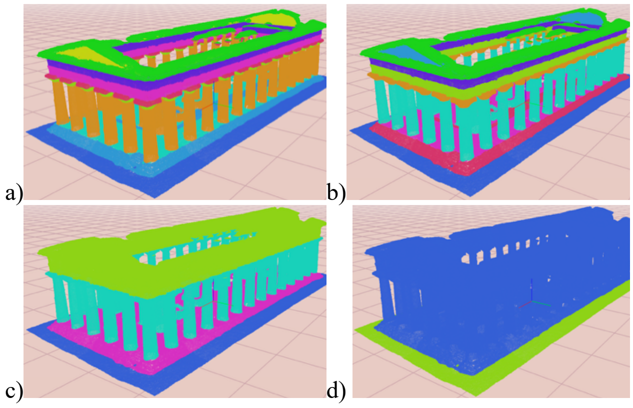



Figure 7: Visual results of the proposed DSS for the Temple of Neptune dataset with *Mr* progressively decreasing: *Mr*=4 (a); *Mr*=3 (b); *Mr*=2(c); *Mr*=1(d).

## 7. Conclusions

The paper introduces the Dynamic Semantic Segmentation (DSS) approach, which allows to extract more information out of the very same 3D point cloud classification effort, allowing domain experts to tailor semantic segmentation output to the scale and the specific needs of their concerned tasks. Eventually, the proposed approach has no drawbacks compared to its static counterpart, while leading, at the same time, to clear benefits in the amount of semantic information which can be extracted by the same classified dataset. More broadly, the contribution is less a new segmentation algorithm than a shift in how a classified point cloud can be treated by end-users: not as a finished artefact frozen at one granularity, but as a resource to be interrogated at whatever scale a question demands. The classification happens once; the range of semantics it can answer for stays open-ended and the cost of asking a new question is a query rather than a re-annotation. As future works, targeted segmentation possibilities will be explored, in which targeting a set of classes triggers a DSS with the right (*Mr*, *Tr*) pair to make them stand out.

## References

Armeni, I., Sener, O., Zamir, A.R., Jiang, H., Brilakis, I., Fischer, M., Savarese, S., 2016. 3D Semantic Parsing of Large-Scale Indoor Spaces. *Proc. CVPR*, 1534–1543.

Bayrak, O. C., Remondino, F., Uzar, M., 2023. A new dataset and methodology for urban-scale 3D point cloud classification. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XLVIII-1/W3-2023, 1-8.

Behley, J., Garbade, M., Milioto, A., Quenzel, J., Behnke, S., Stachniss, C., Gall, J., 2019. SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences. *Proc. ICCV*, 9297-9307.

Codiglione, M., Beber, R., Remondino, F., 2024a. Leveraging Ontology for Enhanced Queries and Analyses of Urban Point Clouds. *Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci.*, XLVIII-2/W8-2024, 101-108.

Codiglione, M., Mazzacca, G., Remondino, F., 2024b. Dimensional discoveries: Unveiling the potential of 3D heritage point clouds with a robust ontology framework. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci*, XLVIII-2/W4-2024, 119-125.

Grilli, E., Remondino, F., 2019. Classification of 3D Digital Heritage. *Remote Sensing*, Vol. 11(7), 847.

Gröger, G., Plümer, L., 2012. CityGML – Interoperable semantic 3D city models. *ISPRS J. Photogramm. Remote Sens.*, 71, 12-33.

Guo, Y., Wang, H., Hu, Q., Liu, H., Liu L., Bennamoun, M., 2021. Deep Learning for 3D Point Clouds: A Survey. *IEEE TPAMI*, Vol. 43, no. 12, pp. 4338-4364.

Hackel, T., Savinov, N., Ladicky, L., Wegner, J.D., Schindler, K., Pollefeys, M., 2017. SEMANTIC3D.NET: A New Large-Scale Point Cloud Classification Benchmark. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-1/W1, 91–98.

Hu, Q., Yang, B., Khalid, S., Xiao, W., Trigoni, N., Markham, A., 2021. Towards Semantic Segmentation of Urban-Scale 3D Point Clouds: A Dataset, Benchmarks and Challenges. *Proc. CVPR.*

Liu, H., Chen, H., Liu, S., 2026. CSEANet: a 3D point cloud semantic segmentation model based on channel spatial point attention and edge information awareness. *International Journal of Remote Sensing*, 47(14), 6331–6352.

Qi, C.R., Su, H., Mo, K., Guibas, L.J., 2017a. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. *Proc. CVPR*, 652–660.

Qi, C.R., Yi, L., Su, H., Guibas, L.J., 2017b. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. *Proc. NeurIPS*, 5099–5108.

Sarker, S., Sarker, P., Stone, G., Gorman, R., Tavakkoli, A., Bebis, G., Sattarvand, J., 2024. A comprehensive overview of deep learning techniques for 3D point cloud classification and semantic segmentation. *Machine Vision and Applic.,* 35, 67.

Song, Y., Li, Q., Wang, W., Yan, Z., 2026. PointGS: Semantic-consistent unsupervised 3D point cloud segmentation with 3D Gaussian splatting. Proc. *CVPR*.

Stojanovic, V., Trapp, M., Richter, R., Döllner, J., 2020. Ontology-Driven Analytics for Indoor Point Clouds. *Proc. 25th CAADRIA Conference*, Vol. 2, 537–546.

Winston, M.E., Chaffin, R., Herrmann, D., 1987. A Taxonomy of Part-Whole Relations. *Cognitive Science*, 11(4), 417-444.

Xie, Y., Tian, J., Zhu, X.X., 2020. Linking points with labels in 3D: A review of point cloud semantic segmentation. *IEEE Geosci. Remote Sens. Mag.*, 8, 38–59.

Zhang, J., Zhao, X., Chen, X., Lu, Z., 2019. A Review of Deep Learning-Based Semantic Segmentation for Point Cloud. in *IEEE Access,* 7, pp. 179118-179133.

## Appendix

Table 1 reports a clear explanation of the labels used in Figure 3.

| | Fig. 3a | Fig. 3b | Fig. 3c | Fig. 3d | Fig. 3e | Fig. 3f | Fig. 3g | Fig. 3h |
|---|---|---|---|---|---|---|---|---|
| STONE | | | X | | X | X | X | X |
| ROOF | | | X | X | X | X | X | X |
| ANCILLARY ELEMENT | | X | X | X | X | X | X | X |
| STREET LAMP | | | X | | X | X | X | X |
| WATER AREA | | | X | X | | X | | X |
| PARKING LOT | | | X | | | X | | |
| WINDOW | | X | X | X | X | X | X | X |
| FACADE | | | X | X | X | X | X | X |
| SOIL | | | X | | X | X | | |
| OTHER URBAN FURNITURE | | | X | | X | X | X | X |
| MEADOW | | | X | X | X | X | | X |
| SIDEWALK | | | X | | | X | | |
| POLE | | | X | | X | X | X | X |
| CAR | | | X | | X | X | X | X |
| VENTILATION | | X | X | X | X | X | X | X |
| FENCE | | | X | | X | X | X | X |
| TWO-WHEELED VEHICLE | | | X | | X | X | X | X |
| BUSH | | | X | | X | X | X | X |
| VAN | | | X | | X | X | X | X |
| PIPELINE | | X | X | X | X | X | X | X |
| GARBAGE BOX | | | X | | X | X | X | X |
| STREET | | | X | | | X | | |
| SOLAR PANEL | | X | X | X | X | X | | X |
| TREE | | | X | | X | X | X | X |
| STAIR | | | X | | X | X | X | X |
| DOG | | | X | | X | X | | |
| PERSON | | | X | | X | X | | |
| STREET SEPARATOR | | | X | | X | X | X | X |
| STREET WALL | | | X | | X | X | X | X |
| CHIMNEY | | X | X | X | X | X | X | X |
| TENT | | | X | X | X | X | X | X |
| BUILDING | | | X | | | X | X | |
| NATURAL ELEMENT | X | X | X | X | X | X | | X |
| URBAN ELEMENT | X | X | X | X | X | X | | |
| ANCILLARY ELEMENT | X | X | X | X | X | X | | X |
| FUNCTIONAL ELEMENT | X | X | X | X | X | X | | X |
| STRUCTURAL ELEMENT | X | X | X | X | X | X | | X |
| FOUR-WHEELED VEHICLE | X | X | X | X | X | X | | |
| TWO-WHEELED VEHICLE | X | X | X | X | X | X | | |
| ENVIRONMENTAL ELEMENT | X | X | X | X | | X | X | X |
| ARCHITECTURAL ELEMENT | X | X | X | X | | X | X | X |
| ENERGY-PRODUCING ELEMENT | X | X | X | X | | X | X | X |
| VEHICLE | X | X | X | X | | X | X | X |
| ANIMAL | X | X | X | X | | X | X | X |
| GROUND AREA | X | X | X | X | | X | X | X |
| IMMOVABLE OBJECT | X | X | X | X | X | | X | X |
| MOVABLE OBJECT | X | X | X | X | X | | X | X |
| NATURAL AREA | X | X | X | X | X | | X | X |
| BUILT AREA | X | X | X | X | X | | X | X |
| DECORATIVE GREEN AREA | X | X | X | | X | X | X | |
| OBJECT | X | X | | X | X | X | X | X |
| TOPOLOGICAL SURFACE | X | X | | X | X | X | X | X |

Table 1: Colour legends for Figure 3. In green the labels considered during classification, in orange the additional ones defined within the taxonomy.