SEMANTICALLY ENRICHED VOXELS AS A COMMON REPRESENTATION FOR COMPARISON AND EVALUATION OF 3D BUILDING MODELS

M. Heeramaglore*, T. H. Kolbe

Technical University of Munich, Chair of Geoinformatics, Munich 80333, Germany
(medhini.heeramaglore; thomas.kolbe)@tum.de

Commission IV, WG IV/9

KEYWORDS: RichVoxel, Voxelisation, Semantic linking, 3D City Models, CityGML, IFC/BIM, Levels of Detail (LoD)

ABSTRACT

3D city models are fast becoming a key factor for planning, design and analysis in the AEC industry, and urban informatics. Data are being acquired from a variety of sources and each consequent representation contains inherent information about its location, geometry, structure and usability. These digital representations can range from unstructured models with very little semantic data such as 3D point clouds, and 3D mesh models or they can be highly structured and semantically enriched models following CityGML and IFC/BIM standards. Matching these diverse representations against one another can facilitate information flow and eventually lead to a coherent interconnected Digital twin. Consistency measures would be needed to compare one representation against another. For comparison of any kind, a common baseline needs to be established. A tentative approach, as outlined in this paper, is to convert all the model types into a common representation such as 'volumetric pixels' or 'voxels' but the concept of 'voxel' as we know it, is not enough to deal with the rich semantic and organizational structure of modelling representations such as IFC and CityGML. Voxelisation is a complex process and conventional conversion methods concentrate on translating the geometry between representation types but the semantic and class hierarchy information is usually not translated. This paper assesses the need for matching 3D models using a common representation (voxels), discusses the challenges of the voxelisation process and proposes the concept of a 'RichVoxel'.

1. INTRODUCTION

In recent years, there has been considerable research in identifying inconsistencies in 3D model generation and their correction. An automatic supervised classifier was developed by (Boudet et al., 2012) to perform self-diagnosis in dense urban areas using high-resolution aerial images. Ennafii et al., 2018 discuss an automatic method to assess the quality of 3D city models by compiling potential errors in a hierarchical and parametric taxonomy. Fanfani & Colombo, 2019 discuss the detection of structural changes by exploiting vision-based 3D models of a time-changing environment to detect changes while (Nguyen et al., 2017) proposed an approach to compare CityGML models on both geometric and semantic levels by employing a graph database. Kaartinen et al., 2005, in their research, have compared accuracies of photogrammetric and laser scanning methods for building extraction and partly compared the methods used for the same. All of the research cited above and more focus on one aspect of modelling, a single type of representation or the method of generation. There is not much research that has been published on the comparison of models across types of representations and levels of detail. Kolbe & Donaubauer, 2021 outlines the need for matching building models and the potential ways in which it can be achieved. Yao et al., 2020 outline the need for the preservation of semantic information during editing operations conducted on city models.

3D representation types, be it mesh models, point clouds, CityGML or IFC models have their own geometric, semantic and topological specifications characterised into classes. Some representations also have Levels of detail (LoD) which are described by the level of abstraction that each model possesses in terms of content, value, structural and semantic hierarchy, usability, etc., (Biljecki et al., 2017).

With abundantly available data, numerous applications and model representation types, there are always instances of overlap, bias or irregularities in different datasets. These could be due to limitations on the data type, method of acquisition, algorithm used for model generation, the application the data was acquired for, etc. Each representation, depending on the factors outlined above, could then describe one or more aspects of the real world while leaving out those that are not within its purview. In such cases, the primary investigation would be to figure out what parameters would be necessary to match the coherency of two given models. As described in the following chapters, a common representation type is required to match 3D models.

This paper focuses on the need for measuring the consistency between 3D representations using voxels as a potential 'common representation'. The voxelisation process of each representation is discussed in detail. Further, in the course of this paper, we define the concept of a Semantically Enriched Voxel that can handle the complexity presented by semantically enriched models.

2. BACKGROUND RESEARCH AND RELATED WORK

The research discussed in this section focuses on exploring, in detail, a) the complexities associated with the various 3D city representations, b) the need for consistency measures to match and connect the abovementioned representations, and c) the need for voxels as a common representation.

2.1 The complexity of 3D City modelling aspects

Digital representations of cities and city objects are complex in terms of geometry, and semantics, and contain layers of information. Though specific to the CityGML standard, (Kolbe, 2009) describes in detail the various modelling aspects which can be applied to other representations as well, such as a) Geometry, b) Semantics, c) Appearance, d) Topology and e) Time. The complexity of modelling aspects changes with representation type (Figure 4).

https://doi.org/10.5194/isprs-annals-X-4-W2-2022-89-2022

For example, a mesh model would consist of triangular or polygonal surfaces that represent the geometry of the real-world object while it does not describe the nature of the object. An IFC model, on the other hand, would consist of a richly organised semantic structure where each component has a definite place and a label describing what it is. In the same way, a model based on CityGML would have all objects classified into definite classes and sub-classes based on the Level of Detail (LoD) of the model.

2.1.1 Geometrical complexity

Irrespective of model type, all 3D models (such as point clouds, polygon models, CityGML and BIM) vary widely in terms of the modelling aspects, but they are essentially made up of points, curves, surfaces, solids or a combination of the above. Each of these geometries has different dimensional values viz. points are zero-dimensional, curves are one-dimensional, surfaces are two-dimensional and solids are three-dimensional. Since they are dimensionally different, it is not possible to directly compare these against one another. Clusters of points make up a surface and a group of surfaces make up a solid. Two models that need to be compared have to be broken down into common components. If a 3D point cloud needs to be compared with CityGML data, the boundary representations (B-rep) in CityGML cannot be matched with point data in 3D point clouds. The same situation arises when comparing 3D point clouds and IFC data, CityGML and IFC data where the solid components from IFC cannot be directly matched with points or B-reps. Between the commonly used representation types (3D point clouds, CityGML, IFC), there exists a clear hierarchy of geometry that needs to be converted into a common, comparable form to match any two given city models.

2.1.2 Semantic complexity

Beyond the geometry of the model, it is also important to compare the semantics of the object components to understand how well a 3D representation is describing a real-world object. For example, if a component in model ‘A’ is described as a wall and the same component in model ‘B’ is described as a window, it is important to identify which of the two models describes the real-world object better. Further, it is also possible that the models ‘A’ and ‘B’, while belonging to the same or different geometry types, could also have a different LoD. If ‘A’ belongs to CityGML LoD2 and ‘B’ is an IFC model, the classes should be scaled down to match the lowest level of detail i.e. the class ‘window’ would need to be aggregated into a higher class ‘wall’ (Figure 5). The above comparison instance is an example of the role of semantic comparison as well as the problems that arise when dealing with varying geometry types and semantic structures.

In a 3D city model, semantics and geometry are often linked and dependent on each other. When geometry and semantics are considered together, the permutation of scenarios is numerous. As per the crux of the topic discussed in (Stadler & Kolbe, 2007), the permutations of the relationship between geometry and semantics can be broadly classified into a) Unstructured Geometry, Unstructured Semantics; b) Unstructured Geometry, Structured Semantics; c) Structured Geometry, Structured Semantics and d) Structured Geometry, Unstructured Semantics. When these permutations are further coupled with different geometry types (point clouds, mesh, B-reps, CSG and sweep volumes), the complexity of the modelling paradigm grows exponentially.

Based on these various permutations, Figure 4 summarises which modelling aspects from which dataset can be compared against which modelling aspects from another dataset.

2.2 Evaluation and consistency measures.

Visual overlap of models can help with the comparison of visible, surface elements but the objective comparison of coherency between models requires some definite parameters and metrics against which a decision can be made. For example, two models of the same scene can be overlapped and look very similar to each other, but due to the variation in geometry, LoD, and maybe, semantics, there could be elements that are not visible but have geometry and occupy volume which cannot be compared or there could be topological differences in the model which are not visibly apparent.

Apart from an accurate representation of real-world objects, matching models across representation types can facilitate the flow of data and models can be routinely updated. This eases the understanding of interlinked processes i.e. enriching one 3D model with information from another. E.g., models with semantic structuring (IFC, CityGML) are quite expensive and time-consuming to create and update. They require high computation power and intensive manual interference. However, other 3D representations such as mesh models can be automatically generated and updated frequently. If a match can be established between these model types, changes in the built environment can be identified and updated across model types. A good example of such a data interdependence is explained succinctly in (Tuttas et al., 2015), where an IFC model is compared against a point cloud model for monitoring construction progress. Another example of the need for data interoperability is shown in (Wysocki et al., 2021) where data from MLS point cloud is combined with semantic city models to improve the quality of 3D data capture.

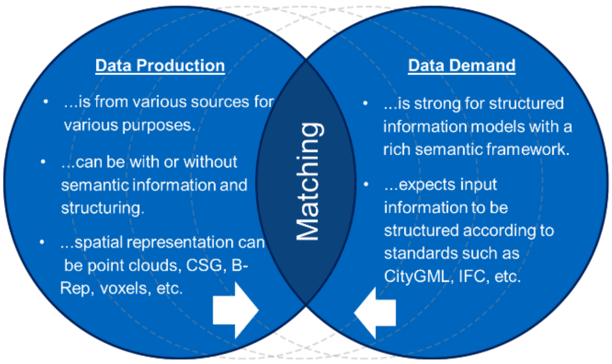

Figure 1 Role of evaluation in bridging the gap between data production and data demand.

Currently, data generation is focused on employing and refining AI algorithms to automatically generated accurate (spatially and visually) data that can be used for various applications. Automatic model generation is still widely focused on getting the most accurate visual representation of the built environment out of raw data without a specific focus on the semantic structuring of the final models. What is required for research and analysis are highly structured models with detailed semantic data. Hence the gap between data production and data demand. When 3D models can coherently be matched against each other, the resulting information flow would bridge this gap (Figure 1). While this project focuses on matching two 3D models to check how coherent they are against each other, similar matching techniques can be employed to match an automatically generated 3D model with a 'ground-truth' model which has been manually corrected and matched with reality. This way, the results of such a match can be fed back into the AI algorithms to refine and automatically generate better, more structured semantic models.

Therefore, consistency measures are invaluable for generating and/or updating accurate models, improving the generation capabilities of neural networks and allowing information flow between end-user applications by facilitating the development of an inter-connected Urban Digital Twin.

2.3 Need for a common representation

Objective comparison is always based on measurable facts. As in mathematics, it is not possible to randomly compare any two entities without first identifying a common denominator.

The ideal common representation would need to have

- Definite geometry with locational aspects – there should be no ambiguities!
- Provision to hold semantic information along with structural organization information.
- Should have a schema that can support class hierarchy and each entity in the model should be mapped or linked to the original representation.
- Link surfaces to texture information.

2.4 'Voxel' as an ideal common representation

Volumetric pixels, Voxels are not a new concept in the realm of 3D modelling. Such volume-based representations provide a good alternative to surface and solid representations (Young & Krishnamurthy, 2018). The reasons a voxel is beneficial to 3D analysis are compellingly outlined by (Zlatanova et al., 2016) where a voxel has been defined as a single primitive data that represents the properties of a real object. Further enumerating the advantages of voxels, (Zlatanova et al., 2016) also discusses the ease of volumetric analysis with voxels and the fact that every object is represented by only one primitive type as opposed to other representation such as CityGML and IFC, which are collections of point, lines, surfaces, etc. Voxels have definite and tangible volumes which can be used for volumetric calculations and geometric comparison.

Voxels also offer a discrete approximation of the geometry with the advantage that individual voxels can be associated with thematic information. Each voxel can be mapped onto real-world coordinates as an unambiguous, definite representation. In principle, it is also possible to attach or link semantic information from the original representation to the voxel representation.

Further adding to the advantages, the resolution of the voxels can be adjusted according to the LoD of the models being compared. For example, if an IFC building model is being compared to a LoD4 CityGML model, the size of the resolution can be comparable to the width of the wall or any other element that is common to both. Similarly, coarse resolutions can be adopted for models with lower LoD. Voxel resolution can also be arbitrarily decided based on the semantic tags and class hierarchy present in the input model. For the CityGML LoD3 model, it would be prudent to decide the voxel size based on the size of the components present in the schema. The sizing of the voxel is based on semantic and geometric identifiers and is enumerated in (Zlatanova et al., 2016).

While there are advantages to the voxel format, there are also disadvantages. The conversion process is often lossy and reverting to the original presentation would be difficult or impossible, especially with the semantic models. In voxelisation, each surface or the solid is broken down into discrete cubes or 3D pixels. The integrity of the overall geometry is lost and replaced by the collective geometry of the voxels. The semantics of the original component would need to be mapped to individual voxels to carry on the meaning from the original representation.

Despite limitations, voxels can be considered as a common denominator for other types of 3D representations as they can be considered a geometric primitive (analogous to the 2D pixel) and need not have to be further broken down. Their ability to be assigned real-world spatial coordinates and hold semantic information, while consisting of definite geometry (useful for volumetric calculations) is an advantage that is absent from other model types.

2.5 Voxels as a medium for cross-representation comparison

Since all of the geometry types (points, surfaces, and solids) under consideration can be converted to voxels, it would seem an ideal representation for comparison. Additionally, as discussed above, a voxel representation supports volumetric and morphometric analysis.

Once voxelised, any given models 'A' and 'B' can be directly compared against each other using simple mathematical operations such as directly counting and adding the volumes of individual voxels to compare the overall occupancy of the models. Simple theories such as set intersection can be used to identify inconsistencies in models based on their geometry and semantics, etc. As explained in Figure 2., when a Model 'A', voxelised from CityGML LoD3 is compared against a Model 'B', voxelised from BIM/IFC, a simple 3D intersection can identify the volumetric differences in the model and identify missing elements by the semantic tags attached to the voxels.

While in theory, voxels are an ideal common representation for analysis and comparison, the voxelisation process of various representations is a challenge. Voxels have been a highly sought-after medium for 3D City modelling and analysis. There has been quite a lot of research done to support voxels as a tool for spatial and volumetric analysis. Krijnen et al., 2021; Li et al., 2020; Liu et al., 2021; Nourian et al., 2016; Poux & Billen, 2019; Wang et al., 2020; Xu et al., 2021, all advocate the use of voxels for analysis of the building and built environments for various applications. Each of the works cited above focuses on the conversion of one specific geometry type to voxels viz. point clouds, surface and boundary representations, and solid geometry. While the voxelisation algorithms proposed in (Nourian et al., 2016) are good examples to convert point, curve and surface datasets to voxels, they would be insufficient for a direct voxelisation of large 3D city datasets made up of complex geometry and semantic structuring (e.g. a 3D model conforming to IFC or CityGML standards).

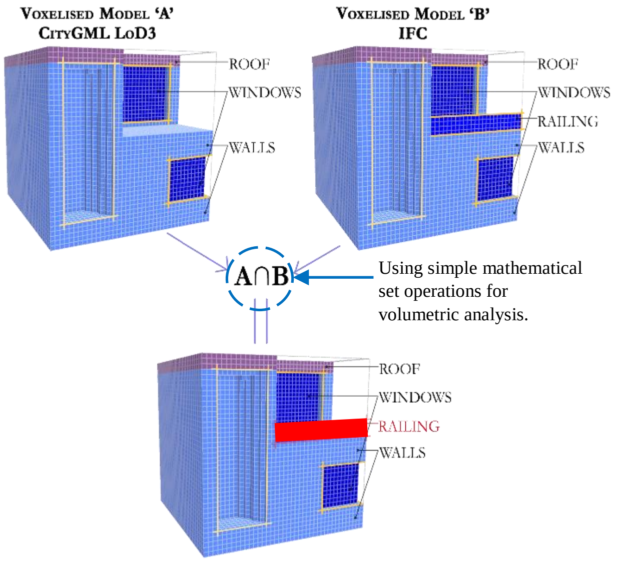

Figure 2 Conceptual approach to the 3D intersection of two voxel models. (The red voxels in the railing highlight the differences)

2.5.1 Voxelisation of 3D Point clouds

On a semantic level, point clouds (such as the LAS format) have a much more simply organised semantic structure as compared to CityGML or IFC. Such a semantic structure could be translated directly to a voxel format. The advantages of using voxels over point clouds are enumerated by (Xu et al., 2021). As part of their study, they have analysed the potential of voxel-based representations in the construction industry. Poux & Billen, 2019 developed a semantic segmentation framework that groups points into voxels for further analysis while (Aijazi et al., 2013) employed a "super-voxel" approach to segmentation of 3D urban point clouds. Nourian et al., 2016 discuss various voxelising algorithms for converting spatial data into voxels. In (Nourian et al., 2016), the developed voxel algorithms have been tested on point, curve, and surface models in Rhino3D.

2.5.2 Voxelisation of Surface representations

Surface representations such as B-rep can be voxelised and these voxels can then be aggregated into a category based on the semantic tags available from the original representation. The concept of "super-voxel" as defined in (Aijazi et al., 2013) is especially useful if a cluster of voxels (belonging to a certain surface) needs to be grouped and identified as one. The research and experimentation by (Willenborg et al., 2016) show that the conversion of surface models (CityGML in this case) into voxels can be done while also translating the semantic information. Konde & Saran, 2017 have used a coarse voxel representation of a CityGML model to simulate traffic data on Cesium ion Web. The research by (Zlatanova et al., 2016) points to the challenges that come with trying to convert CityGML surfaces to voxels. The research also identifies corresponding challenges in conversion of IFC to voxels as well.

2.5.3 Voxelisation of IFC models

When solids (e.g. BIM Components) are converted to voxels, not only semantics, but the geometry also has a hierarchy. The solid could be converted to a collection of surfaces and then to a voxel but it would lose the inherent information that comes from its volume. When a solid component is converted to a voxel, the volume enclosed by the solid in question is also converted to voxels (Figure 3). This is an interesting area of research with contributions from several studies. Krijnen et al., 2021 converted IFC to voxels for thermal analysis since voxels offer a robust processing method across building scales and types. Liu et al., 2021 discuss the use of voxel index analysis to automatically identify and tag semantic information to exterior building elements. A part of the research by (Wang et al., 2020) puts forth interesting points regarding the topology of voxels in a solid IFC component.

3. A "SEMANTICALLY ENRICHED VOXEL"

The initial process of voxelisation is complicated in itself and the comparison amongst different representation types presents additional challenges. Each voxel model generated would need to support multiple levels of information classification to support the semantic structuring of the various representation types. Translating information from classes and sub-classes of objects, specifically for IFC and CityGML models will require careful planning and research when converting semantic models surfaces to voxels. Apart from the semantic information, voxels models should also adapt to the dimensionality of the geometry of the original model. For example, a comparison of the CityGML model with two-dimensional surfaces against an IFC model with Constructive Solid Geometry (CSG) (Figure 3). Just as a 2D image cannot be compared to a corresponding 3D object directly, a surface cannot be compared to a solid even if it is describing the same real-world object.

With the voxel model of the B-rep, the semantic information that is tagged to the voxels is directly inherited from the original representation. With solid components, the voxels must have additional associations i.e. interior or exterior voxels. Since voxelisation of solids voxelises the enclosed volume as well as the boundary surfaces, voxels belonging to each sub-category need to be marked i.e as belonging to the interior volume or bounding surface. Then, boundary voxels need to be extracted for comparison against models with similar geometric properties from other representations viz. CityGML (Figure. 3).

Further adding to the challenge in the case of solid geometries, voxels in the voxelised IFC model can belong to multiple components (e.g. in the junctions of overlap between walls) and in such cases, the precedence of a certain class of semantics needs to be defined to avoid inconsistencies in matching. In such a case the voxel would have to be semantically linked to both walls (interior and exterior) but the tag of one component is given precedence for comparison with another similar model. This requires that the schema of the common representation be flexible and be associated with different classes of information as per the input model. Current voxel formats are neither standardised nor can support this kind of semantic information structure.

While the concept of 'super-voxel' exists (Aijazi et al., 2013; Babahajiani et al., 2015), this simply refers to the aggregation of voxels into superclusters to reduce the computational efforts while preserving the geometric and topological robustness of individual voxels. While an interesting approach, this method reduces the computational load and divides large datasets into

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-X-4-W2-2022-89-2022

manageable parts. However, the core schema of a voxel needs to be modified so that it can support semantic information from various models.

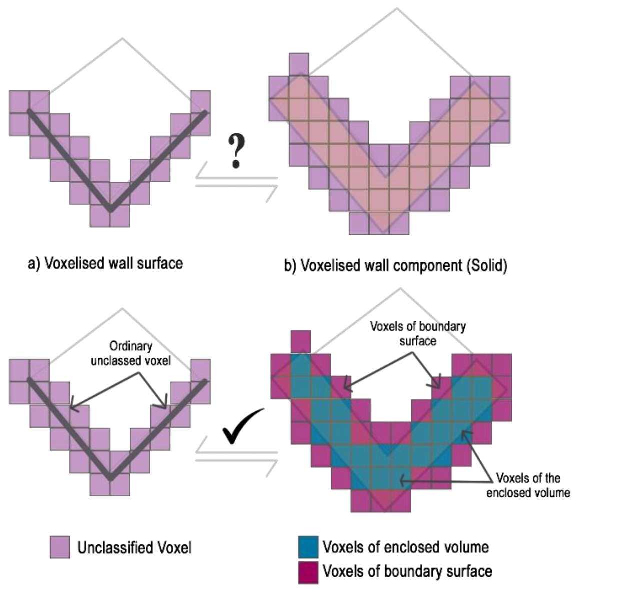

Figure 3 Considerations when comparing multidimensional data against one another.

This led to the proposal of a 'Semantic Voxel' or a 'Semantically Enriched Voxel'. It is a proposed voxel format that can act as a common representation and also support information from all types of geometric and semantic representations. For comparison across geometry types and semantic structuring, a standard data schema consistent with other representation types is very important. Even if the schema is not as richly structured as IFC or CityGML, it should be comparable with a simple one like the LAS format. With a simple data structure, it should still be able to translate information from complex schemas (Figure 5).

As opposed to a standard voxel, a 'RichVoxel' would have a simple semantic structure like that of the LAS format but also flexibly expand to support complex semantics of CityGML and IFC. For example, a CityGML LoD2 model is to be compared with an IFC model. As described in Figure 4, the modelling aspects that can be compared in such a case would be geometry (G), Semantics up to level 1 (S1) and possibly appearance. The IFC model has detailed components, interior spaces, openings, roof superstructures, etc. The LoD2 CityGML model, on the other hand, only has a basic roof profile and wall surfaces. Hence, the semantic information level is S1 (Figure 5). When two models are compared against each other, the semantic information is brought down to the level common to both. Another example is if a LoD1 CityGML model is compared against a LoD3 CityGML model. A LoD3 model would have semantic information up to S3 (Figure 5) i.e. windows, doors, balconies, etc. A LoD1 model only consists of unclassified surfaces that make up a building i.e. S1 (Figure 5). Hence, when compared against each other, they can be compared against geometry (G) and semantics (S1). The structure of the data hierarchy for the 'RichVoxel' in Figure 5 has been derived based on the modelling aspects described in Figure 4.

The following are a few important aspects that have been taken into consideration for the proposal of the 'RichVoxel'.

- The geometry of the 'RichVoxel' or a 'Semantically Enriched Voxel' is similar to that of a voxel. But, there is a need for a standardised schema along with I/O capabilities and a definite geometry that can act as a readily-readable data format. The schema should be able to store each voxel as an individual entity along with all the associated information.

		3D Point Clouds	3D Mesh models	CityGML				IFC**
				LoD1	LoD2	LoD3	LoD4	
3D Point Clouds		G, S1*,A*	G, A*	G, S1*	G, S1*, A*	G, S1*, A*	G, S1*, A*	G, S1*, A*
3D Mesh models		G, A*	G, A*	G	G, A*	G, A*	G, A*	G, A*
City GML	LoD1	G, S1*	G	G, S1	G, S1	G, S1	G, S1	G,S1
	LoD2	G, S1*, A*	G, A*	G, S1	G, S2, A*	G, S2, A*	G, S2, A*	G, S2, A*
	LoD3	G, S1*, A*	G, A*	G, S1	G, S2, A*	G, S3, A*	G, S3, A*	G, S3, A*
	LoD4	G, S1*, A*	G, A*	G, S1	G, S2, A*	G, S3, A*	G, S3, A*	G, S3, A*
IFC**		G, S1*, A*	G, A*	G, S1	G, S2, A*	G, S3, A*	G, S3, A*	G, S3, A*

G – Geometry
S – Semantics
S1 – Semantics Level 1 / Coarse semantics (Identification of city objects such as building, vegetation, etc.)
S2 – Semantics Level 2 (Identification and classification of building surfaces into walls, roofs, etc.)
S3 – Semantics Level 3 (Identification of façade elements, materials etc.)
(Described in Figure. 5)
A – Appearance
*May or may not contain such information at any level.
**Levels of development is not considered yet. Final IFC model is considered to be fully georeferenced.

Figure 4 Modelling aspects that have to be taken into account when comparing different 3D model representations.

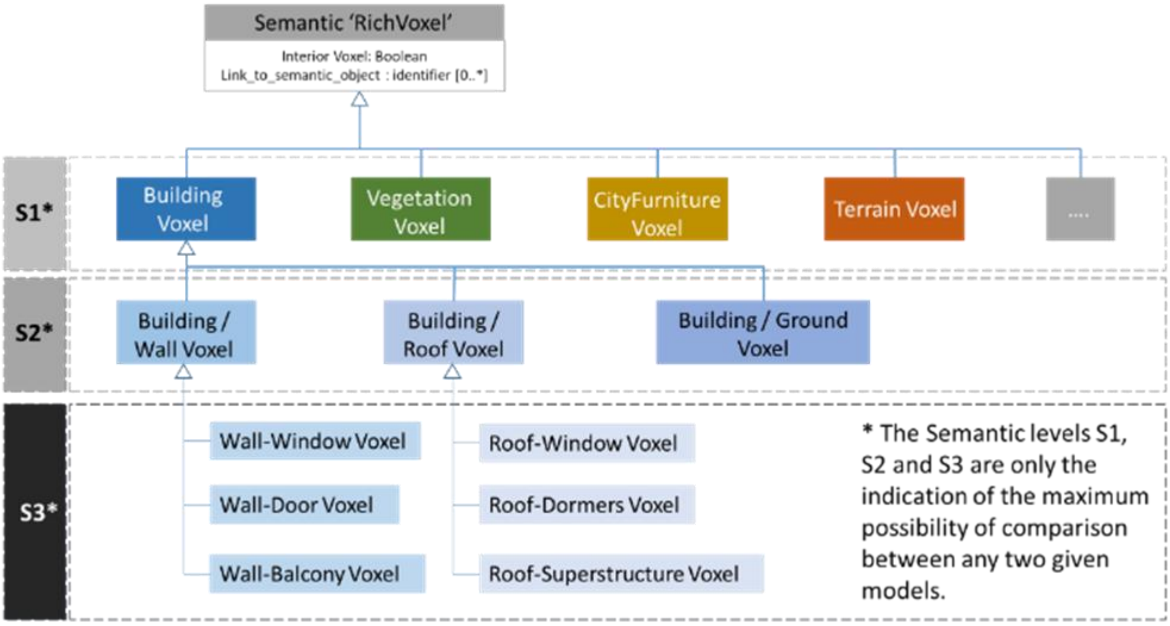

Figure 5 A conceptual understanding of the class hierarchy required in a 'RichVoxel

- The 'RichVoxel' would also need to hold additional data such as the location coordinates, spatial reference system, RGB colour information, opacity/transparency data, material information, class labels, etc.
- Also, each voxel should include a reference to the IFC or CityGML object from which it was derived. For example, when an IFC building model is converted to a RichVoxel, the voxels corresponding to each component should be tagged as per their class label. On voxelising the BIM models, the RichVoxel of a component should be able to incorporate additional tags of subcomponents viz. a class wall with sub-classes of interior/exterior voxel, openings, material change, etc.
- A RichVoxel should have a clear class hierarchy. As explained in the examples previously and Figure 4, a class hierarchy helps us identify which class of semantic information is common for any two given models (Figures 4&5).
- A 'RichVoxel' should be able to support multiple associations. Suppose an IFC model has overlapping components like two walls. The voxels in such an overlapping junction would not only need to be classified as belonging to the interior volume or the boundary surface (Figure 3) but also as belonging to two components. Such a case of multiple associations could help us identify inconsistencies in the model as well as hidden objects or differences in geometry and topology.

Additionally, each RichVoxel could be aggregated into super-voxels based on semantic tags to reduce the computation efforts for algorithms.

4. VOXELISATION OF 3D BUILDING/CITY MODELS

Conversion from one representation to another affects the core structure of the model. In the voxelisation of 3D city/building models, the addition of semantic structure and spatial attributes significantly challenges the conversion process as discussed in the previous chapters. Readily available tools and packages do not directly read and convert 3D spatial data formats to voxels but use intermediary conversions which results in loss of semantic data. While there are studies such as (Nourian et al., 2016; Xu et al., 2021) that discuss voxels as a representation of 3D city models for modelling and analysis, the focus is on the visualisation aspect of the representation. Data storage is another challenge as there are no standardised voxel formats that are directly read by geospatial programs. Tools developed in C++, Python and PostGIS have been used for the initial voxelisation of semantic models to support voxels as a valid common representation.

4.1 Initial voxelisation approaches.

The first attempt at converting CityGML models to voxels was done using 3DCityDB and a 'city_voxeliser' tool (Schwab, 2021). The voxelisation algorithm uses a CityGML model as input through the 3DCityDB. The objects in the model are detected and based on the resolution defined, voxels are visualised as a point cloud with each point being the centre of a voxel. The output, written to a common point cloud format such as point files, can then be visualised using a platform such as the Feature Manipulation Engine (FME). The advantage of this algorithm is that each point is unique and can be selected individually. Such a discretised dataset might help for better objective analysis. However, most commonly used point cloud formats (except LAS) hold only the location information and do not support semantic organizational structure.

The next experiment was with Open3D in Python. Open3D tools are generalized and are not equipped to deal with semantically rich modelling formats such as CityGML and IFC. For such tools, the input needs to be simpler such as the OBJ format. Such an intermediate conversion results in the loss of semantic and structural organisation information. Voxels can then be exported as point files with these points as centres of the individual voxels. This approach poses the same problem in terms of visualisation and data storage as the previous method.

4.2 Voxelisation using 3DCityDB and PostGIS.

This algorithm developed by and discussed in (Willenborg et al., 2016) uses the 3DCityDB and PostGIS to create a voxel model of any given CityGML model. In this method, a bounding box for the CityGML model is first defined and a generic voxel grid is generated. The GML model is then intersected with the voxel grid to generate the voxelised model of the original representation.

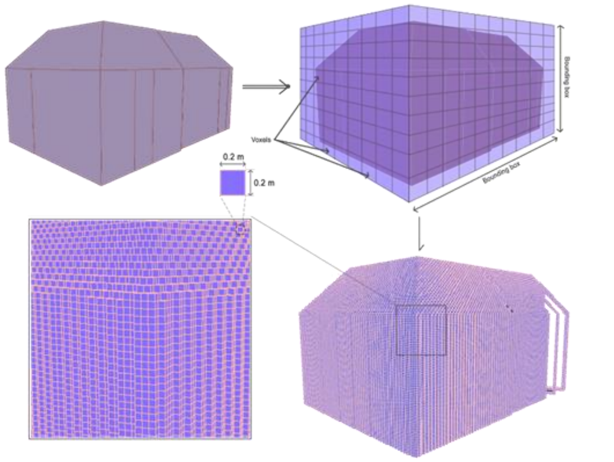

Figure 6 Voxelisation of GML models in PostGIS into Voxels.

Using this intersection method, the geometry, as well as its corresponding semantic information, can be absorbed into the voxel. The size of the voxel can be flexibly defined depending on the LoD in the model. The resulting voxels are stored in the database tables as polyhedral surfaces which can be directly visualised in FME or QGIS.

5. RESULTS AND DISCUSSION

Two 3D city models of the same real-world building from the city of Ingolstadt, Germany with varying LoD were chosen for voxelisation.

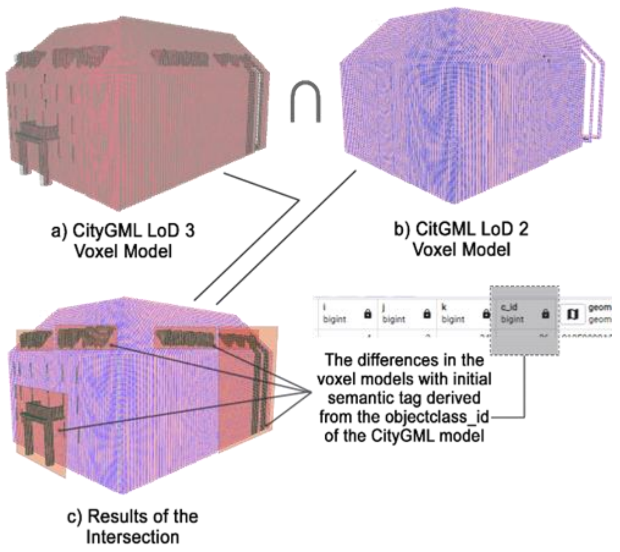

Figure 7 Comparison of Voxel models of LoD2 and LoD3 CityGML models from Ingolstadt, Germany

Following the voxelisation of the models as per the method described in Chapter 4.2, the two models were overlaid and visualised. As can be seen in Figure 7 below, a 3D intersection of two voxel models can help in the identification of differences between the models. Further, voxels can absorb the required semantic information from the original model. In this instance, the semantic tag is stored as a numerical value indicating the objectclass_id from the original model.

Unlike regular geometry types used in spatial analysis – point, line, polygon – polyhedral surfaces are not readily readable on most platforms. This is a viable process but it is time-consuming and computationally intensive. The process needs to be optimized for faster voxel model generation. Though the above results are promising, it still does not address the challenges with voxel data storage and I/O. As outlined in chapter 4, initial voxelisation shed light on the lack of standardised formats that can support the storage and visualisation of voxels. The concept of a 'RichVoxel' plays a key role here to bridge the gap between processing, visualisation and analysis of the voxel models.

As mentioned in chapter 3, the ideal representation format should be able to handle the storage and visualisation of voxel primitives as unique and unambiguous entities. From background research, it is evident that this issue has already been under consideration and there are several solutions outlined in (Krijnen et al., 2021; Li et al., 2020; Liu et al., 2021; Willenborg et al., 2016; Zlatanova et al., 2016). Li et al., 2020; Willenborg et al., 2016 have used PostgreSQL and PostGIS for voxel data storage and management. For attribute linking and storage, as well as simple morphometric operations on voxels, this is a suitable platform for analysis.

As enumerated in Chapter 3, the format chosen should also be able to expand or retract to a particular class level depending on the highest possible level of comparison between input model representations. For example, input model 'A' belonging to CityGML LoD3 would have classes such as Wall surfaces, roof surfaces, ground surfaces along with windows and other openings etc. An input Model 'B', in CityGML LoD2, developed from a 3D point cloud might just have the exterior wall surfaces, and roof profile. In such a case, when voxelising and comparing, it should be possible to assimilate the windows and other façade elements into the 'wall' class, since it is common amongst both. This means that the voxel model of the LoD3 representation needs to automatically adapt to the S2 level of classification since that is the highest possible instance of common features between LoD2 and LoD3.

This need for a flexible class hierarchy further supports the idea of having a standardised voxel format such as the 'RichVoxel' on par with LAS, Shapefiles, CityGML, IFC, etc.

6. CONCLUSION AND WAY FORWARD

Based on related work and experimentation with open source libraries, 'RichVoxel' has been conceptualised to address the gaps in current research on 3D city models and voxelisation. While theoretically and logically the concept covers most of the issues that one would encounter when converting models from one representation to voxels, further experimentation with a variety of datasets would be needed before encoding a standardised format for voxels. With the gaining prominence of

https://doi.org/10.5194/isprs-annals-X-4-W2-2022-89-2022 | © Author(s) 2022. CC BY 4.0 License.

cloud-based data storage, RichVoxel would need to be able to be hosted on the web and queried and managed using cloud-based services as well.

In the immediate future, the workflow for this project involves defining a generic methodology that can convert the various representation into voxels while preserving the semantic and topological relationships. Experimentation and research to target the storage, management and visualisation of data, and the development of the 'RichVoxel' is ongoing. Once a common representation is achieved, further research would address the development of metrics for the comparison of 3D city models and objectively measuring their coherency based on the 'RichVoxel' representation.

7. ACKNOWLEDGEMENTS

This project is a part of a larger project called 'AI4TWINNING' and is funded by the Georg Nemetschek Institute (GNI) Artificial Intelligence for the Built World at the Technical University of Munich (TUM).

REFERENCES

Aijazi, A., Checchin, P., & Trassoudaine, L. (2013). Segmentation Based Classification of 3D Urban Point Clouds: A Super-Voxel Based Approach with Evaluation. *Remote Sensing, vol. 5, issue 4, pp. 1624-1650, 5*, 1624-1650.

Babahajiani, P., Fan, L., Kamarainen, J., & Gabbouj, M. (2015, 19-21 Oct. 2015). Automated super-voxel based features classification of urban environments by integrating 3D point cloud and image content. 2015 IEEE International Conference on Signal and Image Processing Applications (ICSIPA),

Biljecki, F., Ledoux, H., & Stoter, J. E. (2017). Delft University of Technology An improved LOD specification for 3 D building models.

Boudet, L., Paparoditis, N., Jung, F., Martinoty, G., & Deseilligny, M. (2012). A Supervised Classification Approach Towards Quality Self-Diagnosis Of 3D Building Models Using. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 36*.

Ennafii, O., Le-Bris, A., Lafarge, F., & Mallet, C. (2018). Semantic evaluation of 3D city models.

Fanfani, M., & Colombo, C. (2019). *Structural Change Detection by Direct 3D Model Comparison*.

Kaartinen, H., Hyyppä, J., Gülch, E., Vosselman, G., Hyyppä, H., Matikainen, L., Hofmann, A., Mäder, U., Persson, Å., Söderman, U., Elmqvist, M., Ruiz, A., Dragoja, M., Flamanc, D., Maillet, G., Kersten, T., Carl, J., Hau, R., Wild, E., & Vester, K. (2005). Accuracy of 3D city models: EuroSDR comparison. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *36(3/W19)*.

Kolbe, T. H. (2009). Representing and Exchanging 3D City Models with CityGML. In J. Lee & S. Zlatanova (Eds.), *3D Geo-Information Sciences* (pp. 15-31). Springer Berlin Heidelberg.

Kolbe, T. H., & Donaubauer, A. (2021). Semantic 3D City Modeling and BIM. In W. Shi, M. F. Goodchild, M. Batty, M.-P. Kwan, & A. Zhang (Eds.), *Urban Informatics* (pp. 609-636). Springer Singapore.

Konde, A., & Saran, S. (2017). Web enabled spatio-temporal semantic analysis of traffic noise using CityGML.

Krijnen, T., El-Diraby, T., Konomi, T., & Attalla, A. (2021). Thermal analysis of IFC building models using voxelized geometries. 11.

Li, W., Zlatanova, S., & Gorte, B. (2020). Voxel data management and analysis in PostgreSQL/PostGIS under different data layouts. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *VI-3/W1-2020*, 35-42.

Li, W., Zlatanova, S., Yan, J. J., Diakite, A., & Aleksandrov, M. (2019). A geo-database solution for the management and analysis of building model with multi-source data fusion. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, *XLII-4/W20*, 55-63.

Liu, X., He, C., Zhao, H., Jia, J., & Liu, C. (2021). ExteriorTag: Automatic Semantic Annotation of BIM Building Exterior Via Voxel Index Analysis. *IEEE Computer Graphics and Applications*, *41*(3), 48-58.

Nguyen, S., Yao, Z., & Kolbe, T. (2017). *Spatio-semantic comparison of large 3d city models in citygml using a graph database* (vol. Iv-4/w5).

Nourian, P., Gonçalves, R., Zlatanova, S., Ohori, K. A., & Vu Vo, A. (2016). Voxelization algorithms for geospatial applications: Computational methods for voxelating spatial datasets of 3D city models containing 3D surface, curve and point data models. *MethodsX*, *3*, 69-86.

Poux, F., & Billen, R. (2019). Voxel-based 3D Point Cloud Semantic Segmentation: Unsupervised Geometric and Relationship Featuring vs Deep Learning Methods. *ISPRS International Journal of Geo-Information*, *8*(5).

Stadler, A., & Kolbe, T. H. (2007). Spatio-semantic coherence in the integration of 3D city models.

Schwab, Benedikt (2021). CityDB voxeliser alogorithm.

Tuttas, S., Braun, A., Borrmann, A., & Stilla, U. (2015). Validation of bim components by photogrammetric point clouds for construction site monitoring. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *II-3/W4*, 231-237.

Wang, Q., Zuo, W., Guo, Z., Li, Q., Mei, T., & Qiao, S. (2020). BIM Voxelization Method Supporting Cell-Based Creation of a Path-Planning Environment. *Journal of Construction Engineering and Management*, *146*, 04020080.

Willenborg, B., Sindram, M., & Kolbe, T. (2016). *Semantic 3D City Models Serving as Information Hub for 3D Field Based Simulations*.

Wysocki, O., Xu, Y., & Stilla, U. (2021). Unlocking point cloud potential: fusing mls point clouds with semantic 3d building models while considering uncertainty. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *VIII-4/W2-2021*, 45-52.

Xu, Y., Tong, X., & Stilla, U. (2021). Voxel-based representation of 3D point clouds: Methods, applications, and its potential use in the construction industry. *Automation in Construction*, *126*, 103675.

Yao, S., Ling, X., Nueesch, F., Schrotter, G., Schubiger, S., Fang, Z., Ma, L., & Tian, Z. (2020). Maintaining Semantic Information across Generic 3D Model Editing Operations. *Remote Sensing*, *12*(2), 335.

Young, G., & Krishnamurthy, A. (2018). GPU-accelerated generation and rendering of multi-level voxel representations of solid models. *Computers & Graphics*, *75*, 11-24.

Zlatanova, S., Ghadikolaee, P. N., Goncalves, R., & Vo, A.-V. (2016). Towards 3D raster GIS: On developing a raster engine for spatial DBMS.