INTEGRATION OF 3D POINT CLOUDS WITH SEMANTIC 3D CITY MODELS – PROVIDING SEMANTIC INFORMATION BEYOND CLASSIFICATION

C. Beil*, T. Kutzner, B. Schwab, B. Willenborg, A. Gawronski, T. H. Kolbe

Chair of Geoinformatics, Technical University of Munich, Germany -
(christof.beil, kutzner, benedikt.schwab, b.willenborg, alexander.gawronski, thomas.kolbe)@tum.de

KEY WORDS: 3D Point Clouds, 3D City Models, CityGML 3.0, LAS, Semantics, Digital Twin

ABSTRACT:

A range of different and increasingly accessible acquisition methods, the possibility for frequent data updates of large areas, and a simple data structure are some of the reasons for the popularity of three-dimensional (3D) point cloud data. While there are multiple techniques for segmenting and classifying point clouds, capabilities of common data formats such as LAS for providing semantic information are mostly limited to assigning points to a certain category (classification). However, several fields of application, such as digital urban twins used for simulations and analyses, require more detailed semantic knowledge. This can be provided by semantic 3D city models containing hierarchically structured semantic and spatial information. Although semantic models are often reconstructed from point clouds, they are usually geometrically less accurate due to generalization processes. First, point cloud data structures / formats are discussed with respect to their semantic capabilities. Then, a new approach for integrating point clouds with semantic 3D city models is presented, consequently combining respective advantages of both data types. In addition to elaborate (and established) semantic concepts for several thematic areas, the new version 3.0 of the international Open Geospatial Consortium (OGC) standard CityGML also provides a PointCloud module. In this paper a scheme is shown, how CityGML 3.0 can be used to provide semantic structures for point clouds (directly or stored in a separate LAS file). Methods and metrics to automatically assign points to corresponding Level of Detail (LoD)2 or LoD3 models are presented. Subsequently, dataset examples implementing these concepts are provided for download.

1. INTRODUCTION

Three-dimensional point cloud data are increasingly relevant in the context of several (emerging) applications such as digital urban and environmental twins, Building Information Modeling (BIM), autonomous driving, city modeling, and many others (Virtanen et al., 2017). One of the advantages of point clouds is the possibility to generate these data automatically for large areas and, thus, to enable frequent updates. Furthermore, due to its uniform structure based on point geometries, point clouds are easy to use by geospatial algorithms (Xu and Stilla, 2021). Point clouds are often the foundation from which semantic models in different formats are derived. Processes such as ”Scan-to-BIM” still vastly rely on manual modelling, especially when it comes to object structuring and aggregation. Most semantically rich data formats do not support the integrated representation of point cloud geometries. Thus, the link between original point cloud data and resulting semantic models is mostly lost. Conversely, semantic models are often generalised and could benefit from geometrically more detailed information provided by corresponding point cloud data. Also, there are tools and applications working with point cloud data directly, that avoid the sometimes costly and complex step of deriving semantic models from point cloud data (Peters et al., 2015). Semantic interpretations of 3D point clouds are often done but mostly limited to classification. Common point cloud formats (such as LAS) typically offer labels for assigning points within a point cloud to a certain category (such as Building or Vegetation). However, point cloud data with additional semantic information would be beneficial for several domains (Poux, 2019). In this paper, a new approach for bridging the gap between geometrically highly detailed point clouds and semantically rich 3D city models is presented. The following research methodology is applied. First, semantic capabilities of established point cloud formats such as LAS are investigated and relevant literature in the context of semantically enriching point cloud data is discussed. Then, based on these findings capabilities of the newly introduced PointCloud module of the international OGC standard CityGML version 3.0 are examined. Different methods for integrating point clouds with semantic city models using the PointCloud module are presented and discussed. It is shown that semantic structures provided by CityGML can be applied to point cloud data stored in typical formats such as LAS. Then, an exemplary algorithm, including methods and metrics for assigning point cloud (parts) to corresponding semantic models is presented using existing LoD3 models in combination with Mobile Laser Scanning (MLS) data. Furthermore, in order to demonstrate the practicability of the presented concepts, several examples for point cloud data coupled with LoD2 city models are shown and provided as Open Data.

2. DISCUSSION ON (SEMANTIC) POINT CLOUDS AND RELATED WORK

2.1 Three-dimensional point cloud data

Point clouds are generally defined as a set of 3D points, each represented using X-/Y-/Z-coordinates and optionally accompanied with additional information on color (e.g. RGB values), intensity, or other attributes. The acquisition of point cloud data has become faster, more affordable and, thus, more accessible in recent years. (Xu and Stilla, 2021) provide an extensive review on point cloud data generation methods. Point cloud data acquisition techniques can be divided into two main categor-

* Corresponding author

https://doi.org/10.5194/isprs-annals-VIII-4-W2-2021-105-2021 | © Author(s) 2021. CC BY 4.0 License.

ies - on the one hand ranging-based principles that use active methods for deriving 3D coordinates, such as ”Light Detection and Ranging” (LiDAR) or time-of-flight (ToF) systems; on the other hand imaging-based concepts that use triangulation techniques and stereo images to generate 3D points. (Xie et al., 2020) give an overview on advantages and disadvantages of various point cloud data sources and list corresponding applications. (Shahzad and Zhu, 2015) present an approach for reconstructing 3D building shapes from spaceborne synthetic aperture radar (SAR) point clouds. Potential future data sources may include laser scanners used by autonomous driving cars scanning their environment (Behley et al., 2013). Tackling the task of storing, managing and analysing the often huge amounts of point cloud data is subject to ongoing research (van Oosterom et al., 2015, Vo, 2017). (Meyer and Brunn, 2019) discuss possibilities for storing and processing massive point cloud data using spatial database systems. Among other advantages of databases, this allows relating point cloud data to other types of spatial (or non-spatial) data within the database. Raw point cloud data already provides highly detailed geometric representations of real-world objects but does not include additional semantic information. Thus, point cloud data are often segmented, classified, and further processed.

2.2 Point cloud segmentation, classification, and processing

While the aforementioned methods allow the acquisition of large and geometrically accurate point clouds, this data needs to be further processed in order to be usable for many applications. In a first step, point clouds can be grouped into parts with similar properties (segmentation), in a second step these regions can then be labeled according to a specific class (classification). Some methods directly conduct a semantic segmentation generating labelled points from raw data. In computer vision and deep-learning applications, the task of assigning labels to points within a point cloud is also referred to as ”Point Cloud Semantic Segmentation” (PCSS) (Xie et al., 2020). (Grilli et al., 2017) present a detailed review on different techniques for segmenting and classifying point cloud data. Recently, 3D neuronal networks have been used for improving segmentation and classification methods. An example of a classified point cloud with each point assigned to a specific category is depicted in Figure 1. These classified point clouds can then be the foundation for creating semantic models in formats such as OpenDRIVE, Industry Foundation Classes (IFC), or CityGML. Detecting and generating shapes and objects from point clouds has been done for quite some time (Vosselman and Dijkman, 2001, Schnabel et al., 2007). (Xu and Stilla, 2021) present numerous methods for object reconstruction from point clouds in the context of building and civil infrastructure. Digital Terrain Models (DTM), Digital Surface Models (DSM) or Mesh models are also commonly generated from point cloud data.

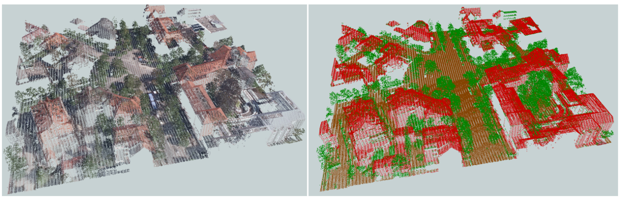


Figure 1. Colored point cloud (left) and classified point cloud (right); brown=ground, red=building, green=vegetation

2.3 Point cloud data formats / LAS

While there are many data formats available for working with point cloud data, the LAS format is regarded as the industry standard for the storage and interchange of LIDAR (or other) point cloud data. Thus, the current *LAS Specification 1.4*, specified by the *American Society for Photogrammetry and Remote Sensing* (ASPRS, 2018) is examined and discussed. LAS files contain binary data with a so-called Public Header Block at the beginning of each document, followed by any number of (optional) Variable Length Records (VLRs, e.g. projection information, metadata, etc.), Point Data Records, and any number of (optional) Extended Variable Length Records (EVLRs, allow a higher payload than VLRs and can be appended to the end of an LAS file). There are currently 11 different Point Data Record Formats, differing in available data fields (sometimes referred to as ”components”). Point Data Records contain data fields with information for each individual point within a point cloud including X-/Y-/Z-coordinates, intensity, classification, and other attributes. The specification provides a code list for several classification types such as (but not limited to) Unclassified, Ground, Building, Water, Vegetation, Rail, or Road Surface, while also allowing user definable codes. Another component called ”Point Source ID” is used for indicating the file from which a point originated. A compressed variant of LAS called LAZ is also commonly used. LAS does not provide any concepts for semantic object structures such as hierarchies or aggregation. One of the reasons for the popularity of the LAS format is its simplicity and easy-to-use structure. While in principle extending LAS point data records with additional user-defined attributes (using extra bytes) is possible, there are no guidelines for making use of this concept in the context of extended semantic capabilities. Furthermore, software tools likely will not be able to interpret these additional attributes. Since attributes can only be assigned to points, the LAS format is not capable of storing information on objects that are not represented with at least one point.

2.4 Related work on extending the semantic capabilities of point clouds

The term ”semantic point cloud” usually merely refers to points assigned to certain categories (see section 2.2), whereas ”semantic 3D city model” has a more comprehensive meaning. This includes hierarchically structured and topologically connected representations of objects in combination with a number of available attributes. Integrating point cloud data with structured knowledge and semantics (beyond classification) can be beneficial for several applications, domains, and decision making (Poux, 2019). However, there are few concepts for extending semantic capabilities of point clouds. (Rusu et al., 2008) discuss a point-cloud-based object map for robotic assistant systems. (Ben Hmida et al., 2012) propose a knowledge-based detection of objects in point clouds by extracting this information from databases, CAD plans, GIS, and other sources. Detected objects are then annotated with semantic information. (Aljumaily et al., 2019) present a method for enriching point clouds with semantic information based on GIS data aiming to improve point cloud classification results. User-defined LAS labels are used in order to introduce additional classification tags. (Poux, 2019) describes a data model for a ”Smart Point Cloud (SPC) infrastructure” providing a conceptual framework for the semantic enrichment and structuring of point clouds. This model is split into three conceptual levels. Level-0 describes a generalized SPC meta model. Here, ”SemanticPatches” are defined as small subsets of points with a shared relationship based on available knowledge (e.g. point belonging to a plane). Each semantic patch can retain attributes including classification status and confidence of classification. Level-1 contains a connection-layer meta model divided into two sublevels. Level L1-1 contains "ConnectedElements" aggregated from closely related "SemanticPatches", which again can be aggregated to "AggregatedElements" (e.g. a table could be described using 5 "ConnectedElements" (4 feet and 1 tabletop), or 1 "AggregatedElement" for the entire object). Additionally, "ConnectedElements" can relate to "Spaces" defined as a set of dimensions. In a lower abstraction level L1-2, "Spaces" and "ConnectedElements" are connected to "Sub-spaces" and "WorldObjects", that constitute the entry points on which different domain ontologies / specializations can be integrated in level-2 (Poux et al., 2017), (Poux et al., 2018). (El-Mahgary et al., 2020) present a concept for splitting semantic and geometric information of point cloud data using two separate files (called "Points" and "Semantics"). Both files have exactly the same ordering of points and corresponding semantic information and, thus, do not need indexing. While the file "Points" contains four columns (X, Y, Z, and Intensity) the file "Semantics" only provides one semantic label or class ID per corresponding point. This does not allow the representation of hierarchical structures (such as a door being part of a facade part of a building) but enables each point to be referenced to a specific object (e.g. via an integer representing the building ID of a specific building), rather than a classic label only indicating that points are of a certain category (e.g. buildings in general). Methods presented so far always use point cloud data as the foundation for extended semantic concepts. Thus, objects where no point data are available (e.g. obscured backyard facades) also cannot be represented. In contrast, standards such as CityGML allow the storage of objects independently from geometric correspondences (e.g. rooms within a building can be represented, even when no geometric representation is available). (Krijnen and Beetz, 2017) suggest an extension to the Industry Foundation Classes (IFC) (ISO16739-1, 2018) to integrate point cloud datasets for harmonizing these two data types. They point out that a point cloud structured according to the semantics of the IFC schema allows the search, selection, and use of specific point cloud subset clusters. Furthermore, potentials of segmenting point cloud data according to explicitly modelled semantic objects for efficient visualization and analysis purposes are presented. However, drawbacks of combining these two different data types within one common storage format are also discussed. These include a more complicated update process of the data, missing software for working with unified IFC / point cloud data and the increasing file size compared to original IFC files. (Virtanen et al., 2017) describe the possibility of including object-specific semantic information in a point cloud by adding a national building ID to all points belonging to a corresponding building.

2.5 Potential of coupling point clouds with semantic 3D city models

Digital 3D city models are commonly used for a variety of applications (Biljecki et al., 2015). One of the main advantage of these models is the consistent integration of geometric, semantic and topological information (Kolbe and Donaubauer, 2021). The international OGC standard CityGML is widely used for modelling semantic 3D city models. Real-world objects are modelled in a hierarchical structure and segmented by thematic modules. These include models of buildings, terrain, vegetation, or the streetspace (Beil et al., 2020). Semantic capabilities of 3D city models in combination with geometrically highly detailed point cloud data have great potential for several applications. Point cloud data have been a source for generating digital urban environment models from the beginning (Haala and Brenner, 1997), (Kolbe et al., 2009). (Prieto et al., 2012) describe the complex task of generating CityGML (building) models from point cloud data. LoD2 building models are often derived from LiDAR generated point clouds (Ortega et al., 2021). (Murtiyoso et al., 2020) present a workflow for automatic roof extraction from point clouds for the generation of CityGML models. Workflows for modelling CityGML tree objects from point clouds are presented in (Gobeawan et al., 2018). This close relationship between these two data types is mostly lost. However, concepts for integrating point clouds with semantic 3D city models would come with several advantages. Information on semantic affiliation of point cloud subsets to individual objects can be beneficial for quickly identifying parts of a point cloud relevant for a specific task or application. A thematic search of point cloud data can be performed using semantic information. As mentioned before, large point cloud datasets require spatial indexing for efficient visualization and analysis. Semantic information could be used for improving these decomposition strategies by segmenting point cloud data according to semantic structures (Krijnen and Beetz, 2017). There are tools for visualizing semantic 3D city models using an extension to the CesiumJS WebGL virtual globe (Yao et al., 2018). Similarly, point cloud data can be transformed to Cesium 3D tiles to enable a combined visualization of both data types[1]. This allows for a direct comparison of large semantic city models with corresponding point cloud data in an efficient way. Conversely, semantic city models such as buildings are generalised representations of real-world objects. While this is sufficient in many cases, emerging applications such as virtually testing autonomous driving vehicles require more detailed geometrical information of the environment (Schwab et al., 2020). While building models in LoD2 have become a standard for most city models and can be generated (semi-) automatically, creating even more detailed structures or indoor models is time and cost intensive. Thus, it might be useful to represent these models using point cloud geometries directly. Several cities including Helsinki or Munich have created visually appealing mesh models as well as semantically rich 3D city models from (imaging-based) point cloud data. In the future, a link between these two datasets could be directly realised during the generation process of these models. (Poux, 2019) points out that while point clouds often serve as a foundation for creating models in GIS, CAD or BIM, ways for a direct integration of point clouds with these data types are very limited. Thus, in the following section possibilities for bridging the gap between 3D point clouds and semantic 3D city models are presented and discussed using capabilities of the new version 3.0 of the international CityGML standard.

3. INTEGRATION OF POINT CLOUDS WITH SEMANTIC 3D CITY MODELS

3.1 Conceptual approach using CityGML 3.0

Version 3.0 of the international OGC standard CityGML will come up with several new and revised concepts to improve the use of CityGML in many areas of application including Smart Cities, urban planning, and traffic analyses (Kutzner et al., 2020).

[1] `https://github.com/tum-gis/cesium-point-cloud-generator`

CityGML has a modular structure allowing applications to employ only those concepts that are relevant to them. The so-called *Core* module defines the base concepts that need to be implemented by all applications, such as the LoD concept with the geometries. The objects of the built environment are defined in thematic modules like Building, Transportation, or Vegetation. In addition, several modules define concepts that are applicable to all thematic modules, such as Appearance, or Generics. The newly introduced *PointCloud* module in CityGML 3.0 also falls into this category.

CityGML 3.0 introduces a new *Space concept* in the Core module that is of fundamental importance to the semantic structuring of 3D city models. All city objects are semantically distinguished into *spaces* and *space boundaries*. Spaces represent objects with volumetric extent in the real world (e.g. buildings, water bodies, or transportation spaces), whereas space boundaries represent objects with areal extent in the real world that delimit and connect spaces (e.g. wall surfaces, water surfaces, or road surfaces). Spaces are, furthermore, distinguished into *physical spaces* and *logical spaces* depending on whether they are bounded by physical or virtual boundaries. Buildings and rooms are considered physical spaces, as they are bounded by wall and roof surfaces, whereas the division of a building into publicly accessible areas and areas with restricted access is of purely virtual nature. Physical spaces are further classified into *occupied spaces* and *unoccupied spaces*, depending on whether they occupy space in the urban environment or not. Buildings and trees, for instance, represent occupied spaces, as the space that is blocked by these objects is no longer available for placing other objects in that space. Rooms and transportation spaces, in contrast, are considered unoccupied spaces, as their space can still be filled with other objects.

In the CityGML UML model, the spaces and space boundaries are represented by specific abstract classes in an hierarchical order (cf. (Kutzner et al., 2020)). The concrete city objects in the thematic modules are defined as subclasses of these space and space boundary classes. Figure 2 illustrates this for the city objects *Building*, *WallSurface*, and *RoofSurface*. Building is defined as a subclass of *AbstractOccupiedSpace*, and transitively as subclass of *AbstractPhysicalSpace* and *AbstractSpace*; thus, buildings represent occupied physical spaces. WallSurface and RoofSurface are defined as subclasses of *AbstractThematicSurface* and transitively as subclass of *AbstractSpaceBoundary*; thus, these surfaces represent thematic space boundaries. The association between AbstractSpace and AbstractSpaceBoundary denotes that these surfaces bound the building.

In CityGML 3.0, the LoD concept and the geometries are not specified individually in each thematic module anymore, but they are defined centrally in the Core module and are associated there with the space and space boundary classes. In this way, the geometries are inherited by all concrete classes in the thematic modules and redundancy can be avoided. In addition, the new PointCloud module allows for representing the geometry of city objects by 3D point clouds. Figure 3 shows the UML diagram of the PointCloud module. All city objects that represent physical spaces or thematic surfaces can now be represented as point clouds, this also applies to the objects Building, WallSurface, and RoofSurface from Figure 2.

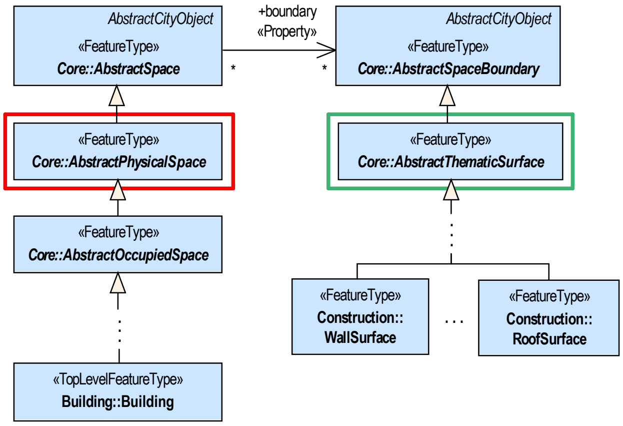


Figure 2. Classification of Building elements based on the CityGML 3.0 Space concept. Classes not relevant for illustrating the classification hierarchy are omitted. The prefixes before :: denote the modules in which the concepts are defined.

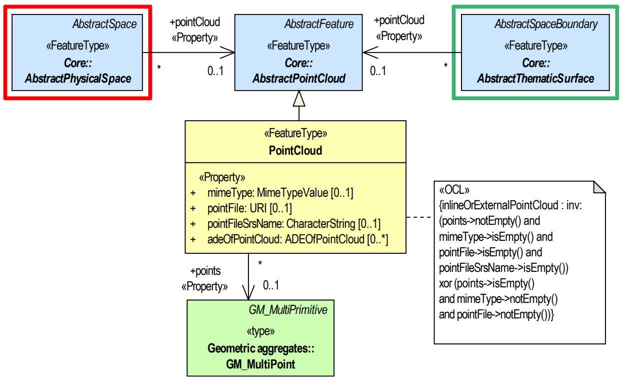


Figure 3. CityGML 3.0 PointCloud module (CityGML SWG, 2021)

The conceptual design of the PointCloud module allows for coupling 3D city models with point clouds in different ways:

1. The point clouds are represented inline with the city objects using MultiPoint geometries. In this way, each city object (e.g. buildings and even the individual surfaces that bound the building) can be complemented with its specific point cloud representation directly in the CityGML file. Due to the large file size that results from storing the points directly in the CityGML file, this approach is only recommended for data sets containing a small number of city objects, for example, for providing self-contained data sets for archiving, or for homogeneous structuring in databases.

2. For each city object, a separate point cloud file is provided (e.g. in LAS or LAZ format) and each city object in the CityGML file references the corresponding point cloud file. The disadvantage of this approach, however, is that this can result in a huge amount of individual point cloud files, one for each city object represented in the CityGML file.

3. One point cloud file is provided that contains all points from a specific area. Each point contains information to which city object the point belongs. In LAS files, this can be implemented by using the component *Point Source ID* and setting it to the same value for all points belonging to a specific city object. Each city object in the CityGML file references the point cloud file and all points with the corresponding value in the *Point Source ID* component. Figure 8 illustrates this approach. Since the *Point Source ID* can only store 16 bits, only 65,536 different values are allowed and, thus, a maximum of 65,536 city objects can reference the point cloud file. This could be improved

by including the component *Classification* that allows for specifying up to 256 different classes. By combining the two components, $256 \times 65,536$ different city objects can reference the point cloud file. As mentioned in 2.3, the actual semantics of the *Point Source ID* component differs slightly according to the LAS standard, however, it is considered a reasonable option for reuse. Furthermore, it would generally also be possible to store the *gml:id* attribute of the corresponding CityGML object in an extended data field of every point. However, many point cloud tools are not yet able to deal with extended point cloud formats.

Although each approach has its drawbacks, the huge advantage of all three approaches is that the rich semantics of CityGML 3.0 can be coupled with the simple structure of point clouds. It is possible to derive semantic information from point clouds as described in section 2, however, not with the semantic richness and structure that is required by many applications and that is provided by CityGML. By coupling CityGML with point clouds, there is no need to extend point cloud formats to allow for representing more semantic information, the existing semantic concepts from CityGML 3.0 can directly be used to the full extent. The concepts presented in this section make use of point cloud data in the LAS format; however, it is possible to apply this method to all point cloud data formats that allow the storage of additional attributes for each point. As described in (Meyer and Brunn, 2019), point cloud data can be managed by spatial database systems. Thus, using the concepts presented here, CityGML objects and corresponding point cloud data can also be stored and linked within a common database.
A remaining challenge is the correct allocation of point cloud data (parts) with corresponding semantic 3D city models. In the following section potential metrics for achieving this task are presented.

3.2 Metrics for associating and integrating 3D point clouds with semantic models

The methods for associating and integrating point cloud subsets with objects from a semantic city model depend on the available input datasets (Wysocki et al., 2021). Thereby, one major use case is to link an existing semantic city model with a point cloud obtained from a surveying campaign or more generally a sensor observation. Figure 4 depicts a proposed approach for this, where the first step is to cut out a portion of the point cloud that matches the geometry of a city object and presumably represents it. However, if the model was created based on the point cloud, modeling inaccuracies or missing objects can lead to association discrepancies. If the point cloud was surveyed at a later time than the model, changes in the physical environment will also lead to discrepancies. Thus, metrics are needed to evaluate whether a point cloud subset adequately represents an object from the city model and is therefore suitable for integration.

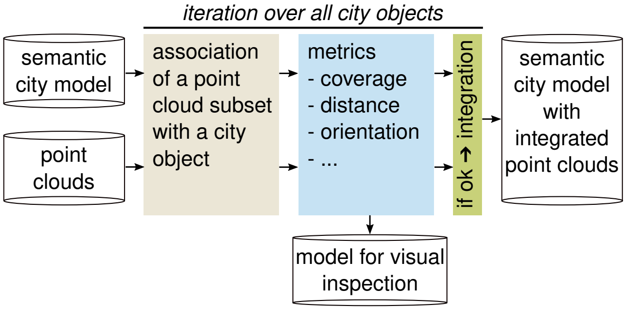
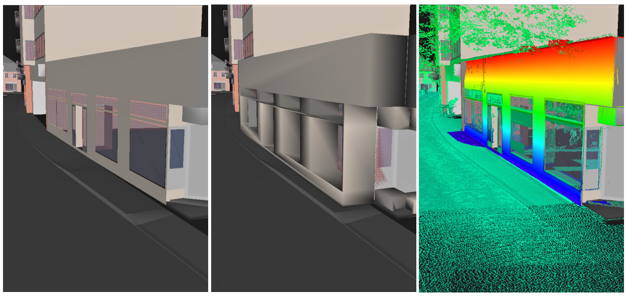

iteration over all city objects
semantic city model
point clouds
association of a point cloud subset with a city object
metrics
- coverage
- distance
- orientation
- ...
if ok → integration
semantic city model with integrated point clouds
model for visual inspection

Figure 4. Method overview for integrating point clouds with semantic models utilizing quality metrics.



(i) Semantic model (ii) 3D Buffering (iii) Subset (colorful)

Figure 5. Identification of the point cloud subset (colorful) that presumably represents the wall surface of the semantic model.

Our test dataset contains MLS point clouds (relative accuracy 1-3cm; density up to 3000 pts/m^2) as well as 50 corresponding LoD3 building models[2], as shown in Figure 5i. To clip the point cloud for each city object, a 3D buffer for each geometry surface is created in the direction of the surface normal, as depicted in Figure 5ii. The obtained 3D solids serve as point cloud filters for the respective object, for instance a wall, door, or road.
Based on the associated point cloud subset and the city object, a set of metrics are evaluated, which are listed in Table 1. In order to determine whether the surveying campaign has sufficiently covered the object, the metric *point density* $[pts/m^2]$ on the object surface is computed. In order to automatically refine geometries, (Wysocki, 2020) introduces a metric with a similar objective, but related to the 2D footprint. If a minimum coverage is achieved, further reliable statements can be drawn about the relationship between the point cloud and the model surface. First, the shortest distances between each point and the model surface are calculated. Figure 6 shows a histogram of the calculated distances for a particular wall object. Significant displacements can be identified by means of measures from descriptive statistics. For reliable conclusions with respect to outliers, it is evaluated whether the median is located within a certain range of values. As the relative accuracy of the given point cloud dataset ranges from 1-3 cm, a median of more than 5 cm is interpreted as a significant deviation. As the street surface consists of cobblestones, this threshold is relaxed to 11.3 cm for road features.
A further indication of geometric deviations may be obtained by comparing the object's surface with an estimated plane. For

Metric	Indication for
Point density	object coverage by observation data
Point distance distribution	displacement in the direction of the surface normal
Estimated plane orientation	orientation deviation to surface geometry of object
Intensity distribution	material class/characteristic

Table 1. List of implemented metrics to assess the matching between point cloud and city object.

[2] https://github.com/savenow/lod3-road-space-models

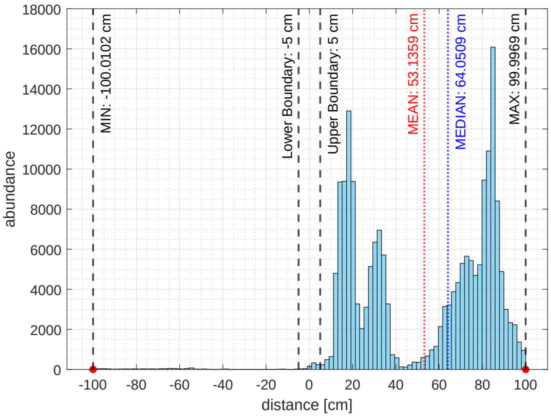


Figure 6. Distribution of the distances between the points and the surface of the large wall with the three balconies in Figure 7. This wall is occluded by a scaffolding with a typical width of around 75 cm, accounting for the bimodal distribution.

this purpose, the outliers of the point cloud subset are filtered using the RANdom SAmple Consensus (RANSAC) algorithm. This is implemented using the software Feature Manipulation Engine (FME) and Python scripts. To determine the normal vector of the best fitting plane, the eigenvectors are calculated using the Principal Component Analysis (PCA). The orientation difference between object surface and estimated plane is interpreted as significant if greater than ±1 degree, whereas the threshold was exploratively selected based on the dataset. Apart from the purely geometrical deviations, the reflection intensity can be used as an indicator for the class of the material. E.g. road markings are characterized by a higher reflective portion compared to pure asphalt.
Figure 7 shows the semantic model colored according to the *point distance distribution* metric. Here, the large wall in the left half of the image with its three balconies relates to the histogram in Figure 6 and the lower image of Figure 7 depicts the point cloud for comparison. The results show that, for example, the scaffolding was correctly classified as a geometric deviation and no deviation was detected for the free wall on the left. The identified deviations in the street elements are either caused by parked cars or by insufficiently modelled crossfalls.

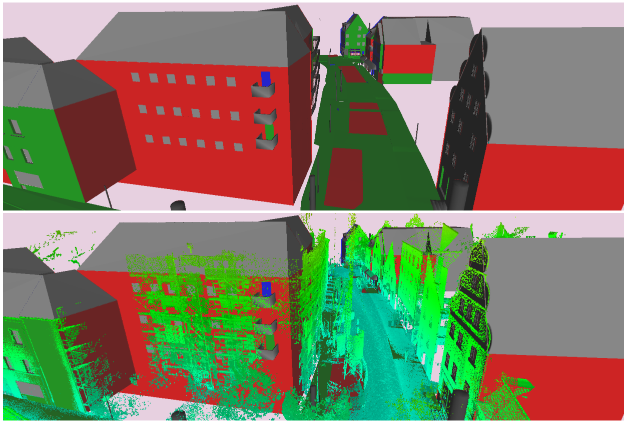


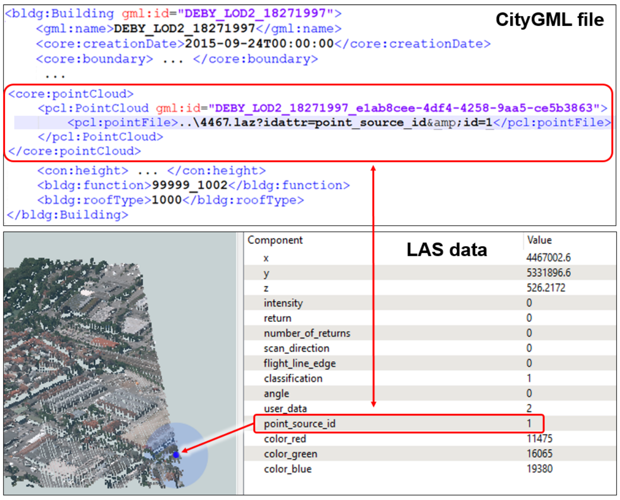
Figure 7. Colored building and street models for the inspection of the point distance distribution - green object surfaces indicate that the median is located within the tolerance range. Blue indicates a significant displacement towards the inner of the building and red indicates to the exterior.

The rear walls of the building models are entirely gray since the point density of the MLS is too low. In order to quantify the completeness of the model, the ratio of non-associated points to the total number of points could be used as a further metric. However, the implemented metrics still need to be evaluated on larger datasets and the robustness of the method to inaccurate georeferencing needs to be further investigated in future work.

4. EXAMPLES

4.1 Semantic 3D city models coupled with point cloud data

Using FME, several examples are generated to illustrate concepts described in subsection 3.1. Figure 8 shows a CityGML file encoded in XML and a corresponding point cloud visualized in the FME Inspector. The point cloud data used in this example covers an area of 1 $[km^2]$ with a point density of 6,25 $[pts/m^2]$ and was derived by the Bavarian Agency for Digitisation, High-Speed Internet and Surveying using a dense image matching method. Additionally, corresponding LoD2 CityGML building models are available. While the concept is demonstrated using one 1x1km tile, this data is available for the whole of Bavaria. For this example, the recommended method of linking CityGML objects to one external point cloud file is used. A CityGML building object providing attributes such as function or roof type also contains information on corresponding point cloud data. On the one hand, each point within an LAS file contains a Point Source ID linking it to a city object. Objects in the CityGML file, on the other hand, provide a data path to the LAS file and the Point Source ID value of all points representing this object. This example was generated by counting all building objects within a CityGML file and assigning this counted value to points in a corresponding point cloud. Figure 8 illustrates this concept for a building object coupled to all points within an LAS file with a Point Source ID value equal to 1.



```
<bldg:Building gml:id="DEBY_LOD2_18271997">
    <gml:name>DEBY_LOD2_18271997</gml:name>
    <core:creationDate>2015-09-24T00:00:00</core:creationDate>
    <core:boundary> ... </core:boundary>
     ...
<core:pointCloud>
    <pcl:PointCloud gml:id="DEBY_LOD2_18271997_e1ab8cee-4df4-4258-9aa5-ce5b3863">
        <pcl:pointFile>..\4467.laz?idattr=point_source_id&amp;id=1</pcl:pointFile>
    </pcl:PointCloud>
</core:pointCloud>
    <con:height> ... </con:height>
    <bldg:function>99999_1002</bldg:function>
    <bldg:roofType>1000</bldg:roofType>
</bldg:Building>
```

Component	Value
x	4467002.6
y	5331896.6
z	526.2172
intensity	0
return	0
number_of_returns	0
scan_direction	0
flight_line_edge	0
classification	1
angle	0
user_data	2
point_source_id	1
color_red	11475
color_green	16065
color_blue	19380

Figure 8. Example for linking points within a LAS file to CityGML objects via the ”Point Source ID” point data record

Sample data making use of the concepts described in subsection 3.1 are available for download [3]. This includes example datasets for representing point cloud geometries inline as well as data using the coupling method of CityGML models and an external LAS file shown in Figure 8.

[3] https://github.com/opengeospatial/CityGML-3.0Encodings/tree/master/CityGML/Examples/PointCloud

4.2 Point cloud data derived from semantic 3D city models

While the previous examples coupled point clouds with corresponding semantic city models from different datasets, it is also possible to generate point cloud data from city models directly. While point clouds created in this way do not represent the real-world geometries as good as measured point clouds (e.g. from laser scanning), they are directly linked to the city model objects they have been derived from with corresponding attributes and semantic structure. This strategy of creating an artificial *sampling point cloud* enables the immediate use of point cloud tools and algorithms on the semantic 3D city model and the potentials discussed in subsection 2.5.
For instance, (Willenborg et al., 2018, Chaturvedi et al., 2017) present a method for a solar potential simulation based on CityGML models. The application estimates the direct, diffuse, and global solar irradiation on city model features. The visibility analysis of the simulation is conducted using a ray-casting approach, that requires a set of sampling points on the city model surfaces. The simulation results are attached to the sampling points via attributes and stored in a PostGIS database with the identifiers of the corresponding city model objects. This allows to compute aggregate values of the simulation results per city model feature (e.g. WallSurface, RoofSurface, Building) and the generation of textures. However, it is also possible to export the point grid including simulation results directly. This data can easily be transformed to LAS or other point cloud formats. Links to original building or surface models can be preserved using the method described in subsection 3.1. Figure 9 shows a point cloud generated from building models in the described process of a solar irradiation simulation. Each point is colored according to simulation results with a colour gradient ranging from blue (no irradiation) to red (maximum irradiation) in $[kWh/m^2year]$ and contains attributes from the original models. Similar processes would be possible for applications such as visibility analysis. This is also not limited to models of buildings but could be extended to all thematic objects such as vegetation or streetspace models.

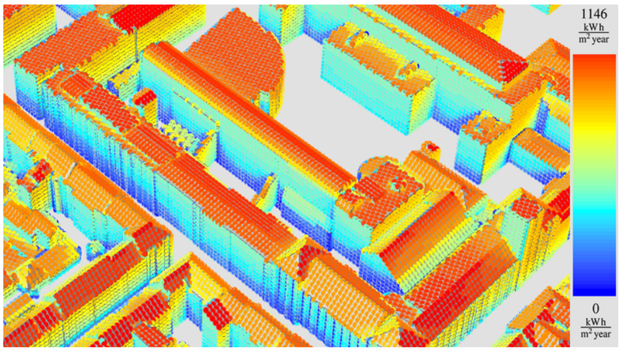


Figure 9. Point cloud generated from CityGML building models in the process of a solar potential simulation (Willenborg et al., 2018)

5. DISCUSSION AND OUTLOOK

In this paper, concepts for integrating point clouds with structured semantic information (far beyond classification) provided by 3D city models are presented using concepts of the OGC standard CityGML version 3.0. This can be achieved using different methods. While it is possible to integrate points of a point cloud inline within a CityGML file, this may only be feasible for small datasets in order to provide self-containing data useful for applications such as archiving. Since it is ineffective to store point cloud geometries within GML directly, the presented concept to link CityGML models with corresponding point clouds provided as separate LAS files (or in similar formats) is recommended. Obviously, the coupled point cloud and city model datasets should represent a scenario at the same point in time (same acquisition date) and provide information in a comparable resolution and accuracy. Since city models are often directly derived from point cloud data, these problems could be avoided by already establishing relations between these two datasets during the generation process of these models. Conversely, it was shown how point clouds can be generated from city models, thus also enabling a close connection. However, if one of the two datasets is updated (independently from the other), links between these datasets need to be updated too. Depending on the intended application, representations in multiple LoDs are needed. CityGML enables an object to be uniquely identified, while allowing representations in different thematic as well as geometric resolutions (LoDs), thus serving as an anchor point linking these different types of representation. Until now availability of LoD3 models is limited. In the context of creating digital urban twins, cities such as Munich currently gather more detailed data for large areas. Concepts and methods presented in this paper could be applied to this data. Several examples focusing on city scale objects such as buildings are presented. However, CityGML also contains a Construction module that includes concepts for representing smaller objects such as furniture, installations, or other constructive elements. Thus, applying the concepts for coupling point clouds with semantic models of tables, chairs, etc. is possible. Since both types of data will play a significant role in the context of generating (and updating) a digital twin of the urban environment, relations between point clouds and corresponding semantic city models will get increasingly relevant. While point clouds, mesh models, or semantic city models are sometimes described as a ”digital twin” in their own right, a true digital twin of the urban environment should be a system of systems integrating different and heterogeneous information. Concepts presented in this paper can be a step towards that direction by bridging the gap between point clouds and semantic 3D city models.

ACKNOWLEDGEMENTS

We would like to thank the Bavarian Agency for Digitisation, High-Speed Internet and Surveying for providing the data used to generate most sample datasets. We also want to thank Safe Software for providing the academic license enabling us to use the software FME.

REFERENCES

Aljumaily, H., Laefer, D. F., Cuadra, D., 2019. Integration of Lidar Data and GIS Data for Point Cloud Semantic Enrichment at the Point Level. *Photogrammetric Engineering & Remote Sensing*, 85(1), 29–42.

ASPRS, 2018. LAS specification version 1.4 - 17-030rl. American Society for Photogrammetry & Remote Sensing.

Behley, J., Steinhage, V., Cremers, A. B., 2013. Laser-based segment classification using a mixture of bag-of-words. *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 4195–4200.

Beil, C., Ruhdorfer, R., Coduro, T., Kolbe, T. H., 2020. Detailed Streetspace Modelling for Multiple Applications: Discussions on the Proposed

CityGML 3.0 Transportation Model. *ISPRS International Journal of Geo-Information*, 9(10), 603.

Ben Hmida, H., Cruz, C., Boochs, F., Nicolle, C., 2012. From Unstructured 3D Point Clouds to Structured Knowledge - A Semantics Approach. *Semantics-Advances In Theories and Mathematical Models*, 213.

Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., Çöltekin, A., 2015. Applications of 3D city models: State of the art review. *ISPRS International Journal of Geo-Information*, 4(4), 2842–2889.

Chaturvedi, K., Willenborg, B., Sindram, M., Kolbe, T. H., 2017. Solar Potential Analysis and Integration of the Time-dependent Simulation Results for Semantic 3D City Models Using Dynamizers. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, IV-4/W5, 25-32.

CityGML SWG, 2021. CityGML 3.0 GitHub repository. `https://github.com/opengeospatial/CityGML-3.0CM`. [accessed 2021-03-29].

El-Mahgary, S., Virtanen, J.-P., Hyyppä, H., 2020. A Simple Semantic-Based Data Storage Layout for Querying Point Clouds. *ISPRS International Journal of Geo-Information*, 9(2), 72.

Gobeawan, L., Lin, E., Tandon, A., Yee, A., Khoo, V., Teo, S., Yi, S., Lim, C., Wong, S., Wise, D. et al., 2018. Modeling Tress for virtual Singapore: From Data Acquisition to CityGML Models. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 42.

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.

Haala, N., Brenner, C., 1997. Generation of 3D city models from airborne laser scanning data. *Proceedings EARSEL Workshop on LIDAR remote sensing on land and sea, Tallinn/Estonia*, 105–112.

ISO16739-1, 2018. Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries. Technical report, International Organization for Standardization, Geneva, Switzerland.

Kolbe, T. H., Donaubauer, A., 2021. Semantic 3D City Modeling and BIM. *Urban Informatics*, 609–636.

Kolbe, T. H., Nagel, C., Stadler, A., 2009. CityGML–OGC Standard for Photogrammetry? *Photogrammetric week*, 9, 265–277.

Krijnen, T., Beetz, J., 2017. An IFC schema extension and binary serialization format to efficiently integrate point cloud data into building models. *Advanced Engineering Informatics*, 33, 473–490.

Kutzner, T., Chaturvedi, K., Kolbe, T. H., 2020. CityGML 3.0: New functions open up new applications. *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science*, 88(1), 43–61.

Meyer, T., Brunn, A., 2019. 3D Point Clouds in PostgreSQL/PostGIS for Applications in GIS and Geodesy. *GISTAM*, 154–163.

Murtiyoso, A., Veriandi, M., Suwardhi, D., Soeksmantono, B., Harto, A. B., 2020. Automatic Workflow for Roof Extraction and Generation of 3D CityGML Models from Low-Cost UAV Image-Derived Point Clouds. *ISPRS International Journal of Geo-Information*, 9(12), 743.

Ortega, S., Santana, J. M., Wendel, J., Trujillo, A., Murshed, S. M., 2021. Generating 3D City Models from Open LiDAR Point Clouds: Advancing Towards Smart City Applications. *Open Source Geospatial Science for Urban Studies*, Springer, 97–116.

Peters, R., Ledoux, H., Biljecki, F., 2015. Visibility analysis in a point cloud based on the medial axis transform. *UDMV*, 7–12.

Poux, F., 2019. The Smart Point Cloud: Structuring 3D Intelligent Point Data. PhD thesis, Université de Liège, Liège, Belgique.

Poux, F., Neuville, R., Hallot, P., Billen, R., 2017. Model for reasoning from semantically rich point cloud data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4, 107–115.

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(9), 1412.

Prieto, I., Izkara, J. L., Delgado, F., 2012. From point cloud to web 3D through CityGML. *2012 18th International Conference on Virtual Systems and Multimedia*, IEEE, 405–412.

Rusu, R. B., Marton, Z. C., Blodow, N., Dolha, M., Beetz, M., 2008. Towards 3D point cloud based object maps for household environments. *Robotics and Autonomous Systems*, 56(11), 927–941.

Schnabel, R., Wahl, R., Klein, R., 2007. Efficient RANSAC for point-cloud shape detection. *Computer graphics forum*, 26(2), 214–226.

Schwab, B., Beil, C., H., K. T., 2020. Spatio-Semantic Road Space Modeling for Vehicle–Pedestrian Simulation to Test Automated Driving Systems. *Sustainability*, 12(9), 3799.

Shahzad, M., Zhu, X. X., 2015. Automatic detection and reconstruction of 2-D/3-D building shapes from spaceborne TomoSAR point clouds. *IEEE Transactions on Geoscience and Remote Sensing*, 54(3), 1292–1310.

van Oosterom, P., Martinez-Rubi, O., Ivanova, M., Horhammer, M., Geringer, D., Ravada, S., Tijssen, T., Kodde, M., Gonçalves, R., 2015. Massive point cloud data management: Design, implementation and execution of a point cloud benchmark. *Computers & Graphics*, 49, 92–125.

Virtanen, J.-P., Kukko, A., Kaartinen, H., Jaakkola, A., Turppa, T., Hyyppä, H., Hyyppä, J., 2017. Nationwide Point Cloud - The Future Topographic Core Data. *ISPRS International Journal of Geo-Information*, 6(8), 243.

Vo, A.-V., 2017. Spatial data storage and processing strategies for urban laser scanning. PhD thesis, University College Dublin (Ireland).

Vosselman, G., Dijkman, S., 2001. 3D building model reconstruction from point clouds and ground plans. *International archives of photogrammetry remote sensing and spatial information sciences*, 34(3/W4), 37–44.

Willenborg, B., Sindram, M., Kolbe, T. H., 2018. Applications of 3D City Models for a Better Understanding of the Built Environment. M. Behnisch, G. Meinel (eds), *Trends in Spatial Analysis and Modelling*, Geotechnologies and the Environment, 19, Springer International Publishing, Cham, 167–191.

Wysocki, O., 2020. Semantic-based Geometry Refinement of 3D City Models for Testing Automated Driving. Master's thesis, Chair of Geoinformatics, Technical University of Munich. `http://mediatum.ub.tum.de/1580077`.

Wysocki, O., Xu, Y., Stilla, U., 2021. Unlocking Point Cloud Potential: Fusing MLS Point Clouds with Semantic 3D Building Models While Considering Uncertainty (accepted). *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*.

Xie, Y., Tian, J., Zhu, X. X., 2020. Linking points with labels in 3D: A review of point cloud semantic segmentation. *IEEE Geoscience and Remote Sensing Magazine*, 8(4), 38–59.

Xu, Y., Stilla, U., 2021. Towards Building and Civil Infrastructure Reconstruction from Point Clouds: A Review on Data and Key Techniques. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 1-1.

Yao, Z., Nagel, C., Kunde, F., Hudra, G., Willkomm, P., Donaubauer, 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*, 3(5), 1-26.