

Article

A Knowledge Base for Automatic Feature Recognition from Point Clouds in an Urban Scene

Xu-Feng Xing ^{1,2,*} , Mir-Abolfazl Mostafavi ^{1,2}  and Seyed Hossein Chavoshi ^{1,2}

¹ Department of Geomatics Sciences, Université Laval, Québec, QC G1V 0A6, Canada;

Mir-Abolfazl.Mostafavi@scg.ulaval.ca (M.-A.M.); seyed-hosseini.chavoshi.1@ulaval.ca (S.H.C.)

² Center for Research in Geomatics, Université Laval, Québec, QC G1V 0A6, Canada

* Correspondence: xufeng.xing.1@ulaval.ca; Tel.: +1-581-888-9786

Received: 4 October 2017; Accepted: 11 January 2018; Published: 16 January 2018

Abstract: LiDAR technology can provide very detailed and highly accurate geospatial information on an urban scene for the creation of Virtual Geographic Environments (VGEs) for different applications. However, automatic 3D modeling and feature recognition from LiDAR point clouds are very complex tasks. This becomes even more complex when the data is incomplete (occlusion problem) or uncertain. In this paper, we propose to build a knowledge base comprising of ontology and semantic rules aiming at automatic feature recognition from point clouds in support of 3D modeling. First, several modules for ontology are defined from different perspectives to describe an urban scene. For instance, the spatial relations module allows the formalized representation of possible topological relations extracted from point clouds. Then, a knowledge base is proposed that contains different concepts, their properties and their relations, together with constraints and semantic rules. Then, instances and their specific relations form an urban scene and are added to the knowledge base as facts. Based on the knowledge and semantic rules, a reasoning process is carried out to extract semantic features of the objects and their components in the urban scene. Finally, several experiments are presented to show the validity of our approach to recognize different semantic features of buildings from LiDAR point clouds.

Keywords: LiDAR; feature recognition; urban scene; ontology; knowledge base; semantic reasoning

1. Introduction

Virtual Geographic Environments (VGEs) are a new generation of geospatial technologies providing advanced modeling, simulation, and visualization capacities for better representation, analysis and understanding of the complex geographic world [1,2]. Construction of virtual geographic environments for urban scenes allows better understanding of diverse static and dynamic geographic phenomena, including urban development, traffic [3], air pollution [4,5], crowd behavior [6], urban planning, etc. Geometrically precise and semantically enriched representation of geographic environments allow spatial reasoning such as navigation and path planning based on Multi-Agent Geo-Simulation in VGEs [7]. LiDAR technology makes it possible to observe real-world environments rapidly and record very detailed geographic information in the form of point clouds in support of the generation of precise 3D VGEs. However, automatic 3D modeling and feature recognition from LiDAR point clouds are very complex tasks. This becomes more complex due to the presence of occlusion problems, and uncertainty in the data.

In general, automatic 3D modeling from point clouds implies: (1) classification of points belonging to the same object; (2) segmentation of objects and their components; (3) definition of relations between objects components; and (4) recognition of object types and their components. Extraction and recognition of objects from a point cloud imply not only the extraction of geometric features of the object (geometric primitives, size, shape, borders, etc.) but also it involves their semantics. We will

refer to the latter as semantic feature extraction throughout this paper. Semantic and geometric features are then two complementary sets of knowledge that we need to extract and recognize different object types from point clouds. Object extraction and recognition requires the integration of semantic features with geometric features. For example, a planar segment extracted from point clouds could represent a wall, a component of a roof or a part of the road. Assigning the right semantics to geometric objects detected in a point cloud is a complex task.

Recently, knowledge-based solutions have been introduced to support automatic 3D modeling and object recognition from LiDAR point clouds. For example, knowledge of size, shape, position, orientation, topological relations between building components as well as physical properties, such as color and texture, can be used to recognize and model its components such as walls, doors, roofs, and windows [8]. Semantic network technologies are also employed to describe potential relations between different components of buildings [9]. Indeed, the topological relations among the components of objects with complex structures is essential to identify semantic features of objects with varying topologies among components, for example, complex geometries and roof shape. Additionally, the recognition of higher-level semantic features (such as the architectural styles of buildings) requires more detailed qualitative knowledge. Semantic reasoning based on this knowledge would be essential for their modeling and recognition.

Ontologies can formally represent knowledge of spatial objects. Ontology is defined as the specification of conceptualizations that helps to make information communication and sharing among programs and humans more efficient [10,11]. It can be represented as a set of logical axioms to explain the intended meaning of a concept [12]. For sharing information in automatic 3D modeling, the formalized representation of knowledge is an essential step in building a knowledge base. An ontology can be represented as a semantic network, which is a graph where vertices indicate concepts and edges describe the relations among those concepts. For machine processing, more specialized formalization of knowledge, such as Resource Description Framework (RDF), Web Ontology Language OWL and Semantic Web Rules Language (SWRL), is needed for representing knowledge, defining rules and carrying out semantic reasoning on the knowledge.

Knowledge-based solutions are increasingly used to improve the accuracy, and the quality of results, especially for feature recognition in the automatic 3D modeling from LiDAR point clouds [8,9,13,14]. However, there are still challenges for automatic 3D modeling and object recognition from point clouds in a complex urban scene. These challenges include the diversity of object types, the complexity of their shape and their spatial relations.

In this paper, we propose a knowledge-based approach for automatic object recognition from LiDAR point clouds in urban scenes. First, we define several modules for the ontology to organize concepts describing an urban scene from different perspectives. The main components of the ontology, including concepts, properties, and relations, are designed to take into account the requirements of automatic feature recognition from point clouds. More specifically, we have integrated formalized information on objects and their relations, which allows us to reason on both geometric and semantic features of objects at different levels of detail. Hence, the main contribution of this paper is automatic recognition of objects and their components based on reasoning on their geometric and semantic features that are formally represented and described in a knowledge base. In order to demonstrate the validity of the proposed approach, we present a case study for automatic recognition of semantic features of buildings from point clouds. For this purpose, prior knowledge of related concepts, their properties and relations, as well as a set of semantic rules, has been defined and included in a knowledge base and the reasoning results have been presented and discussed.

We expect that the approach proposed here in this paper can be extended to help the recognition of any type of object and its components in an urban scene. However, for the sake of simplicity and in order to show the potential of the proposed knowledge base, we have focused our experiments on the recognition of buildings and their components. In this case, we have a man-made object composed of simple planar segments where the extraction of properties and relations from point clouds are

relatively simpler. This is also true for the definition of the rules to support a reasoning process using the knowledge base.

The remainder of this paper is organized as follows: Section 2 investigates the existing knowledge-based methods for automatic 3D modeling and feature recognition and the solutions of using ontologies in practical applications. Section 3 presents the motivations for building a knowledge base for automatic 3D modeling. Then it describes in detail our proposed conceptual framework for this purpose and defines the scope of the proposed knowledge base and its content. Section 4 presents a case study for the evaluation of the proposed approach. Finally, Section 5 presents conclusions and perspectives for future works.

2. Related Works

Current approaches for 3D modeling from point clouds are mostly based on geometric approaches and are not sufficient to create complete and semantically enriched 3D urban scene models and virtual geographic environments from point clouds. An urban scene can be described using both quantitative and qualitative information on objects and their relations. Objects can be described by their geometric features (length, width, height, area, shape, boundary, etc.), and their geometric relations (parallel, perpendicular, coplanar), as well as their topological and logical relations. Any additional specification and constraints defining the properties and relations of an object are essential for efficient object recognition in the scene. Geometric features can be extracted from point clouds. However, semantic feature extraction is more complex and needs semantic reasoning on the entire knowledge including the extracted information from point clouds as well as the prior qualitative knowledge of the urban scene and its objects.

Knowledge of geometric relations can help 3D modeling and the extraction of semantic features of objects in an urban scene. For example, reasoning on geometric relations to determine the connections among the components of man-made objects is helpful for the creation of 3D geometric models from point clouds. There are two approaches to reason on geometric relations, deductive and algebraic reasoning [15]. For example, Loch-Dehbi et al. [15] introduced an algebraic method to demonstrate that constraints are deducible within sets of premises, aiming to support the interactive 3D city modeling and the automatic reconstruction of objects such as buildings and their components. The method is also capable of extracting geometric relations from uncertain observations. For the automatic feature extraction from a complex urban scene, the deductive and algebraic reasoning methods can be used to determine geometric relations between components extracted from point clouds. These relations can be represented as formal expressions such as “isParallelTo”, “isPerpendicularTo” in a knowledge base and used with other information for the extraction of semantic features on objects in subsequent steps.

Knowledge-based solutions have been proposed for the identification of the semantic meanings of objects from point clouds. Pu et al. [8] introduced a knowledge-based method for the reconstruction of buildings facades from terrestrial LiDAR data. In this study, information derived from point clouds such as size, position, orientation and topology is used to recognize building components, such as walls, doors, roofs, protrusions, intrusions, and windows. In addition, the fact that LiDAR cannot detect glass is used for the recognition of doors and windows in the point cloud. In other studies, knowledge-based approaches are proposed for the recognition of railway facilities from point clouds [16,17]. Further studies have attempted to label indoor components of buildings [18] and identify objects in kitchen environment [13] through learning algorithms. However, these methods are still very limited for the cases where we have many types of object to be detected and recognized at different levels of detail.

Prior knowledge of an urban scene must be formally defined and represented in a knowledge base. Ontologies have been employed for knowledge representation in many practical applications. For instance, ontologies describing railway facilities are used together with 3D modeling algorithms for processing point clouds to guide 3D object detection and labeling [16]. In [19], authors employ an ontology with a set of semantic rules to select algorithms and related parameters for detecting specific types of objects in a point cloud. In other fields, ontologies are used for better representation,

sharing and reuse of spatial data. An ontology is presented for searching most appropriate work automatically according to work conditions for avoiding subjective decision-making in the field of construction [20]. As another example, knowledge about safety management and construction risk is represented as an ontology for the development of a knowledge-based risk management system [21]. The ifcOWL is proposed for connecting semantic web technologies and the IFC standard in the construction industry [22,23]. For mobility requirements, ontologies are used to support indoor and outdoor navigation systems developments [24,25]. These examples show the potential of ontologies for better representation of knowledge in an urban scene in support of object recognition in a LiDAR point cloud.

In summary, for automatic recognition of semantic features of objects in support of the construction of 3D urban virtual geographic environments, an ontology of an urban scene is necessary to formally represent knowledge about objects in the scene. Semantic information on an urban scene can be formalized and represented in a knowledge base, and semantic rules can be added to allow reasoning on semantics of objects and their relations in support of 3D modeling and object recognition from point clouds.

3. Building a Knowledge Base for Automatic Feature Recognition

In philosophy, ontology is designed to explain the nature and relations of all beings, and it does not depend on a particular language [12]. In the domain of Artificial Intelligence (AI), ontology refers to knowledge representation, consisting of terminologies of a specific domain to describe certain realities that usually is conceptualized as concepts and their relationships [12,26]. Ontology is preponderant to represent and formalize the domain knowledge and experience for knowledge sharing and reuse [11]. If necessary, some rules are integrated to explain the activities of concepts. In the domain of 3D urban modeling, ontology is employed to formally represent knowledge about urban scenes, including concepts, names, properties, and their relations with other concepts. Then, semantic rules designed based on these concepts are integrated into the ontology to build a knowledge base for automatic feature recognition.

In general, ontologies are classified into top-level ontologies, domain ontologies and task ontologies, and application ontologies [12,27]. Top-level ontologies describe general concepts such as space, time, matter, object, event, action, etc. that are independent of a specific domain. Domain and task ontologies aim at generic domains and tasks. The related terms and vocabularies for generic domain, generic task or activity are defined. The semantic meanings of terms are additionally stated in certain domain and task, which is a process of specializing the terms in the top-level ontology [27]. Application ontologies represent concepts in a concrete domain, and the concepts are specialized continually in specific applications. They are designed for describing particular domain entities or a certain activity [12]. Therefore, choosing the appropriate level of ontology is important before developing an ontology because it indicates what concepts and relations should be considered to be included in the ontology and what would be their definitions and their specification.

An ontology is an abstraction of reality. The mismatch between an ontology and reality that it describes will appear if the concepts are not well specified. When specific vocabularies are used to explain the concepts in certain domains, the ontology closely depends on the language that is used. In other words, the expression scope, and the meaning of vocabularies and terms decide the accuracy of describing realities. Hence, minimal ontology commitment is an important criterion in developing an ontology [11]. Also, building an ontology does not aim to reason knowledge at the domain level. However, it attempts to help to understand the underlying knowledge with a computer-interpretable format in the practical applications. Thus, defining the ontology scope is a major step before developing an ontology. This allows the ontology to accurately represent concepts and their relations that are abstract of the physical objects and relations in a specific domain.

In this paper, we apply ontologies to represent the knowledge about objects in an urban scene and then to extract semantic features of objects by reasoning on the prior knowledge provided on a scene.

The knowledge of an urban scene may include different concepts and their properties from architectural domain (terms of architecture, building components and their relations), geometry (the definition of geometric primitives and their geometric relations), and their topological relations (topological relations among objects and among their components). The ontology designed for automatic feature recognition is positioned as an application ontology. The scope of the ontology is defined so that it can help the classification of object types, and the extraction of semantic features of objects and their components from point clouds of urban scenes. Based on the METHODOLOGY approach presented in [28,29], we used the following steps in the development of our ontology:

- (1) identification of motivating scenarios and the scope of ontology;
- (2) definition of competency questions;
- (3) building the ontology (ontology capture, ontology coding and integrating the existing ones);
- (4) validation of the ontology according to the requirements set by competency question;
- (5) maintenance of ontology after verification.

Based on these steps, our motivation for building the proposed ontology is to realize automatic feature recognition from point clouds. The ontology should represent the formalized knowledge of objects in an urban scene. Then we use semantic rules to reason on the knowledge provided on the urban scene to recognize semantic features of objects obtain from segmentation results. The validation of its scope will be conducted by the experiments for the extraction of semantic features of objects in a given urban scene. The maintenance of an ontology in its life cycle is an evolving process. The ontology needs to be maintained and updated continually after implementation based on the evaluation of ontologies [29]. In our application, the ontology supports the extraction of geometric features of objects at the first level and then it supports their recognition. For the implementation of this ontology, the knowledge acquisition is conducted from multisource. Hence, integrating existing ontologies is acceptable in the process of building ontologies [30]. Next, the expected achievements of the built ontology are represented as competency questions, such as the recognition of complex geometric shape based on planar segments and the identification of building roof shapes from point clouds. After building the ontology, it is integrated into a knowledge base together with a set of semantic rules. These rules are used to reason on the knowledge to answer the competency questions. This will help to validate if the ontology is competent to solve the problems mentioned in motivating scenarios.

3.1. Conceptual Framework for Automatic 3D Modeling and Feature Recognition from Point Clouds of Urban Scenes

In our proposed conceptual framework, the task of automatic 3D modeling and feature recognition from point clouds of urban scenes is divided into five main steps: object detection, object recognition (point clusters forming and object), segmentation, feature recognition and 3D model generation by connecting the components of objects (as shown in Figure 1).

The process of determining the range of the subset that belongs to a single object in the point cloud is usually called object detection [31,32]. The clustering algorithm uses Euclidian distance to cluster the points belonging to a single object [33].

In the object recognition, the object types are roughly classified according to the geometric properties of objects and by the reasoning on the geometric features of the concepts in the knowledge base. The purpose of object recognition is to select the segmentation algorithms to segment specific object types. In this step, the knowledge about different types of objects is provided by the knowledge base.

The aims of segmentation for a single object are to partition points into simpler groups, to decrease the search range, to reduce the computational cost and to simplify or alter the representation as segments that are more meaningful and easier to analyze [32,34]. Segmentation operation aggregates points with similar attributes or meaning into a single segment. Geometric features are usually used to segment point clouds to regular shapes. For instance, man-made objects in the urban area are

ontology for representing the knowledge of urban scenes. Reasoning on the knowledge provided in the knowledge base allows the extraction of semantic features to support objects recognition and 3D modeling process.

3.2. Definition of Concepts

Reasoning on objects embedded in an urban scene necessitates the extraction of quantitative (such as geometric dimension, coordinates) and qualitative properties (geometric shape, surface type, geometric relations, dependency, topologies, functions, surrounding attributes, etc.) from a point cloud and its integration as facts in the knowledge base. Facts on objects are obtained from segmentation operation. This operation can be conducted using region growing method based on robust normal estimation [36] and Random Sample Consensus (RANSAC) algorithms [33]. The semantic features of objects are expected as output. These facts are obtained based on the concepts, their properties, and their relations defined in the ontology. Formal representations of this information are crucial for the knowledge base. Because formalized representations of the knowledge are necessary to conduct semantic reasoning. Semantic reasoning uses facts and semantic rules to produce new knowledge of the object in the urban scene.

Hence, identification of different concepts, their properties, and their relations is fundamental for the building of an application ontology that will be used to support automatic 3D modeling and feature recognition in an urban scene. Table 1 presents different properties and relations that are included in the definition of different concepts in our ontology.

Concepts of an ontology describing an urban scene can be organized in a hierarchical manner using a graph structure. It is also possible to organize the concepts based on different views. A well-balanced ontological hierarchy gives a comprehensible representation of domain knowledge [37]. Some tips could be helpfully considered to formulate the balanced hierarchical conceptual tree. For example, concepts should be linked with a single relationship (is-a, is-part-of), the depth of the tree should be around equal, and cross-links should be as little as possible [38]. Therefore, concepts described by multi-dimensional information is an expressive way to describe urban scenes.

Table 1. Detailed quantitative and qualitative properties in the ontology.

Information Type	Terms	Examples
Quantitative Elements	Geometric Dimension	length, width, height, radius, thickness, area, volume
	Geographic Coordinate	latitude, longitude, elevation
	Local Coordinates	X, Y, Z
	Properties of Point Clouds	intensity, return number, point source ID, classification, color
Qualitative Elements	Object Types	building, car, road, tree, pole, etc.
	Geometric Shape	circle, rectangle, ellipsoidal, cross-sectional shape, line, cylinder, cuboid
	Surface Type	plane, curved surface
	Dependence	logical dependence, geographic dependence, physical dependence
	Topology	2D and 3D topology
	Function Relevance	interrelated relation for functions
	Surrounding Attributes	the neighboring information and their relations
	Architecture Components	wall, roof, floor, door, windows, balcony, etc.
Roof Shapes	flat, shed, gable, hip, barrel, etc.	
Material Attributes	concrete, wood, asphalt	
Geometric Relations	parallel, perpendicular, intersecting, coplanar, etc.	

3.3. Modularity of Concept in an Urban Scene

Modularity is an effective means to decrease the complexity in engineering, such as software development in software engineering. In the design of an ontology, modularity is a generic way to keep ontologies small to ensure reasoning performance and maintenance in knowledge management [39]. Concepts in an ontology can be categorized based on their types. In the lower level, an object is decomposed into its components. In the higher level, objects having similar function could be aggregated as a subsystem. In addition, the modules of spatial and topological relationships in ontologies are designed to represent the relations among objects and their components. Other modules,

such as functionality, attributes, constraints, relationship, and axiom, are defined to describe the concepts and their relations.

Identifying the concepts and the partitions of modules are the most significant steps in building an ontology. Firstly, the definition of concepts with understandable way is summarized from the real world in the urban scene. The quantitative and qualitative information that could be extracted from the segmentation results of point clouds is essential to describe objects in the ontology. The definitions of concepts should take this information into account. Besides, the relation module for describing the relevant relations (such as geometric, topological and logical relations) among concepts is defined. Finally, objects can be described by their topological relations, functionality, and semantic features.

In the following subsection, several modules are defined to organize concepts in the ontology, based on elevation, functionality, the source of objects, geometry, composition, and spatial relations.

3.3.1. Elevation Perspective

Coordinates are the most fundamental spatial information in point clouds that define objects shape and position. A cluster of points for an individual object forms a meaningful label. The core principle of clustering algorithms is to find a cluster based on a set of specific criteria. In point clouds, the closer points are more related to each other. Therefore, the spatial distance is a criterion to cluster points belonging to the same objects.

Considering elevation property of concepts given by Z coordinates, objects can be classified into ground, near-ground and non-ground categories following the generic category of objects in point clouds defined in [40]. For example, a road and a lawn belong to the ground class, curbs and small shrubs belong to the near-ground class, and buildings, trees, cars, and poles are categorized under the non-ground class. For defining building concept and its components, we benefit from concepts defined in the domain of architectural design [41]. Based on the classification of elevation, road curbs are closely associated with the road surface, and they can be used to determine the local width of a road. The lawn can be regarded as a part of the ground. The module of elevation is designed as follows in Figure 2.

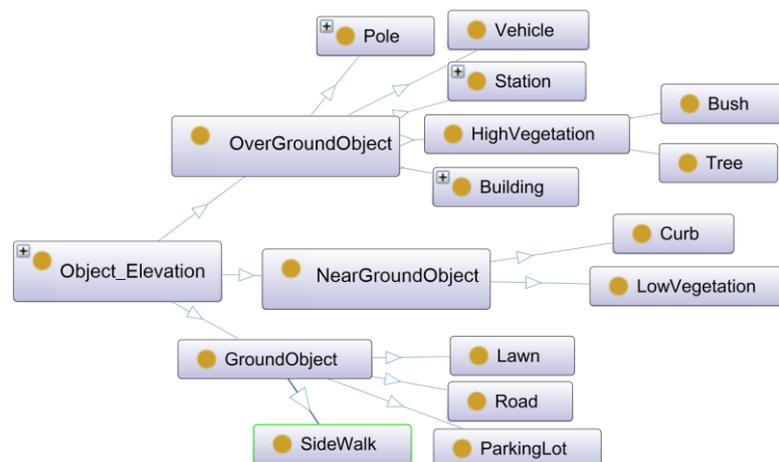


Figure 2. Classification of concepts according to the elevation perspective.

3.3.2. Functionality Perspective

Another perspective to modularize concepts in an ontology is based on objects' functionality and spatial proximity. For example, a building and a vehicle are all classified into non-ground objects based on elevation modularization. However, their functionalities are different as a building built for living and working while the vehicle is designed as a means for transportation purpose. From this point of view, objects in an urban area can be linked to the transportation system, and all function units

are connected to the road. For example, buildings are individual functional groups, such as business buildings, residential, or school buildings. However, a parking area consists of a parking lot, poles for paying the parking fee, sign poles and some possible parked vehicles. Transportation system contains roads and associative supporting facilities (such as traffic sign pole, light pole, traffic lights pole and bus station). Also, lawn, trees, and bushes are parts of the landscape. A public square, an open area at the meeting of two or more streets, is comprised of a part of the ground. There may contain some plants, bushes or statues in some case. Finally, the main concepts in the functionality module are shown in Figure 3.

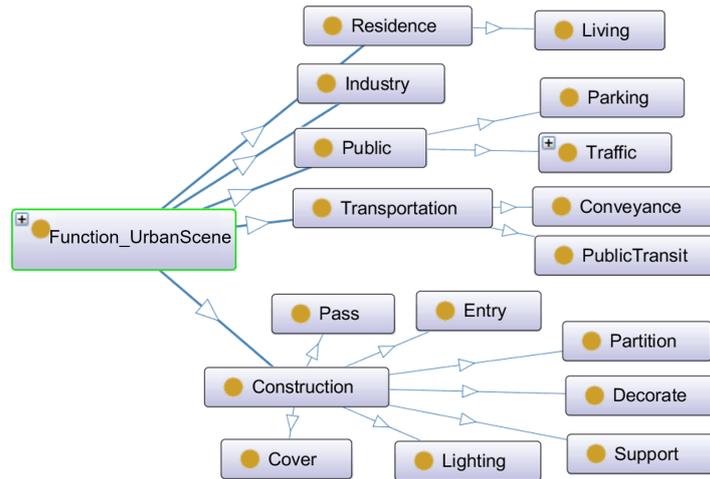


Figure 3. Classification of concepts according to functionality perspective.

3.3.3. Nature of Objects Perspective

Objects are either natural or artificial (man-made). From this perspective, trees, grasslands, bushes, etc. are classified as the plant in the class of natural objects. Roads, buildings, bridges, traffic poles, vehicles, etc. are placed in the class of manmade objects. What is more, a living organism lacking the power of locomotion is called as plant [42], which is a type of natural objects. The definition of the building is that a structure that has a roof and walls and stands more or less permanently in one place [42]. The building is also a kind of structure or construction. Thus, the module of source of objects can divide concepts into some small subsystems, such as structure, transportation, and plant (Figure 4).

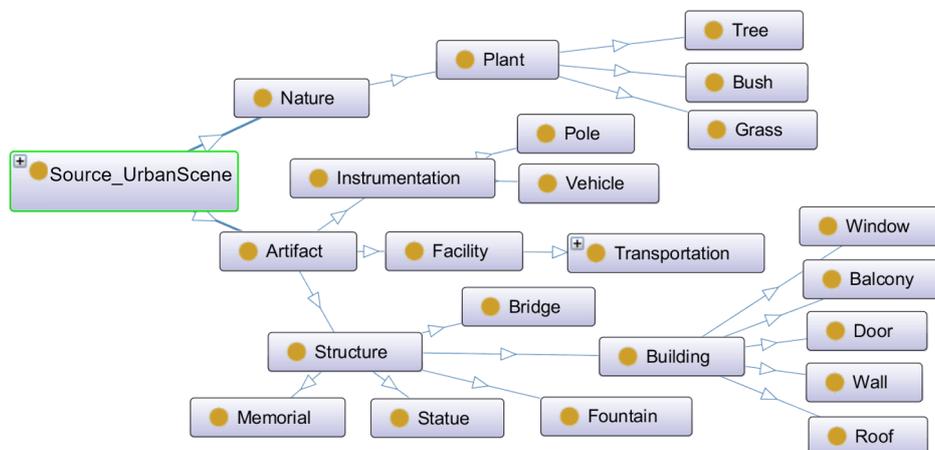


Figure 4. Classification of concepts according to their source or nature.

3.3.4. Geometry Module

Geometry is a branch of mathematics. Geometric information can be used to describe the spatial properties of objects such as length, area, and volume etc. Additionally, they determine the relative position of geometric shapes in the defined space. For example, the spatial relations can be inferred from existing geometry theorems. The geometric models offer fundamental geometric information to represent objects. For example, a building with simple shape can be modeled as a cube and a common wall is represented as a planar rectangle with its boundary points. Therefore, a geometric module is essential to an ontology to accomplish the task of extracting semantic features of objects from point clouds.

Geometric shapes can be divided into 0D, 1D, 2D and 3D geometric shapes. In 2D space, shapes are decided by their boundaries. In contrast, in 3D space, geometric shapes are not only determined by their boundaries, but also the types of geometry where they locate. Those shapes located in a plane in 3D space are defined by the parameters of the plane equation in 3D space and their boundaries. Those complex 3D geometries such as a polyhedron comprised of several planes can derive from basic planar geometries. For other geometries, such as sphere, cylinder and cone also need the parameters of equations and their boundaries to be defined. Finally, the concepts of geometries in 3D space are classified by their geometric properties following the geometry classes of ISO 19107 [43] (Figure 5).

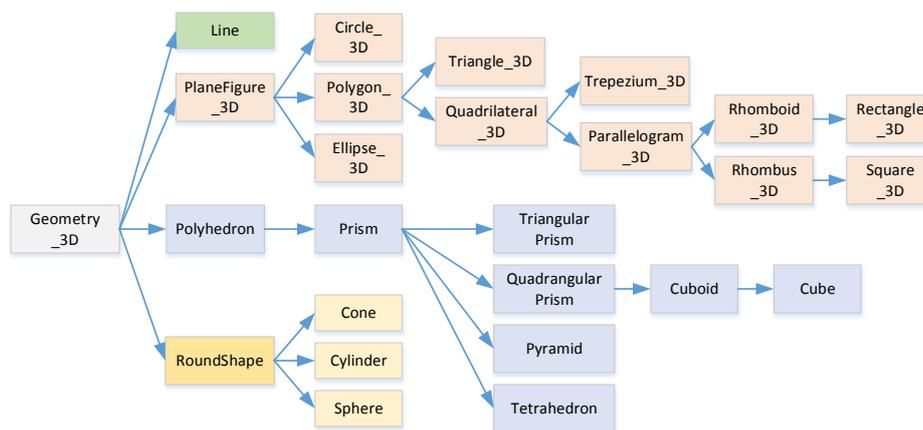


Figure 5. Classification of concepts related to geometry in 3D space.

3.3.5. Composition Module

The composition indicates the concepts of the aggregation of objects. Three levels of composition relationship for objects are: (1) aggregation of whole object by its components; (2) a subsystem combined with some objects; (3) a system comprised of several subsystems.

- **Components aggregation:** an individual object can be broken down into some components that cannot be decomposed into any small parts. For example, in geometric model of a building, the patch representing a wall may not be divided into smaller pieces.
- **Subsystem aggregation:** this relationship indicates the abstract concepts for representing functionally relevant sets. For example, a parking lot area comprises of a piece of ground with some vehicles, some sign poles and some poles for paying the parking fee.
- **System aggregation:** this level is used to represent the top-level aggregation relationships among objects in an independent scene or objects in a network. Examples include transportation system containing many parts severing for transportation.

In this module, the concepts in upper-level represent generic objects models with the relation of function-related aggregation. The lower-level aggregation forms the composition of the components of a single object.

3.3.6. Spatial Relations Module

Spatial relations involves topological, metric and directional relations, which are all capable of describing a scene with some semantic information [44]. Topological relations describe the relative relation of an object in the space with respect to other objects. This qualitative information plays a significant role in spatial analysis because it is independent of the coordinate system definition and transformations such as translation, rotation, and scaling [45]. In the following, the spatial relations in 2D and 3D spaces are described based on the concept of “region”. Based on the topological relations of planar regions in 3D space, the formalized representation of spatial relations in 3D space can be derived from point clouds with geometric information and semantic description.

To define topological relationships in a 2D space, we can use concept of “region”. A region is defined as a 2-cell object with a non-empty, connected interior in 2D space [46]. The region is applied to represent all kinds of 2D spatial objects because regions are the principal bearers of spatial properties and relations [47]. Thus, topological relations among spatial objects come from Region Connection Calculi (RCC-8) [48,49]. In the “4-Intersection” model (4IM) [48] and the “9-Intersection” model (9IM) [44,45,50], eight possible topological relations between regions are defined as disjoint, meet, overlap, cover, coveredBy, contain, containedBy and equal [51].

Topological relations between 3D spatial objects depend on the way objects are modeled. Constructive Solid Geometry (CSG) and Boundary Representation (B-Rep) are among models to represent spatial objects in 3D space. Thus, the topological relations between 3D spatial objects are classified into two categories: topological relations between 3D solid objects, and topological relations between 3D objects with internal space. The easily recognizable eight possible relations of 3D objects with inner space are Disjoint, Meet, Overlap, Equal, Contain, ContainedBy, Cover, CoveredBy [52]. However, the topological relations for 3D solid objects are only Disjoint and Meet relations. For determining the topological relations among 3D objects, RCC-3D [53] was designed for the spatial reasoning on 3D spatial objects based on the RCC-8 model. RCC-3D defines 13 relationships, but the discrimination of some relations require particular projection in the view of reference plane [54]. The RCC-3D relations can be used to describe occlusion between objects. However, it cannot be used to present topological relationships among components of an object to form a whole 3D model. In 3D B-Rep models, complex objects are composed of some components represented by geometric primitives with diverse properties, such as geometric shapes, size, and their topological relations [55]. For connecting these components to form a whole 3D model and extracting their semantic features based on components and their topological relations, the formalized representation of topological relations among components of objects is required.

For the topological relations of B-Rep objects and CSG objects, we can use the existing topological relations [52] in the ontology. The topological relations between object components need to be developed further in this ontology. The topological relations among object components are defined based on the concept “Planar Region”. A planar region in 3D space is defined as a planar area with non-empty, connected interior, which is fundamental to represent topological relations among objects components. In 3D space, the topological relations among the planar region are firstly decided by the spatial relations of two planes that contain planar regions. We can distinguish three cases:

- If the planes are parallel, these two planar regions are disjoint.
- If the planes are coplanar, the relation between two planar regions is determined as in 2D space.
- If the planes are intersecting, two planar regions can have many possible topological relations.

Therefore, the topological relations are classified into three classes: topological relations of B-Rep objects, topological relations of CSG objects and topological relations of planar regions in 3D space. In the class of topological relations between planar regions, there are three cases: planar regions on coplanar planes, parallel planes and intersecting planes (as shown in Figure 6). For each case, there could be several examples as described in [56].

Based on the above category of topological relation in 3D space, the topological relations need to be represented by a formalized representation to distinguish them. The topological relations in the intersecting case of two plane equations are disjoint, meet and intersect. The relation “Disjoint” is defined as there is no common part between two planar regions. The relation “Meet” indicates that there are common parts only located on the boundaries of the planar regions. The relation “Intersect” is the evolution of “Overlap” from RCC-8. There are also several cases of these three topological relations in 3D space. In the following, a formal representation of topological relations between two planar regions representing the components of more complex objects is developed. Based on the 4IM and 9IM topological relations definition, the topological relations among planar regions can be represented by a matrix consisting of boundaries, interiors and the intersection line of two plane equations. The definition of DE-9IM for planar regions [56] is shown as follows:

$$T_p(A, B) = \begin{bmatrix} \dim(A^\circ \cap B^\circ) & \dim(A^\circ \cap \partial B) & \dim(A^\circ \cap Il) \\ \dim(\partial A \cap B^\circ) & \dim(\partial A \cap \partial B) & \dim(\partial A \cap Il) \\ \dim(Il \cap B^\circ) & \dim(Il \cap \partial B) & \zeta \end{bmatrix}$$

where

- A° = indicates the interior of the region A ;
- ∂A = the boundary of the region A ;
- B° = the interior of the region B ;
- ∂B = the boundary of the region B ;
- Il = intersection of two planes containing planar regions A and B ;
- ζ records the topological relations of the primitives comprised of the common parts between the planar region A and B and intersection line. The primitives are all located on the intersection line;
- $\dim()$ = dimension operator.

Based on DE-9IM for planar regions, the details of topological relations between planar regions are decomposed into three parts (Figure 7):

- (1) The relation between the planar region A and the intersection line Il , including Disjoint, Meet and Overlap;
- (2) The relation between the planar region B and the intersection line Il ;
- (3) The relations between primitives on the intersection line Il that are the common part comprised of planar region A and the intersection line and the common part comprised of planar region B and the intersection line.

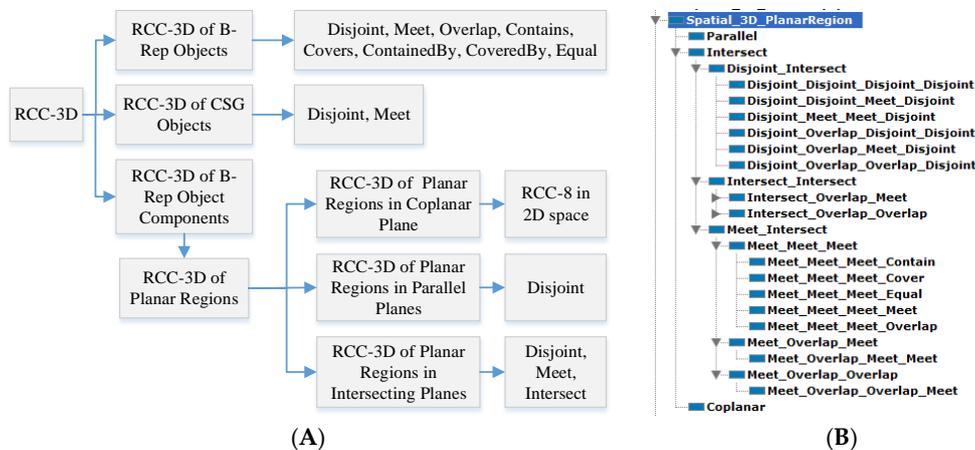


Figure 6. Classification of topological relations in 3D space (A), and their formalized representation in the ontology (B).

For the topological relations between a planar region and an intersection line, the possible relations are Disjoint, Meet and Overlap. The possible primitives on the intersection line comprised of the common parts of the planar region and the intersection line are point (for Meet relation between a planar region and the intersection line) and line segments (for an Overlap and a Meet relations between a planar region and the intersection line) (Table 2). The possible topological relations constitute point-point, point-line segment and line segment-line segment relations [56]. Finally, the formal representation of topological relations between planar regions is composed of four parts: (1) the overall topological relation of two planar regions; (2) the relation between planar region *A* and the intersection line; (3) the relation between planar region *B* and the intersection line; and (4) the topological relations of primitives on the intersection line. Some examples of topological relations in the case of Disjoint, Meet and Intersect under the “RCC-3D planar regions in Intersecting planes” can be found in [56].

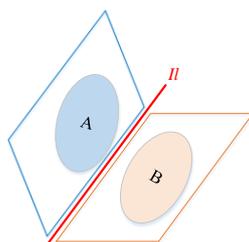


Figure 7. Illustration of an example of a topological relation between two planar regions (*A* and *B*) in 3D space.

Table 2. Basic topological relations between primitives on the intersection line [56].

Type of Relations	Graphical Representation	Topological Relations
Point-point relations		Disjoint, Equal
Line segment-point relations		Disjoint, Meet, Contain
Line segment-line segment relations		Disjoint, Meet, Overlap, Cover, Contain, Equal

In conclusion, the topological relations in 3D space are defined and formalized according to the way of representing 3D spatial objects. The category includes the topological relations of B-Rep objects and CSG objects, and the topological relations between planar regions representing the components of B-Rep objects. In the automatic 3D modeling of point clouds, the B-Rep models are employed to represent 3D objects and their components. Therefore, the topological relations among objects and that among the components of an object are all represented and discriminated by the formalized semantic representation of topological relations. Finally, based on these types of topological relations, the module of spatial relations in the ontology is created to represent the possible topological relations.

3.4. Objects Attributes

Attributes describe the features of objects, including original features such as the dimension of geometries (length, width, height, area and coordinate, etc.) and assign semantic information such as the label names of objects, the functions of objects. Attributes are important to describe objects.

The attributes can be classified from the views of attribute types and attribute modalities referring to classification in [57]. The attributes are classified into six types in the ontology (Table 3).

3.5. Constraints

Constraints limit the properties of an object to differentiate it from other objects. The purpose of constraints is to complete the specific tasks in aid of common sense knowledge and unique features in a certain case. The constraints can describe the knowledge about objects. The constraints are formalized as inferential and computer understandable first-logical-based rules. In summary, constraints are given from different aspects for the recognition of objects.

- Geometric dimensional constraints: for feature recognition, the essential and intrinsic attributes of objects, including measurable attributes, geometry shape attributes, limit the rough classification of objects.
- Spatial relations constraints: spatial constraints link objects in a local part of the urban scene. For some objects belonging to the transportation system, cars are moving on the road surface. Sidewalks are extending following the road or connected to roads. Traffic sign pole or light pole located near to the roads or sidewalks. Especially for man-made objects, components of objects have some topological relations constraints in the aspect of design or functional requirements. These constraints also can be represented as rules in the knowledge base.
- Logical constraints: some constraints are given not for the measurable or spatial constraints but from the view of logic. An example for interpreting logical constraints is that a parking lot is a piece of ground where accommodates a large amount of orderly arranged vehicles. Because logical constraints could associate concepts according to their logical relations of functions, locations, and system relevance, they are defined in the level of relevance among components of objects. Similarly, they can be defined in the level of subsystem consisting of objects.

Table 3. Classification of attributes in the ontology.

Attribute Types	Explanation	Examples
Dimensional attributes	measurable quantitative dimension of objects	size, height, length, width, area
Geometric shape attributes	describe geometric shapes	normal, boundary, surface type (plane, curved), shape(rectangle, square, circle)
Spatial attributes	location-related attributes and spatial relations	X-coordinate, Y-coordinate, Z-coordinate, latitude, longitude
Function attributes	object functions in a system or the roles of objects in a scene	lighting (for light pole), control traffic (for traffic sign), passing (for door)
Dependency attributes	attributes representing the interdependency between components or objects	logical dependency, geographic dependency, location dependency
System (combination) attributes	attributes are the terms for a group of objects or a subsystem	roof styles (such as gable, hip, shed, flat, and mansard and so on), traffic system, intersection

3.6. Relationships Definition

Relationships build the association among concepts. Relationships are of importance in the design of ontology due to their enriched definition and description of linking concepts. For the purpose of easy comprehension of relationships, they can be classified by their meaning.

- Hyponymy: it is the “is-a” relationship. It is the semantic relation of being subordinate or belonging to a lower rank or class [42]. Relationships including the definition of the kinds of concept constitute the backbone of ontological taxonomy tree structure. “is-a” relationship also contains some converted relationships, including synonymy and antonymy relations.

“isEquivalentTo” and “isSimilarTo” belong to synonymy relations. At the same time, “isDisjoint” and “isOpposite” are main relationships of antonymy [57].

- Meronymy: it is the “whole-part” relationship. It indicates the relationship of grouping concepts as a whole or decomposing concepts into parts. The relationships of “isPartof” and “isComposedof” are commonly defined in whole-part relations among concepts. In OWL ontologies, there are listed use cases of whole-part relations, such as defining “whole-part” relationships for individuals and class definition. Although the relationship “subclassOf” and “kind of” all are used to organize concepts hierarchically, their distinction must be made to decide the relationship in hierarchical concepts [58], including descriptive relations, possessive attributes (“has” relation), spatial relationship (locateAt, connect, align, parallel, vertical, direction, above, on, in), function relationship (hasFunction), and composition relations (must-beComposedOf, could-beComposedOf).

In summary, the relationships among concepts in the ontology need to be mapped into the relationship categories as mentioned above. In OWL ontology, “is-a” relationship is mapped as “subClassOf”. “whole-part” relationship is described in detail in accordance with the various cases [58]. Based on these relationships for building an ontology, some descriptive relationships are easily set among concepts. Inexplicit and indefinite relationships also can be identified and defined by property restrictions.

3.7. Axioms

The spatial and geometric relations are necessary for describing relationships between geometric primitives and obtaining accurate boundaries among primitives by interactions among geometries. Moreover, geometric relations can be obtained by reasoning using theorems on solid geometry. For example, if there are no common parts between two planes, then these two planes are parallel, or if two planes are all perpendicular to the same line, they are parallel as well. The theorems are easily predefined in ontologies as semantic rules. As a result, the spatial relations among geometric primitives are not only directly computed from geometric properties, but also from the reasoning on theorems, especially those describing complex geometric relationships.

Plane(?P1), Plane(?P2), Line(?L1), isPerpendicularTo(?P1,?L1), isPerpendicularTo(?P2,?L1) -> isParallel(?P1,?P2)

More constraints are needed for complex geometric theorems in 3D space. For example, in 2D space, if two lines are orthogonal to the same line, these two lines are parallel. The semantic rules are defined as follows:

Line(?L1), Line(?L2), Line(?L3), isPerpendicularTo(?L1,?L3), isPerpendicularTo(?L2,?L3) -> isParallel(?L1,?L2)

However, the above rule cannot hold if we replace the lines as planes for reasoning the parallel relationship between planes in 3D space. More constraints are required to reason spatial relation of planes. The following rule is designed for reasoning the parallel relationship between planes in 3D space.

Plane(?P1), Plane(?P2), Plane(?P3), Plane(?P4), isPerpendicularTo(?P1,?P3), isPerpendicularTo(?P1,?P4), isPerpendicularTo(?P3,?P4), isPerpendicularTo(?P2,?P3), isPerpendicularTo(?P2,?P4) -> isParallel(P1,?P2)

In conclusion, geometric relation axioms can be predefined as semantic rules for reasoning on the geometric relations between objects in 3D space. Based on these rules, new spatial relationships could be reasoned from the known fundamental relations among primitive geometric objects.

4. Experimentation and Results

The primary challenge for evaluation of an ontology is the terminology validation. The terms associate ontology with universal knowledge. Even in philosophical ontologies, the definition of terms

is a complex task. However, for our application ontology, the definitions of generic terms are not the primary requirements as the existing knowledge in the application domain is used in the ontology building process. Additionally, universality is not the objective as well. In general, it is difficult to achieve the consensus knowledge representation because the ontology is subjective by its nature [57]. For upper ontology, the definitions of terms and the determination of universal concepts could not be accepted in a short time. Consequently, the development of widely accepted ontology needs to be criticized and updated by researchers after their use over many years.

For evaluating an application ontology, some requirements of measures need to be established during the definition of the expectation of this ontology and finally to evaluate the corresponding achievements in the use. For application ontology aiming at object recognition in a point cloud of an urban scene, some competency questions are used to test its validity and its capacity to answer those questions. The competency questions are more specifically, including the recognition of geometry composed of planar segments and the recognition of building roof shapes from segmentation results of point clouds.

4.1. Consistency Check in Protégé

The ontology was built and represented by OWL in Protégé. In this software, concepts are represented as classes, and instances are described as individuals. Concepts, individuals and properties are defined using Description Logics (DLs). Additionally, several reasoners, such as Pellet, FACT++, Racer, can perform reasoning on concepts and semantic rules. Protégé can also help to evaluate the overall consistency of an ontology. Pellet reasoner provides functions for checking the consistency of ontologies, explaining inferences for reasoning results, and answering SPARQL queries [59].

4.2. Reasoning Experiments Based on Knowledge Base

As mentioned previously our ontology is designed and implemented in Protégé and Pellet is used as the reasoner on the knowledge base. The ontology is stored as owl files. Moreover, OWL API is a Java API for the operations of ontology, such as creating, manipulating and serializing OWL ontologies. Moreover, it provides OWLReasoner interface to access to the functionality of reasoning, such as consistency checking, computation of class and property and entailment of axioms [60]. More importantly, it is feasible to define semantic rules in Protégé. Then semantic rules can be used to reason knowledge in reasoner Pellet. In summary, automated reasoning based on predefined ontologies and semantic rules proves the feasibility of knowledge reasoning.

4.2.1. Experiment of Recognizing a Cuboid from Planar Regions

The first experiment shows a simple example for the extraction of an object from a point cloud. In this experiment, we show that if several planar regions are extracted from point clouds, the proposed ontology and rules are capable of recognizing a cuboid based on the geometric information extracted from the point cloud. For this purpose, we need the formation on plane-based prism and the topological relations among its components to recognize prism. For instance, for a cuboid, six planar regions can be segmented by geometric detection algorithm from point clouds. Their topological relations can be identified from the quantitative geometric information of planar segments using the proposed formalized topological relations. Each planar region will be added into "PlanarRegion" class as instances. In Figure 8, planar region *Pr1* and *Pr4* are opposite. Likewise, *Pr2* and *Pr5*, *Pr3* and *Pr6* are opposite as well. For manifold geometry, each edge is shared by two adjacent facets in a closed prism. According to the definition of recognizing cuboid formalized by semantic rules, the conclusion will be reasoned based on the known properties of instances in "PlanarRegion" class.

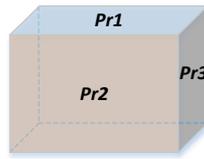


Figure 8. A simple cuboid example consisting of six planar regions.

To recognize prisms from a set of planar regions, the topological relations among regions are very important. A simple subset of concepts is extracted from the ontology in the knowledge base to reason on the cuboid with the help of topological relations among the planar regions and their boundary (Figure 9). In this figure, “PlanarRegion” class defines the concept of planar regions. “Cuboid” and “FacetofCuboid” classes are linked by the object property “isPartOf”. A cuboid concept is composed of several parts that are planar regions with special constraints and relations in 3D space. Thus, the instances of “PlanarRegion” class, such as *Pr1*, *Pr2*, *Pr3*, *Pr4*, *Pr5*, *Pr6*, are defined in the set “A” which is an instance of “Set” class. Moreover, the relations among the instances of “PlanarRegion” class are described, for example, all properties of *Pr1* are shown in Figure 10. Similarly, the properties of other instances are defined as well.

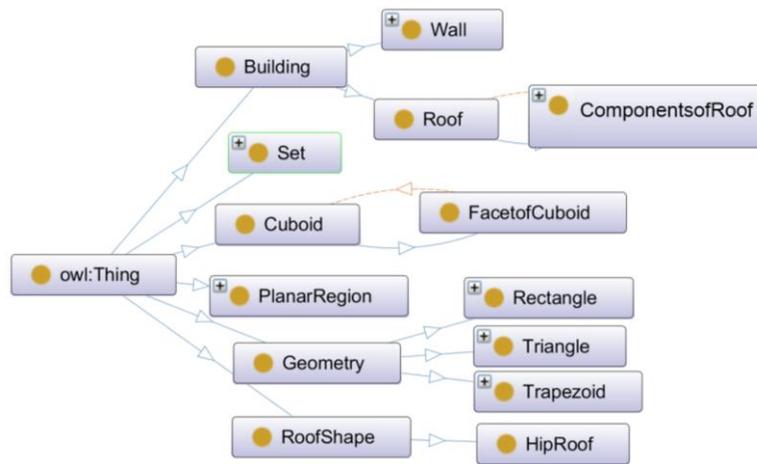


Figure 9. A subset of concepts for the recognition of a cuboid from planar regions.

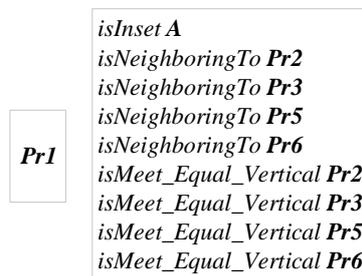


Figure 10. The relations and properties of the instance *Pr1*.

Based on these concepts and topological relations, semantic rules are defined for reasoning on the knowledge related to the specific object defined above. In the following rules, the relation “isMeet_Meet_Meet_Equal” is obtained from basic geometric information of planar regions following the definition of DE-9IM for planar regions and detailed steps presented in [56]. It indicates detailed formalized representation of predefined topological relations of the two planar regions in the spatial relations module. “Vertical” indicates another spatial relation between two planes *Pr1* and *Pr2*. In the

definition of rules, “isInSet (?x, ?A)” indicates that an individual x is in the set A. Based on the above definitions, the following semantic rules are used to reason on the knowledge for the extraction of a cuboid from planar regions, their properties and their relations.

$$\text{PlanarRegion(?Pr1), PlanarRegion(?Pr2), isNeighboringTo(?Pr1,?Pr2),} \\ \text{isMeet_Meet_Meet_Equal(?Pr1,?Pr2), isVerticalTo(?Pr1,?Pr2) -> isMeet_Equal_Vertical(?Pr1,?Pr2)} \quad (1)$$

The rule (1) is designed to test if the topological relation between two planar regions is “Meet_Meet_Meet_Equal” and their spatial relation is vertical to each other.

$$\text{PlanarRegion(?Pr1), PlanarRegion(?Pr2), PlanarRegion(?Pr3), PlanarRegion(?Pr4),} \\ \text{PlanarRegion(?Pr6), Rectangle(?Pr1), isNeighboringTo(?Pr2,?Pr1), isNeighboringTo(?Pr2,?Pr3),} \\ \text{isNeighboringTo(?Pr2,?Pr4), isNeighboringTo(?Pr2,?Pr5), isMeet_Equal_Vertical(?Pr2,?Pr1),} \\ \text{isMeet_Equal_Vertical(?Pr2,?Pr3), isMeet_Equal_Vertical(?Pr2,?Pr4),} \\ \text{isMeet_Equal_Vertical(?Pr2,?Pr6) -> FacetofCuboid(?Pr2)} \quad (2)$$

The rule (2) is defined to determine if a planar region belongs to a cuboid using the topological relations between it and its all neighbors.

$$\text{Set(?A), PlanarRegion(?Pr1), PlanarRegion(?Pr2), PlanarRegion(?Pr3), PlanarRegion(?Pr4),} \\ \text{PlanarRegion(?Pr5), PlanarRegion(?Pr6), isInSet(?Pr1,?A), isInSet(?Pr2,?A), isInSet(?Pr3,?A),} \\ \text{isInSet(?Pr4,?A), isInSet(?Pr5,?A), isInSet(?Pr6,?A), FacetofCuboid(?Pr1), FacetofCuboid(?Pr2),} \\ \text{FacetofCuboid(?Pr3), FacetofCuboid(?Pr4), FacetofCuboid(?Pr5), FacetofCuboid(?Pr6) -> Cuboid(?A)} \quad (3)$$

4.2.2. Axioms and Rules to Formally Define a Hip Roof from Planar Regions

The second experiment is used to recognize a hip roof style from a set of planar regions. The styles of roofs vary from regions to another region. Most common architectural roof styles can be identified and defined in the knowledge base. Here we choose the hip style as an example. A hip roof is defined as a type of roof where all sides slope downwards to the walls with a gentle slope [61]. All sides come together to form a ridge at the top of the roof. A typical hip roof is shown in Figure 11. In the hip style, there are two triangles and two trapezoids consisting of hip roof. They are individuals in the class “PlanarRegion” and they belong to “ComponentsofRoof”. According to the elevation information of planar regions and their relations with a wall, they can be defined as the components of a roof. The “ComponentsofRoof” class represents the concept that defines parts of the roof structure. The object property “isSloptTo” is defined to describe the slope of planar regions. It can be determined by computing the dihedral angle between two planes. For example, if the roof part *Pra1* is sloping to wall W1, the dihedral angle of them will be over 90 degrees. The individual *Pra1* represents the triangle 1 in Figure 11. Additionally, the “Tri” is an instance of the class “Triangle” which is a subclass of “Geometry” class. The properties and relations of the *Pra1* are presented in Figure 12. Other parts of the hip roof can be defined similarly by their properties and their respective relations. These properties and relations are then used to define semantic rules for the subsequent reasoning process.

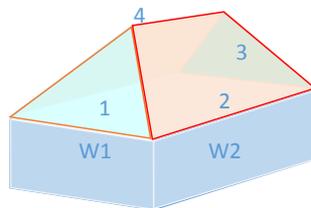


Figure 11. A building model with hip roof structure (the numbers 1 to 4 represent the individuals *Pra1*, *Pra2*, *Pra3* and *Pra4*. W1 and W2 are the individuals of class “Wall”).



Figure 12. The properties and relations of instance *Pra1*.

The following rule is used to reason on the provided knowledge for extraction of the hip roof.

Set(?B), Wall(?W1), Wall(?W2), Wall(?W3), Wall(?W4), Trapezoid(?Trap), Triangle(?Tri),
 PlanarRegion(?Pra1), isInSet(?Pra1,?B), ComponentsofRoof(?Pra1),
 PlanarRegion(?Pra2), isInSet(?Pra2,?B), ComponentsofRoof(?Pra2),
 PlanarRegion(?Pra3), isInSet(?Pra3,?B), ComponentsofRoof(?Pra3),
 PlanarRegion(?Pra4), isInSet(?Pra4,?B), ComponentsofRoof(?Pra4),
 hasShape(?Pra1,?Tri), hasShape(?Pra2,?Trap), hasShape(?Pra3,?Tri), hasShape(?Pra4,?Trap),
 isMeet_Meet_Meet_Equal(?Pra1,?Pra4), isMeet_Meet_Meet_Equal(?Pra1,?Pra2),
 isMeet_Meet_Meet_Equal(?Pra3,?Pra4), isMeet_Meet_Meet_Equal(?Pra3,?Pra2),
 isMeet_Meet_Meet_Equal(?Pra2,?Pra3), isMeet_Meet_Meet_Equal(?Pra2,?Pra1),
 isMeet_Meet_Meet_Equal(?Pra2,?Pra4), isMeet_Meet_Meet_Equal(?Pra4,?Pra3),
 isMeet_Meet_Meet_Equal(?Pra4,?Pra1), isMeet_Meet_Meet_Equal(?Pra4,?Pra2),
 isSlopeTo(?Pra1,?W1), isSlopeTo(?Pra2,?W2), isSlopeTo(?Pra3,?W3), isSlopeTo(?Pra4,?W4) ->
 HipRoof(?B)

In the rule (4), all the roof components, such as *Pra1*, *Pra2*, *Pra3* and *Pra4*, are the instances of the concept “PlanarRegion” and they are in the set B. In this set, if all the instances meet the defined constraints of geometric shapes and the topological relations among them in the rule, a hip roof can be reasoned from a set of planar regions.

4.2.3. Experiment for Recognizing a Hip Roof from Point Clouds

The third experiment presents the case of reasoning the higher levels of knowledge about building roof styles from point clouds based on our proposed knowledge base. After the segmentation of point clouds, a cluster of point clouds is segmented into seven planar segments (Figure 13A). The boundaries of these planar segments can be extracted from segmentation results as shown in Figure 13B. When having the boundaries of planar segments, the topological relations between planar segments are obtained. We employ the method of extracting topological relations between planar regions [56] to express the topological relations of two planar segments. The average distance between each boundary point and its k nearest neighbors are used to decide whether this boundary point should be projected onto the intersection line. If the distance between this boundary point and the intersection line is less than the calculated average distance with its k nearest neighbors, this boundary point is projected onto the intersection line (Figure 13C). The points projected on the intersection line form the primitives (point or line segment). The topological relations between primitives on the intersection line are important to determine the topological relations of planar regions. For example, the boundary points of two trapezoid planar segments are projected on the intersection line to form the ridge of the roof (Figure 13D). Here, we introduce several parameters as prior knowledge to the knowledge base that will help us in the computation of relations between planar regions. For instance, we choose 2 times of the average distance between points and their k -nearest neighbors ($k = 6$) as the threshold value to detect the points on the intersection line. We use this value to decide if the boundary points should be projected onto the intersection line and to determine the relations of the endpoints of line segments. As shown in Figure 13E, the distances between the endpoints of two line segments are 0.089 m and 0.53 m, which are smaller than the calculated thresholds. Thus, the topological relations between two planar regions with trapezoid shape are “Meet-Meet-Meet-Equal” after the comparison

of endpoints of two line segments. Similarly, the topological relations of other planar segments also can be obtained from point clouds.

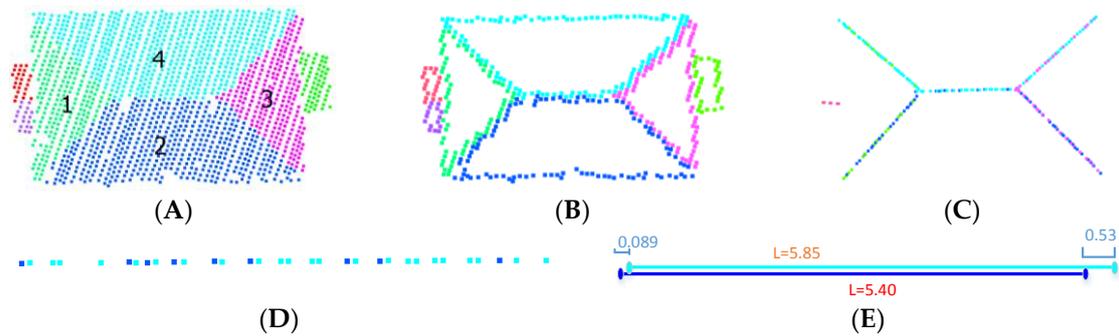


Figure 13. (A) Segmentation results; (B) boundaries and (C–E) the process of the determination of topological relations among components extracted from point cloud.

Based on the segmented planar segments and their dimension properties measured from the extracted planar regions, their spatial properties, and their topological relations, the planar segments are expressed as facts and as the instances of concepts defining a roof structure in the knowledge base. Then semantic rules as defined previously can be used to reason on this knowledge base. Because Airborne LiDAR scanner observes buildings roofs on the top, vertical structure of buildings are less present in the point clouds. We define rules to recognize roof shapes from Airborne LiDAR point clouds without the help of walls. For example, we define that the components of the hip roof have a slope to the ground to replace the properties defined with replace to the walls. The following rules are defined to reason the building roof styles from the segmentation results of point clouds.

```

Set(?B), Ground(?g), Trapezoid(?Trap), Triangle(?Tri),
PlanarRegion(?Pra1), isInSet(?Pra1,?B), ComponentsofRoof(?Pra1),
PlanarRegion(?Pra2), isInSet(?Pra2,?B), ComponentsofRoof(?Pra2),
PlanarRegion(?Pra3), isInSet(?Pra3,?B), ComponentsofRoof(?Pra3),
PlanarRegion(?Pra4), isInSet(?Pra4,?B), ComponentsofRoof(?Pra4),
hasShape(?Pra1,?Tri), hasShape(?Pra2,?Trap), hasShape(?Pra3,?Tri), hasShape(?Pra4,?Trap),
isMeet_Meet_Meet_Equal(?Pra1,?Pra4), isMeet_Meet_Meet_Equal(?Pra1,?Pra2),
isMeet_Meet_Meet_Equal(?Pra3,?Pra4), isMeet_Meet_Meet_Equal(?Pra3,?Pra2),
isMeet_Meet_Meet_Equal(?Pra2,?Pra3), isMeet_Meet_Meet_Equal(?Pra2,?Pra1),
isMeet_Meet_Meet_Equal(?Pra2,?Pra4), isMeet_Meet_Meet_Equal(?Pra4,?Pra3),
isMeet_Meet_Meet_Equal(?Pra4,?Pra1), isMeet_Meet_Meet_Equal(?Pra4,?Pra2),
isSlopeTo(?Pra1,?g), isSlopeTo(?Pra2,?g), isSlopeTo(?Pra3,?g), isSlopeTo(?Pra4,?g) -> HipRoof(?B)

```

In these experiments, first, we have tested the competency of recognized complex geometries such as a cuboid from planar regions. Then we have used the geometric properties and topological relations of planar regions representing the components of roof structures to recognize the types of roof shape. Finally, we have tested the capability of recognizing roof shapes from point clouds using the proposed knowledge base. The experiments showed that our proposed knowledge base represents and describes the knowledge of higher-level semantic features of objects. The automatic extraction of semantic features is achieved based on the knowledge of properties and relations of objects obtained from segmentation results as well as based on the knowledge in an urban scene.

In some cases, missing parts in point clouds can affect the correctness of feature recognition. However, depending on the presence of the missing parts in the data, our approach can go farther in the recognition of objects compared to more standard geometric algorithms. For instance, in the last experiment, if the missing parts do not impact the determination of geometric shapes and topological relations of the objects components, such as the missing parts locate in the interior and boundaries

of planar segment (as shown in Figure 14A,B), semantic features of building roof shape still can be obtained by the reasoning process on the available knowledge. However, if the missing parts limit significantly the available information on the object (Figure 14C,D), the reasoning results from the knowledge base would be uncertain or incomplete.

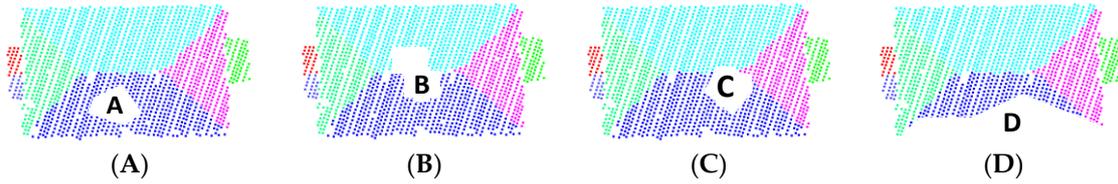


Figure 14. Impacts of missing data on feature recognition. (A) The missing part located in the interior of planar segment has no impacts; (B) The missing parts of the interior and boundary do not impact feature recognition; (C) The missing parts of the interior and boundary impacts the identification of topological relations; (D) A large area of missing part has impacts on the determination of geometric shapes and topological relations.

4.2.4. Experiment for Recognizing Semantic Features of Buildings from Point Clouds

In this experiment, we conduct an experiment on a part of building selected from a mobile LiDAR point cloud. First, we use a clustering algorithm to find clusters corresponding to buildings in the point cloud (Figure 15A). Second, the region growing algorithm is chosen to segment the cluster into planar segments based on robust normal estimation [36] which allow estimating surface normal from point cloud with noise (Figure 15B). Following a region growing process, RANSAC algorithm is used to detect different planes in the building structure (Figure 15C).

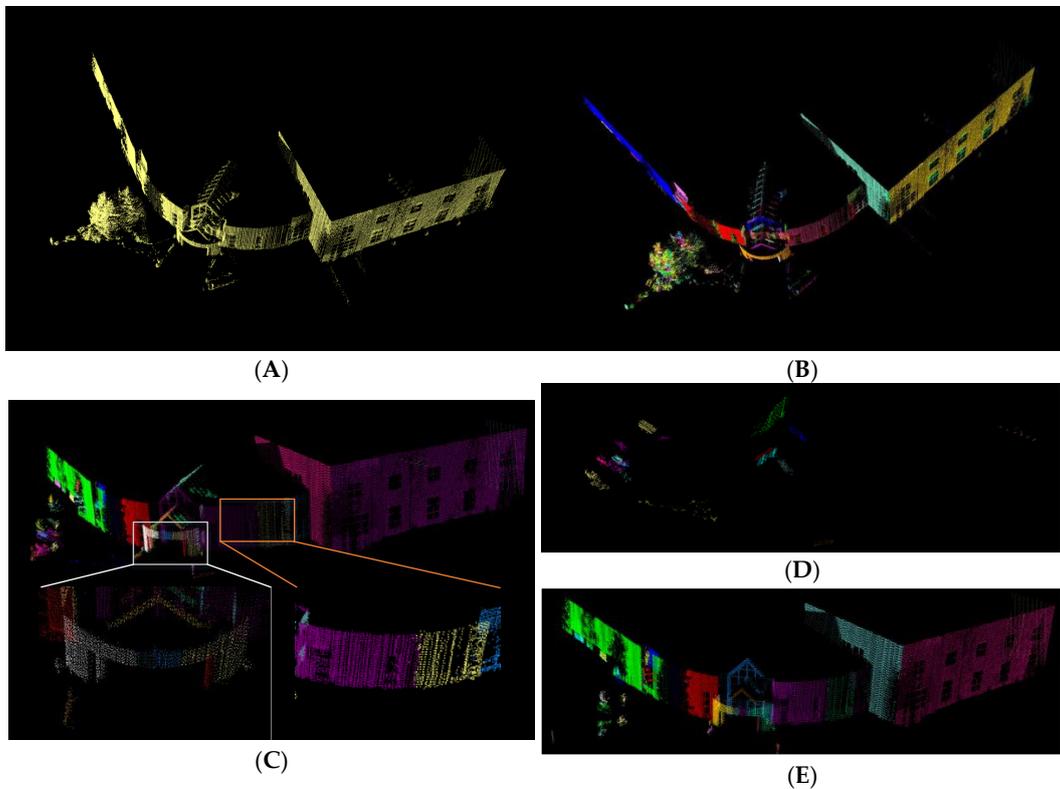


Figure 15. Recognition of semantic features of the components of a building from point cloud. (A) A cluster; (B) planar segments after region growing processing; (C) planes detected by RANSAC algorithm using plane models; (D) a potential roof structure; (E) a potential wall structure.

Now we need to recognize different components of the building. For this purpose we make use of the proposed knowledge base that describes different components of a building. For instance, a wall is defined as any opaque part of the external envelope of a building that makes an angle of 70° or more with respect to the ground [62]. A roof is considered to be locally the uppermost part of a building with a set of specific properties that allows distinguishing it from a wall. We use this knowledge to recognize a possible roof or wall structures extracted following from a point cloud. Figure 15D shows a part of the building that represents a potential roof structure as the planar segments in this case have an angle less than 70° with respect to the ground. In contrast, Figure 15E presents other set of planar segments that correspond to the above semantic definition a wall.

Based on the previous steps, we can preliminarily extract roofs and walls of the building. In addition to the building parts, there are some other small planes detected in the point cloud. These planar segments belong to a tree close to the building as shown in Figure 15D,E. The detailed knowledge about the building extracted from the point cloud is represented as facts in the knowledge base and specific rules are defined to help identify walls and roof.

Some rules are defined to recognize a wall and a roof and its shape following a reasoning process (Table 4). For recognizing a wall, we consider not only the constraints between planes and ground, but also the spatial relations among the planar segments that can be coplanar. The walls of the building are presented in Figure 16B. For a roof, the spatial relations between planes and ground, the areas of planes and the height information are considered. In addition, among the planar segments, there may be some that will not be recognized based on their own properties. In this case, these planes are analyzed within their spatial contexts. For instance, a small segment of a roof that is not recognized by its own, would be analyzed in its context. Thus, its semantic features will be obtained following the reasoning process with respect to the presence of other parts of the roof that are already recognized.

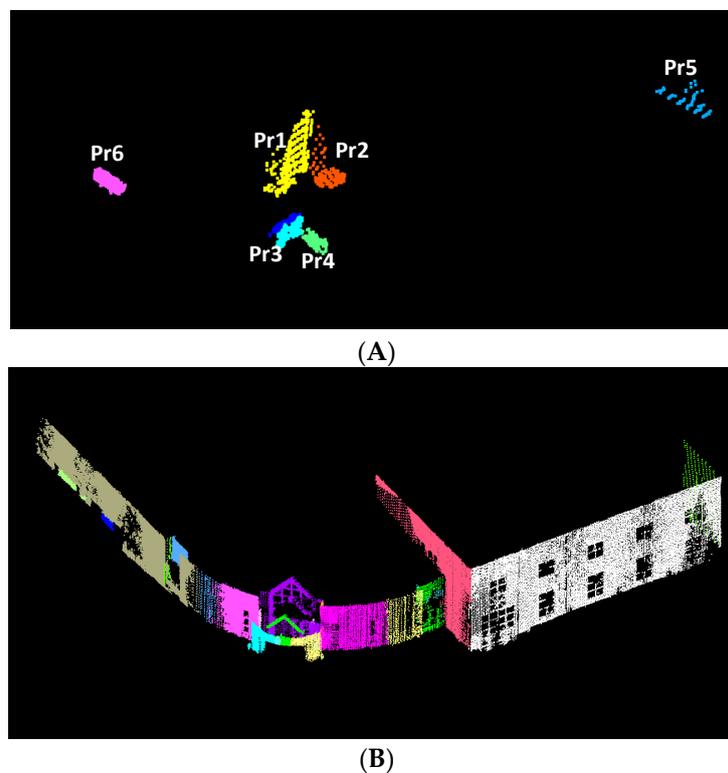


Figure 16. Recognition of roof and roof shape (A); and wall (B) from point cloud.

Table 4. Rules for the recognition of semantic features of buildings.

Semantic Features	Rules	Explanation	Rule ID
Wall	PlanarRegion(?pr_i), isVerticalTo(?pr_i,?ground), Ground(?ground), hasDirection(?ground,(0,0,1)), hasArea(?pr_i,?area_i), greaterThan(?area_i,2) -> Wall(?pr_i)	A wall is a plane that is vertical to ground and its area is greater than 2 m ²	(6)
	PlanarRegion(?pr_j), isCoplanarTo(?pr_j,?plane_i), Wall(?pr_j) -> Wall(?pr_j)	If a plane is coplanar to a wall, it is wall	(7)
	PlanarRegion(?pr_k), isConnectTo(?pr_k,?pr_i), Wall(?pr_i), isVerticalTo(?pr_k,?ground), Ground(?ground), hasDirection(?ground,(0,0,1)) -> Wall(?pr_k)	If a plane connects to a wall and is vertical to ground, it is wall	(8)
	PlanarRegion(?pr_j), Wall(?pr_i), isConnectTo(?pr_j,?pr_i), isCoplanarTo(?pr_j,?pr_i), -> isSameWall(?pr_j,?pr_i)	If a plane connects to a wall and is coplanar to this wall, they belong to same wall	(9)
Roof	PlanarRegion(?pr_i), hasArea(?pr_i,?area_i), greaterThan(?area_i,2), isSlopeTo(?pr_i,?ground), Ground(?ground), hasDirection(?ground,(0,0,1)), hasSlopeAngle(?pr_i,?ang_i), lessThan(?ang_i,70), hasHeightAttribute(?pr_i,?upperMost) -> ComponentsofRoof(?pr_i)	A roof component has covering function on the uppermost part of a building	(10)
	PlanarRegion(?pr_i), ComponentsofRoof(?pr_j), isSlopeTo(?pr_i,?ground), Ground(?ground), hasDirection(?ground,(0,0,1)), isConnectTo(?pr_i,?pr_j), hasSlopeAngle(?pr_i,?ang_i), lessThan(?ang_i,70), hasHeightAttribute(?pr_i,?upperMost) -> ComponentsofRoof(?pr_i)		(11)
Gable roof style	Set(?B), isInSet(?pr1,?B), isInSet(?pr2,?B), ComponentsofRoof(?pr1), ComponentsofRoof(?pr2), isMeet_Meet_Meet(?pr1,?pr2), Line(?line1) -> hasIntersectLine(?B,?line)	A gable roof consists of two roof sections sloping in opposite directions and the highest, horizontal edges meet to form the roof ridge. (v_g = (0,0,1))	(12)
	Set(?B), isInSet(?pr1,?B), isInSet(?pr2,?B), ComponentsofRoof(?pr1), ComponentsofRoof(?pr2), hasDirection(?pr1,?v1), isLeftSide(?v1,?v_g), hasDirection(?pr2,?v2), isRightSide(?v2,?v_g), Line(?line1), isParallelTo(?line1,?ground), Ground(?ground), hasDirection(?ground,?v_g), higherThan(?line1,?pr1), higherThan(?line1,?pr2) -> GableRoof(?B)		(13)

Based on these rules, the reasoning process allows the extraction of semantic features of building components. As we explained earlier, planar segments in point clouds are instances of concepts in the knowledge base. This implies that properties and relations of the instances should be extracted and formalized as facts in the knowledge base and are used in the reasoning process for the extraction of their semantic features.

For instance, the rule (13) in Table 4 represents a roof shape shown in Figure 16A. Here, the topological relation between *Pr1* and *Pr2* is “Meet_Meet_Meet_Cover”. Similarly, the relations between *Pr3* and *Pr4* is “Meet_Meet_Meet_Contain”. These two relations are sub-properties of “Meet_Meet_Meet” (see Section 3.3.6). Based on the topological relations among planar regions, the *Pr1* and *Pr2* constitute a gable roof because *Pr1* and *Pr2* are connected and they can be added into an instance of the concept “Set”. Then, the reasoning on these instances is executed automatically based on the rule (13) in Table 4, which results in the recognition of the semantic features of the roof of the

building. Similarly, a gable roof can be recognized from *Pr3* and *Pr4* as well, as presented in Figure 16A. Due to the absence of relevant context information, *Pr5* is recognized as roof as well. In fact, it is a part of ceiling in reality. In addition, some constraints for defining rules are defined by experiences. This is true also for miss recognition of a tree component as a part of wall.

In summary, the experiments presented in this section indicate that the designed ontology and the knowledge base are expressive enough and can represent well the knowledge related to different objects in an urban scene. As we can see from these experiments, the proposed knowledge base make not only use of higher-level generic knowledge of the concepts found in an urban scene but also it uses the facts on instances of those concepts obtained from segmentation and assessment of objects in a point cloud. Both of these sets of knowledge are used in the reasoning process for the extraction of semantic features of objects and their components presented in a given urban scene.

5. Conclusions and Future Work

In this paper, we have proposed a knowledge-based approach for automatic feature recognition from point clouds in support of the construction of urban virtual geographic environments. In the proposed approach, knowledge about objects in urban scenes is represented by ontology and semantic rules in a knowledge base. The ontology is built based on the steps presented in the METHODOLOGY approach [29]. Due to the advantages of modularity, several modules are defined to organize concepts in the ontology according to different perspectives, such as elevation, functions of objects, source of objects, geometry, composition and spatial relations. In addition, the properties, constraints, and the definitions of relationships among concepts are formally represented. Some theorems in geometry are also expressed as semantic rules for reasoning on spatial relations and other relevant knowledge for object recognition. In the spatial relation module, topological relations for 3D spatial objects are defined and formally represented. Moreover, the topological relations for 3D objects represented by B-Rep are added in the ontology. Based on the concepts, their properties and their relations, three experiments are conducted to test the competencies of the knowledge base for recognizing complex geometries, recognizing roof shape styles from the planar components of roofs, and recognizing the roof shape from point clouds. The designed experiments demonstrate that the proposed knowledge base can be served to reason on the different objects using the knowledge extracted from point cloud as well as the prior knowledge of the urban scene and its concepts and their relations.

Further investigations will be focused on the reasoning with uncertain and incomplete knowledge bases in the future. This will include fuzzy reasoning based on uncertain information on an urban scene due to missing data and uncertainties in the dataset.

Acknowledgments: The authors would like to thank the reviewers for their comments. This research is supported jointly by Natural Sciences and Engineering Research Council of Canada (NSERC) and the China Scholarship Council.

Author Contributions: Xu-Feng Xing conducted the research, performed the experiments and wrote the paper; Mir-Abolfazl Mostafavi gave instructions of organizing the sections during the writing and revised the paper, and Seyed Hossein Chavoshi has helped with revision of the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lin, H.; Chen, M.; Lu, G.; Zhu, Q.; Gong, J.; You, X.; Wen, Y.; Xu, B.; Hu, M. Virtual geographic environments (VGEs): A new generation of geographic analysis tool. *Earth-Sci. Rev.* **2013**, *126*, 74–84. [[CrossRef](#)]
2. Lin, H.; Chen, M.; Lu, G. Virtual geographic environment: A workspace for computer-aided geographic experiments. *Ann. Assoc. Am. Geogr.* **2013**, *103*, 465–482. [[CrossRef](#)]
3. Li, X.; Lv, Z.; Wang, W.; Wu, C.; Hu, J. Virtual reality gis and cloud service based traffic analysis platform. In Proceedings of the 23rd International Conference on Geoinformatics, Wuhan, China, 19–21 June 2015; pp. 1–6.

4. Xu, B.; Lin, H.; Gong, J.; Tang, S.; Hu, Y.; Nasser, I.A.; Jing, T. Integration of a computational grid and virtual geographic environment to facilitate air pollution simulation. *Comput. Geosci.* **2013**, *54*, 184–195. [[CrossRef](#)]
5. Xu, B.; Lin, H.; Chiu, L.; Hu, Y.; Zhu, J.; Hu, M.; Cui, W. Collaborative virtual geographic environments: A case study of air pollution simulation. *Inf. Sci.* **2011**, *181*, 2231–2246. [[CrossRef](#)]
6. Torrens, P.M. Slipstreaming human geosimulation in virtual geographic environments. *Ann. GIS* **2015**, *21*, 325–344. [[CrossRef](#)]
7. Mekni, M. Automated Generation of Geometrically-Precise and Semantically-Informed Virtual Geographic Environnements Populated with Spatially-Reasoning Agents. Ph.D. Thesis, Université Laval, Ville de Québec, QC, Canada, 2010.
8. Pu, S.; Vosselman, G. Knowledge based reconstruction of building models from terrestrial laser scanning data. *ISPRS J. Photogramm. Remote Sens.* **2009**, *64*, 575–584. [[CrossRef](#)]
9. Tang, P.; Huber, D.; Akinici, B.; Lipman, R.; Lytle, A. Automatic reconstruction of as-built building information models from laser-scanned point clouds: A review of related techniques. *Autom. Constr.* **2010**, *19*, 829–843. [[CrossRef](#)]
10. Guarino, N. Formal ontology, conceptual analysis and knowledge representation. *Int. J. Hum.-Comput. Stud.* **1995**, *43*, 625–640. [[CrossRef](#)]
11. Gruber, T.R. Toward principles for the design of ontologies used for knowledge sharing. *Int. J. Hum. Comput. Stud.* **1995**, *43*, 907–928. [[CrossRef](#)]
12. Guarino, N. Formal Ontology and Information Systems. In Proceedings of the First International Conference (FOIS'98), Trento, Italy, 6–8 June 1998; IOS Press: Trento, Italy, 1998; Volume 46, pp. 3–15.
13. Rusu, R.B.; Marton, Z.C.; Blodow, N.; Holzbach, A.; Beetz, M. Model-based and learned semantic object labeling in 3D point cloud maps of kitchen environments. In Proceedings of the IEEE/RISJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10–15 October 2009; pp. 3601–3608.
14. Cantzler, H. Improving Architectural 3D Reconstruction by Constrained Modelling. Ph.D. Thesis, University of Edinburgh, Edinburgh, UK, 2003.
15. Loch-Dehbi, S.; Plümer, L. Automatic reasoning for geometric constraints in 3D city models with uncertain observations. *ISPRS J. Photogramm. Remote Sens.* **2011**, *66*, 177–187. [[CrossRef](#)]
16. Hmida, H.B.; Cruz, C.; Boochs, F.; Nicolle, C. Knowledge base approach for 3D objects detection in point clouds using 3D processing and specialists knowledge. *Int. J. Adv. Intell. Syst.* **2012**, *5*, 1–14.
17. Truong, Q.H. Knowledge-Based 3D Point Clouds Processing. Ph.D. Thesis, Université de Bourgogne, Dijon, France, 2013.
18. Xiong, X.; Adan, A.; Akinici, B.; Huber, D. Automatic creation of semantically rich 3D building models from laser scanner data. *Autom. Constr.* **2013**, *31*, 325–337. [[CrossRef](#)]
19. Truong, H.Q.; Hmida, H.B.; Boochs, F.; Habed, A.; Cruz, C.; Voisin, Y.; Nicolle, C. Automatic detection and qualification of objects in point clouds using multi-layered semantics. *Photogramm. Fernerkund. Geoinform.* **2013**, *2013*, 221–237. [[CrossRef](#)]
20. Lee, S.; Kim, K.; Yu, J. Ontological inference of work item based on BIM data. *KSCE J. Civil Eng.* **2015**, *19*, 538–549. [[CrossRef](#)]
21. Zhong, B.; Li, Y. An ontological and semantic approach for the construction risk inferring and application. *J. Intell. Robot. Syst.* **2014**, *79*, 1–15. [[CrossRef](#)]
22. Pauwels, P.; Terkaj, W. Express to owl for construction industry: Towards a recommendable and usable ifcowl ontology. *Autom. Constr.* **2016**, *63*, 100–133. [[CrossRef](#)]
23. Pauwels, P.; Krijnen, T.; Terkaj, W.; Beetz, J. Enhancing the ifcowl ontology with an alternative representation for geometric data. *Autom. Constr.* **2017**, *80*, 77–94. [[CrossRef](#)]
24. Isikdag, U.; Zlatanova, S.; Underwood, J. A bim-oriented model for supporting indoor navigation requirements. *Comput. Environ. Urban Syst.* **2013**, *41*, 112–123. [[CrossRef](#)]
25. Shayeganfar, F.; Anjomshoaa, A.; Tjoa, A. A smart indoor navigation solution based on building information model and google android. In *Computers Helping People with Special Needs*; Miesenberger, K., Klaus, J., Zagler, W., Karshmer, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2008; Volume 5105, pp. 1050–1056.
26. Hadzic, M.; Chang, E.; Wongthongtham, P.; Dillon, T. *Ontology-Based Multi-Agent Systems*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 37–60.

27. Guarino, N. Semantic matching: Formal ontological distinctions for information organization, extraction, and integration. In *Information Extraction a Multidisciplinary Approach to an Emerging Information Technology, Proceedings of the International Summer School, Frascati, Italy, 14–18 July 1997*; Paziienza, M.T., Ed.; Springer: Berlin/Heidelberg, Germany, 1997; pp. 139–170.
28. Grüninger, M.; Fox, M.S. Methodology for the design and evaluation of ontologies. In *Workshop on Basic Ontological Issues in Knowledge Sharing*; IJCAI-95: Montreal, QC, Canada, 1995.
29. Fernández-López, M.; Gómez-Pérez, A.; Juristo, N. Methontology: From ontological art towards ontological engineering. In *Proceedings of the Symposium on Ontological Engineering of AAAI, Stanford, CA, USA, 24–26 March 1997*.
30. Fernández-López, M.; Gómez-Pérez, A. Overview and analysis of methodologies for building ontologies. *Knowl. Eng. Rev.* **2002**, *17*, 129–156. [[CrossRef](#)]
31. Wang, L.; Shi, J.; Song, G.; Shen, I.-F. Object detection combining recognition and segmentation. In *Computer Vision—ACCV 2007*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 189–199.
32. Dorninger, P.; Nothegger, C. 3D segmentation of unstructured point clouds for building modelling. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*; Vienna University of Technology: Vienna, Austria, 2007; Volume 35, pp. 191–196.
33. Rusu, R.B.; Cousins, S. 3D is here: Point cloud library (PCL). In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, 9–13 May 2011*; pp. 1–4.
34. Shapiro, L.G.; Stockman, G.C. *Computer Vision*; Prentice Hall: Upper Saddle River, NJ, USA, 2001.
35. Jochem, A.; Höfle, B.; Rutzinger, M.; Pfeifer, N. Automatic roof plane detection and analysis in airborne lidar point clouds for solar potential assessment. *Sensors* **2009**, *9*, 5241–5262. [[CrossRef](#)] [[PubMed](#)]
36. Nurunnabi, A.; West, G.; Belton, D. Outlier detection and robust normal-curvature estimation in mobile laser scanning 3D point cloud data. *Pattern Recognit.* **2015**, *48*, 1404–1419. [[CrossRef](#)]
37. Gavrilova, T.; Laird, D. Practical design of business enterprise ontologies. In *Industrial Applications of Semantic Web*; Bramer, M., Terziyan, V., Eds.; Springer: New York, NY, USA, 2005; Volume 188, pp. 65–81.
38. Gavrilova, T.; Carlucci, D.; Schiuma, G. Art of visual thinking for smart business education. In *Proceedings of the 8th International Forum on Knowledge Asset Dynamics (IFKAD-2013), Zagreb, Croatia, 12–14 June 2013*; pp. 12–14.
39. Stuckenschmidt, H.; Parent, C.; Spaccapietra, S. *Modular Ontologies: Concepts, Theories and Techniques for knowledge Modularization*; Springer: Berlin/Heidelberg, Germany, 2009; p. 378.
40. Pu, S.; Rutzinger, M.; Vosselman, G.; Oude Elberink, S. Recognizing basic structures from mobile laser scanning data for road inventory studies. *ISPRS J. Photogramm. Remote Sens.* **2011**, *66*, S28–S39. [[CrossRef](#)]
41. Hois, J.; Bhatt, M.; Kutz, O. Modular ontologies for architectural design. In *Formal Ontologies Meet Industry*; IOS Press: Amsterdam, The Netherlands, 2009; pp. 66–77.
42. Miller, G.; Fellbaum, C. *Wordnet: An Electronic Lexical Database*; MIT Press Cambridge: Cambridge, UK, 1998.
43. Kresse, W.; Fadaie, K. *ISO Standards for Geographic Information*; Springer: Berlin/Heidelberg, Germany, 2004; Volume 11, p. 322.
44. Mark, D.M.; Egenhofer, M.J. Modeling spatial relations between lines and regions: Combining formal mathematical models and human subjects testing. *Cartogr. Geogr. Inf. Syst.* **1994**, *21*, 195–212.
45. Egenhofer, M.J.; Herring, J. A mathematical framework for the definition of topological relationships. In *Proceedings of the Fourth International Symposium on Spatial Data Handling, Zurich, Switzerland, 23–27 July 1990*; pp. 803–813.
46. Egenhofer, M.J.; Herring, J. *Categorizing Binary Topological Relations between Regions, Lines, and Points in Geographic Databases*; Technical Report; University of Maine: Orono, ME, USA, 1990; pp. 94–122.
47. Roeper, P. Region-based topology. *J. Philos. Log.* **1997**, *26*, 251–309. [[CrossRef](#)]
48. Egenhofer, M. A formal definition of binary topological relationships. In *Foundations of Data Organization and Algorithms*; Litwin, W., Schek, H.-J., Eds.; Springer: Berlin Heidelberg, Germany, 1989; Volume 367, pp. 457–472.
49. Randell, D.A.; Cui, Z.; Cohn, A.G. A spatial logic based on regions and connection. In *Proceeding of the 3rd International Conference on Knowledge Representation and Reasoning, Morgan, Kaufmann, 25–29 October 1992*; pp. 1–12.
50. Clementini, E.; Sharma, J.; Egenhofer, M.J. Modelling topological spatial relations: Strategies for query processing. *Comput. Graph.* **1994**, *18*, 815–822. [[CrossRef](#)]

51. Egenhofer, M. Reasoning about binary topological relations. In *Advances in Spatial Databases*; Günther, O., Schek, H.-J., Eds.; Springer: Berlin/Heidelberg, Germany, 1991; Volume 525, pp. 141–160.
52. Zlatanova, S.; Rahman, A.A.; Shi, W. Topological models and frameworks for 3D spatial objects. *Comput. Geosci.* **2004**, *30*, 419–428. [[CrossRef](#)]
53. Albath, J.; Leopold, J.L.; Sabharwal, C.L.; Maglia, A.M. Rcc-3d: Qualitative spatial reasoning in 3d. In Proceedings of the 23rd International Conference on Computer Applications in Industry and Engineering (CAINE 2010), Las Vegas, NV, USA, 8–10 November 2010; pp. 74–79.
54. Sabharwal, C.L.; Leopold, J.L.; Eloë, N. A More Expressive 3D Region Connection Calculus. In Proceedings of the 2011 International Workshop on Visual Languages and Computing, Florence, Italy, 18–20 August 2011; pp. 307–311.
55. Freeman, J. The modelling of spatial relations. *Comput. Graph. Image Process.* **1975**, *4*, 156–171. [[CrossRef](#)]
56. Xing, X.F.; Mostafavia, M.A.; Wang, C. Extension of rcc topological relations for 3D complex objects components extracted from 3D lidar point clouds. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* **2016**, *XLI-B3*, 425–432. [[CrossRef](#)]
57. El-Diraby, T.E.; Osman, H. A domain ontology for construction concepts in urban infrastructure products. *Autom. Constr.* **2011**, *20*, 1120–1132. [[CrossRef](#)]
58. Natasha, N.; Evan, W. Simple Part-Whole Relations in OWL Ontologies. Available online: <http://www.w3.org/2001/sw/BestPractices/OEP/SimplePartWhole/#ref-FMA> (accessed on 23 November 2017).
59. Sirin, E.; Parsia, B.; Grau, B.C.; Kalyanpur, A.; Katz, Y. Pellet: A practical OWL-DL reasoner. *Web Semant. Sci. Serv. Agents World Wide Web* **2007**, *5*, 51–53. [[CrossRef](#)]
60. Horridge, M.; Bechhofer, S. The owl API: A java api for owl ontologies. *Semant. Web* **2011**, *2*, 11–21.
61. Curl, J.S. *A Dictionary of Architecture and Landscape Architecture*; Oxford University Press: Oxford, UK, 2006.
62. Design Buidling Ltd. Design Buidlings Wiki. Available online: https://www.designingbuildings.co.uk/wiki/Wall_types#Wall_definition (accessed on 23 November 2017).



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).