Automatic Detection and Classi cation of Objects in Point Clouds using multi-stage Semantics

Hung Truong, Helmi Ben Hmida, Frank Boochs, Adlane Habed, Christophe Cruz, Yvon Voisin, Christophe Nicolle

Automatic detection and qualification of objects in point clouds using multi-layered semantics

HUNG TRUONG[(1)(2)], HELMI BEN HMIDA[(1)(2)], FRANK BOOCHS[(2)],CHRISTOPHE CRUZ[(1)], ADLANE HABED[(1)], CHRISTOPHE NICOLLE[(1)], YVON VOISIN[(1)]

[(1)] LE2I – UNIVERSITY OF BURGUNDY, DIJON, FRANCE
[(2)] I3MAINZ – UNIVERSITY OF APPLIED SCIENCES, MAINZ, GERMANY

Keywords: 3D processing, point clouds, knowledge modelling, ontology, scene object qualification.

Summary: Due to the increasing availability of large unstructured point clouds from lasers canning and photogrammetry, there is a growing demand for automatic evaluation methods. Given the complexity of the underlying problems, several new methods resort to using semantic knowledge in particular for object detection and qualification support. In this paper, we present a novel approach which makes use of advanced algorithms, and benefits from intelligent knowledge management strategies for the processing of 3D point clouds and object qualification in a scanned scene. In particular, our method extends the use of semantic knowledge to all stages of the processing, including the guidance of the 3D processing algorithms. The complete solution consists of a multi-stage, iterative, concept based on three factors: the modeled knowledge, the package of algorithms, and the qualification engine.

Zusammenfassung:
Automatische Detektion und Qualifizierung von Objekten in Punktwolken unter Nutzung mehrschichtiger Semantik.
Infolge der zunehmenden Verfügbarkeit großer unstrukturierter Punktwolken aus Laserscanning und Photogrammetrie entsteht wachsender Bedarf für automatisierte Auswerteverfahren. Angesichts der häufig hohen Komplexität der in den Punktwolken enthaltenen Objekte stoßen rein Daten-getriebene Ansätze an ihre Grenzen. Es entstehen vermehrt Konzepte, die auf verschiedene Weise auch Gebrauch von Semantik machen. Semantik und Algorithmik sind dabei oft eng miteinander verwoben und führen zu Limitationen in Art und Umfang der nutzbaren Semantik. Mit der vorgestellten Lösung werden Algorithmik und Semantik klar getrennt und mit den exakt auf diese Domänen zugeschnittenen Werkzeugen behandelt. Deren prozedurale Verknüpfung führt dann zu einem neuen Verarbeitungskonzept, das eine bislang nicht erreichte Flexibilität und Vielseitigkeit in der Nutzung unterschiedlichster Semantiken besitzt und auch die Steuerung der Algorithmen integriert. Die iterative Gesamtlösung fußt auf drei Säulen, die aus dem modellierten Wissen, dem Pool der Algorithmen und dem Identifikationsprozess bestehen.

1 Introduction

Object detection, recognition and reconstruction from digitized data, typically images and point clouds, are important tasks that find applications in many fields. Because such processing tasks are extremely laborious and difficult when manually carried out, it is of the utmost importance that they are supported by – or even entirely performed through – numerical algorithms. Most existing algorithms are data driven and rely both on extracting discriminating features from the data set, and also on numerical models characterizing either geometric (e.g. flatness and roughness) or physical (e.g. colour and texture) properties of the sought objects. Regardless of how the numerical model and extracted features are combined to form a decision, such strategies are generally static in that they neither allow dynamic adjustments to the object description to be carried out, nor do they permit changes in the behaviour of the algorithms from one data set to the other. Instead, it is up to the user to decide, often subjectively but generally based on one's experience, which algorithms are better suited for any particular kind of objects and/or data sets. It goes without saying that the success of these approaches is significantly compromised by the increasing complexity of the objects and the decreasing quality of the data. Furthermore, relying on only a restricted set of features and individual algorithms to process the data might lead to unreliable results. One way to overcome the drawbacks of the data-driven approaches is to resort to the use of additional knowledge. For instance, knowledge characterizing the sought objects with respect to the data at hand or their relationships to other objects may generally be derived beforehand. Such knowledge not only allows for a systematic characterization and parameterization of the objects but also supports the quantification of the effectiveness of the algorithms to be used.

The work presented in this paper precisely aims at efficiently exploiting additional knowledge in the processing of point clouds. In particular, our work bridges between semantic modelling and numerical processing strategies in order to benefit from knowledge in any or all parts of an automatic processing chain. Our approach is based on structuring various knowledge components into ontologies containing a variety of elements taken from multiple sources such as digital maps and geographical information systems. However, not only do we rely on information about potentially present objects in the digitized scene (their characteristics, a hierarchal description of their sub-components, spatial relationships) but also on the characteristics of the processing algorithms at hand. During processing, the modelled knowledge guides the algorithms and supports both the analysis of the results and the object qualification. Knowledge is also used to support the choice among different algorithms, the combination of these, and the adopted strategies.

Our paper is structured as follows. An overview of the relevant literature on the topic is presented in Section 2. Our proposed solution is outlined in Section 3. Knowledge building and management are discussed in Section 4. Section 5 is dedicated to our knowledge-based strategy for object detection and qualification. This is followed by a case-study involving a real world example in Section 6. Our conclusion and future work are given in Section 7.

2 State of the art

Early 3D processing techniques were purely data-driven exhibiting obvious limitations with the increasing complexity of the data and scene. Progress has been achieved by considering the use of models approximating geometrical characterisitics of objects. However, despite the robustness and efficiency of many such processing algorithms, they alone cannot resolve existing ambiguities when qualifying objects in a digitized scene. Such ambiguities can be efficiently dealt with when integrating semantic knowledge with data processing.

As far as feature-based object recognition is concerned, some of the same approaches have been used in both 2D images and 3D data. For instance, VOSSELMAN & DIJKMAN, 2001 made use of higher level 3D features, usually simple roof shapes (flat roofs, gable roofs and hip roofs) that are generally present in building structures. The authors relied on the use of the 3D Hough transform to detect planar roof faces in point clouds, and hence reconstructed the scene in a higher level of abstraction. Their segmentation strategy was based on detecting intersecting lines and “height jump edges” (differences in orientation) between planar faces. PU & VOSSELMAN, 2006 have used segmentation and feature extraction algorithms to recognize building components (such as doors, walls, windows) from point clouds. Based on constraints on the sought components, they were able to determine the categories each extracted feature belonged to. However, the results were not satisfying when the data did not clearly describe the object whether in the presence of noise or because of occlusions.

An important processing aspect which partly solves the limitation above - particularly when dealing with data-driven approaches using artificial intelligence - has to do with enforcing the robustness of such methods to recognise the complex objects. A typical work in this category is the one presented by ANGUELOV, et al.2005 in which object segmentation and classification are obtained through a learning procedure employing Markov Random Fields and quadratic programming. Another method worth mentioning in the same category is the one proposed by TRIEBEL, et al.2007 and which allows the classification of more complex objects based on a diverse set of features incorporated within the framework of associative Markov networks for training. Such methods, however, generally require a large number of training data sets in order to obtain good results.

Building on the above results, significant improvements have been brought to the processing of 3D data through additionally incorporating semantic aspects. In the automatic data extraction process from 3D point clouds used by DUAN, et al. 2010, the results are a digital model of the surveyed scene.The method proposed by CANTZLER, et al. 2002 relies on a semantic network defining the relationships among objects in a scene (such as walls being perpendicular to the floor) and rules which the extracted features must obey. Afterwards, the interesting issues come with complex indoor scenes, including many types of object. Semantic knowledge is defined in terms of geometric constraints (NÜCHTER, et al. 2006). This has turned out to be very useful in building indoor 3D maps through classifying groups of points into floors, ceilings and various other objects. The ability to exploit semantic knowledge is limited when the number of objects becomes large, requiring an adequate way for structuring properties of and relationships between objects.

Some other approaches in their solution aim to describe hierarchically the attributes of an object (or a scene). TEBOUL, et al. 2010 segmented building facades using a tree to interpret procedural geometry, and connected grammar semantics and images using machine learning. This approach proposed a dynamic way to perform searches through a perturbation model. RIPPERDA, et al. 2006 also extracted building facades using structural description, and used Monte Carlo Markov Chains to guide the application of derivation steps during the building of the tree. Another application of using knowledge is to infer the missing parts from detection. For example, PU & VOSSELMAN, 2009 reconstructed building facades from terrestrial laser scanning data. Knowledge about size, position, orientation and topology is used to recognize features (e.g. walls, doors and windows), and also to hypothesise the occluded parts. In a similar work (SCHOLZE, et al. 2002), a model-based reconstruction method has been proposed. In this method, semantic knowledge is also used to infer missing parts of the roof and to adjust the overall roof topology. These approaches use knowledge to evaluate results from numerical processes, but do not integrate it into the processing as such.

Since the use of knowledge within processing is also useful, other research has focused more on knowledge management within computation. For example, MAILLOT, et al. 2008 used a visual concept ontology composed of visible features (such as spatial and relations, color and texture) to recognise objects through matching among nummerical features and visual concepts. DURAND, et al. 2007 proposed a recognition method based on an ontology which has been developed by experts of the domain; the authors have also developed a matching process between objects and the concepts of the ontology to provide objects with a semantic meaning. However, knowledge in these approaches has not been fully exploited; other capabilities, such as processing guidance, have not been explored.

This previous research shows that there are various attempts to make the analysis of huge point clouds more robust and efficient. This needs structured processing going from prominent to less characteristic features, and to build a bridge between the objects and their expected geometry. Simple models are efficient and robust, but have limitations for more complex objects. Statistical methods are able to handle more complexity, but they also need large training efforts and are difficult to transfer. Knowledge based methods, however, seem to have the potential to manage even more complex scenarios. Succesful work uses geometric and/or topological relations of objects for their identification, or tries to map the structure of a scene into a semantic framework. Other work introduces knowledge into the processing and allows the use of various characteritics of objects in order to improve their detection.

Various kinds of knowledg-based approaches appear in application development, documenting an increasing interest for semantic approaches. This expresses a certain expectation about the role of semantics in future solutions. What is still missing is an overall approach for knowledge integration, which would guide the numerical processing, the evaluation, and qualification of found objects.

3 System overview

When trying to build an integrated approach with knowledge directing all parts of the process, several aspects have to be considered. At first, the whole process needs to be incorporated into knowledge management tools. Therefore it is necessary to have a process guiding all individual steps, leading from an initial situation to the final result. Inside this overall process, one part has to cover the numerical processing and another part has to handle the processing results. This latter part has to evaluate the results, conclude what has been found, and also what this means for the processing. This includes the need to update the content of the data base with the objects found. This database has to be managed, and transfers all found objects from the initial state to the final state.

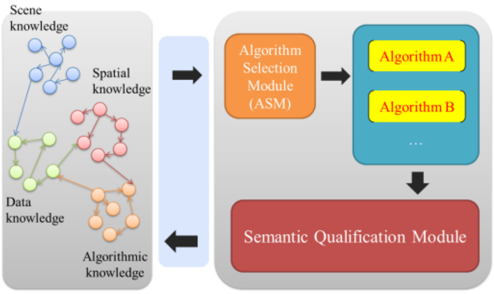

Fig. 1: System architecture

Such a strategy is contained in the framework proposed here. Fig. 1 illustrates this strategy, which is applied to the analysis of 3D point clouds, but can also be extended to other useful data sources. It is based on explicitly formulised prior knowledge to the scene, on spatial relations of objects and on processing algorithms. It's a multi-stage concept based on three supports: the modelled knowledge (Fig 1 left side), the package of algorithms (Fig 1 right side above) and the qualification engine (Fig 1 right side below). In the initial stage, the accessible knowledge is transferred into a corresponding knowledge base. Depending on the particularity of the prior knowledge, this base might be simply generic (if no real object exists in the scene already) or it might be more concrete because of already addressed objects which were contained in the scene. Starting from this initial stage, an update process begins, which invokes the algorithms and the qualification engine. Herein the algorithmic selection module (ASM) guides the processing via selecting a set of processing algorithms based on the nature of target objects, and produces new elements which can be identified. These elements are passed to the qualification engine, which then tries, based on the existing knowledge expressed in the ontology, SWRL rules and DLs constraints, in order to identify the nature or object category of the elements. This qualification handles the output from the algorithms. The result of the qualification step will update the knowledge base by entering newly qualified or updating already existing elements, and then entering the next stage of processing. As soon as no further refinement of the base is achieved, the process ends.
Objects are represented by a point cloud or other data sources, depending on many factors such as the type of the sensing system and the measuring situation. This representation has to be handled by algorithms, which also depend on many additional factors (e.g. noise, other data characteristics, and already existing objects). Strong interrelationships among these factors influence the detection and qualification process. The more that the factors and the interaction are flexibly controlled, the better the expected results. For these reasons, a variety of knowledge from different domains is required, where the quality of these knowledge sets has significant impacts on the results (BEN HMIDA, et al. 2011).The defined solution relies on four main knowledge categories, cooperating together to construct the core of the knowledge base: the Scene Knowledge (SK), the Spatial Knowledge (SpK), the Data Knowledge (DK) and the 3D Algorithmic Knowledge (AK). Each field of knowledge is represented by circles in the above figure, and relations between these concepts are represented by edges. The scene knowledge contains information related to the content of the scene to be processed, important characteristics of objects (e.g. geometric features, appearance, and texture), and the geometry that composes its structure. Such knowledges are not only important for processing the identification and qualification activities, but will also support the selection and guidance of the algorithmic processing. The spatial knowledge models the relationships among objects in the scene. It presents a main key for the qualification process, since it yields to the objects state disambiguation based on its relation with the common environment. The data knowledge expresses important characteristics of the data itself. Finally, algorithmic knowledge characterises the behaviour of algorithms and determines what kind of purpose they fulfil, which input is expected, what output is generated, and to which geometries they are designed for. Based on this knowledge, a dynamic

algorithm selection is possible, and allows dynamic adaption for processing situations given from other domains (Fig. 1).

4 Building knowledge

Basically the concept requires efficient methods of knowledge handling and interaction with algorithms. Efficient knowledge-handling tools are available from the Semantic Web framework, which expresses knowledge through the Web Ontology Language (OWL) (BECHHOFER, et al. 2004). The encapsulation of semantics within OWL through Description Logics (DLs) axioms has made it an ideal technology for defining knowledge from almost any discipline. We use the OWL to define expert knowledge about the scene of interest and for the algorithmic processing. With OWL ontology, we are able to describe complex semantics of a scene. For instance, the statement "A railway track is a linear feature with two linear structures running parallel to each other within a certain distance" can be expressed through logical statements. Likewise, we are defining the semantics of algorithmic processing within OWL. For example, the "Check parallel lines" algorithm is designed for detecting a "Signal," which may contain parallel linear structures.

$$Check\_parallel\_lines \exists\ isDesignedFor.Signal \sqcap Signal.hasParallel.\{true\} \tag{1}$$

As additional technology, the Semantic Web Rule Language (SWRL) (HORROCKS, et al. 2004) is available. It is a program which infers logic from the knowledge base to derive a conclusion based on the observations and hypothesis. For instance, the following rule (eq. 2) asserts that a detected element (expressed by a *GeneralizedGeometry*) which has a distance from *DistanceSignal* of 1000m, has a height equal to or greater than 4m, and which has a linear struture, will be inferred as a $MainSignal$.

GeneralizedGeometry(?x) ^ hasLine(?x, ?l) ^ line(?l) ^ DistanceSignal (?y) ^ DistanceFrom(?x, ?y, ?dis) ^ swrlb:GreaterThan (?dis,1000) ^ hasHeight(?x, ?h) ^ swrlb:GreaterThan (?h, 4) → MainSignal(?x) (2)

Most importantly, SWRL built-ins are keys for any external integration. They help in the interoperation of SWRL with other formalisms and provide an extensible infrastructure for knowledge based applications. They are essential in that they allow entrance to a different world of processing. In the context of this solution, it bridges knowledge management and algorithms, either for 3D processing or topological considerations.

The techniques mentioned above serve as tools to formalize the identified and acquired knowledge. As explained, the actual solution handles four separate domains: the Scene knowledge (SK), the Spatial Knowledge (SpK), the Data Knowledge (DK) and finally the Algorithmic knowledge (AK). All these knowledge domains have their representations in the domain ontology and participate in the whole processing cycle. The graphical structure of the top level concepts of the ontology is given in Fig. 2, where we find six main concepts, called "*Classes*" in the next paragraphs.

In order to proceed, these classes have to resume the different actors used during the detection and the qualification process in a structured hierarchical way. The main actors that have to be modelled are: processing algorithms, point cloud data or images resources, and target objects with their geometry and characteristics. The class DomainConcept represents the different objects found in the target scene and can be considered the main class in this ontology. This class is further specialized into classes representing the different detected objects. The other classes are used to either describe the object geometry through the Geometry class by defining its geometric component (or the Generalized Geometry object is used to indicate its coordinates), or to describe its characteristics through the Characteristics class. Ultimately, the algorithms are recommended based on their compatibility with the object geometry and characteristics via the Algorithm class.

Knowledge of different domains is aquired from the relevent sources. Sources such as domain experts are the most reliable knowledge source. However, other information sources such as CAD, GIS data, or other available documents in the case of detailed input are used to extract knowledge. In our case the algorithmic knowledge is acquired from experts in numerical processing, and scene knowledge is acquired through the existing digital documents as a CAD drawing or GIS dataset.

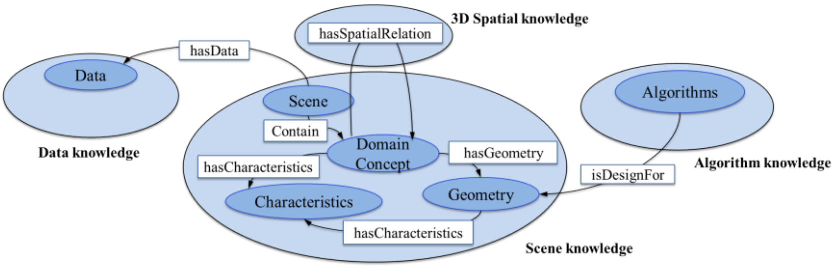

Fig. 2: General ontology schema overview.

The Scene Knowledge is described in the schema of ontology and includes semantics of the objects, such as properties, restrictions, relationships between objects and geometries. The more information about an object is created and used, the more accurate the detection and qualification process. An example of defining a semantic object is the following: an electric pole in a railroad has height between 4m to 6m; it is constructed by a vertical structure that connects to a cube on the ground; at the top, there are two parallel linear structures; and along the track, the distance from an electric pole to a signal column is 1000m.

3D Spatial relation knowledge is used to enhance the qualification process. Information about how objects are dispersed in a 3D scene makes the detection and qualification easier. For instance, given the detection of a wall, the chances increase that a door or window will be detected within it. In fact, 3D spatial knowledge includes standards like the 3D topologic knowledge, 3D metric knowledge and 3D processing knowledge. Each one of the cited spatial knowledge contains a variety of relations modelled on the ontology structure. For example, the top level ontology is designed to include the topological relationships. This is then used to enrich an existing knowledge base to make it possible to define topological relationships between objects in a specific case. Metric knowledge presents important information, since the different elements are respecting very strict metric rules which can also be used for the detection and qualification process. In the example of the rail road, Fig. 3 shows an ontological structure, supported by the SWRL rules, which can automatically specify that an object (with certain characteristics) that has a distance of 1000m from *Distance Signal* should be a *Main Signal*. Because of outside factors such as data noise and the uncertainty of the measurement, the knowledge allows tolerances (±0.5m for example, depending on the quality of data).

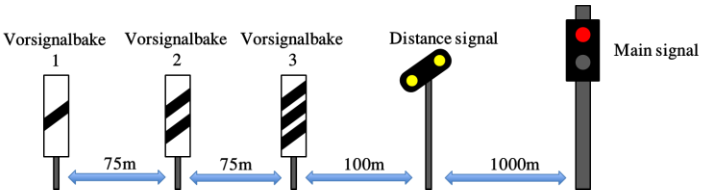

Fig. 3: Metric rules.

Regarding the numerical processing algorithm, effectiveness depends on the quality of the data (resolution, noise), the characteristics of the object that needs to be detected, or other factors depending on a specific case. Algorithms are modelled under specialized classes of algorithms, sharing certain taxonomical and relational behaviours. The hierarchical representation of the algorithms is addressed through dividing the algorithms according to the contexts in which they are executed. Classes including “Geometry Detection,” “Appearance Detection,” “Image Processing” and “Noise Reduction” follow such a hierarchal structure. Likewise, relational semantics are represented through properties. In broader terms, there are two types of relationships: one which applies to the geometries that the objects in Domain Concept possess, and other that applies against each other. The first category of relationship is used for detecting geometries. The object property “isDesignedFor” maps algorithms to the respective geometries. For example: Line Detection 1 (Ransac) isDesignedFor Lines. The second set of algorithm properties “input/output” are inter-relational properties to connect algorithms together, based on the compatibility of output from an algorithm to the outputs of others.

To get more intelligence for the detection and qualification, it is necessary to adapt processing to certain situations, depending on the data, the scene and the object characteristics. The created concept allows for these interactions, as it is able to automatically change the strategy based on a compromise of quality and risks. A part of the knowledge base is dedicated to risk-benefit factors that have influences on the algorithms, and have been deduced from the simulation's knowledge pattern. Since an algorithm might perform best with given parameters in one setting, and fail to deliver the same quality in other settings, it is important to evaluate the risk-benefit factors of every algorithm with various possible settings. The class "Risk Benefits" includes all of the risks and benefits possible due to previously mentioned reasons. The class contains instances such as "Distinct," "Illusive," "Noise," and "Error Detections." These instances are either the risks or the benefits that have influences on the algorithms as a whole, or at least the values of the parameters they contain.

5 Knowledge guidance for the object detection and qualification process

5.1 Knowledge-driven strategy

The knowledge formalization is based on the understanding of underlying semantics, and processing it through different knowledge technologies such as the Web Ontology Language (OWL). The top level ontology presents the main knowledge framework and holds generic semantics for all addressed domains. For the case studies, this contains: the scene, object geometries, spatial relations and algorithms. It originates from existing knowledge sources, such as information systems, or guidelines and rules of the carrying institution (like the Deutsche Bahn (DB)), and an extensive study on the sample scenarios. Logically, quality and completeness of such formalised knowledge has a large impact on the quality of the results and also has to be adapted to the individual application domain. In the most generic and difficult case, such a framework only contains the abstract and general knowledge of object categories, the structure of a scene, geometrical relations between objects, the structure of data, the nature of algorithms and the potentioal relationships among all these components. In a simpler scenario, with concrete information about potentially exisiting objects known through CAD or IFC files, for example, the detecting strategy can be guided more easily and might be reduced to a change detection problem.

Starting from the initial situation, the process iteratively updates the knowledge base (KB) at certain stages. At the beginning of each iteration, the content of the knowledge base is used to detect new features. This might be a new object or a new component of an object. These new feature geometries are then populated in the knowledge framework in order to extend the knowledge base for the next step of qualification. This qualification is performed through the content and the structure of the knowledge base, which has reasoning capacity based on property restrictions or rule languages (such as SWRL), and refines the actual content. This refined content enters into the next iteration.

The process is repeated until all entities have been completely annotated, and meets the following convergence conditions: (1) All objects defined on the knowledge side are detected and annotated (simple change detection). (2) A predefined number of iterations without refinement for any entity have been reached.

5.2 Usage of Algorithms guided by knowledge

The impact of object related knowledge is not restricted to the qualification alone; it also affects the algorithmic processing. The selection and behaviour of algorithms are not independent from other factors such as the type and characteristics of objects and data. Different algorithms are designed for different contexts. These differences can be addressed and properly modelled for usage within processing. For that purpose, the knowledge base hosts the algorithmic knowledge through a dedicated class "*Algorithm*." This class is linked to other classes inside the knowledge base, such as objects. This allows for the modification of the usage (e.g. parameter, sequences) of algorithms corresponding to the knowledge base details. The interrelationships among different algorithms are mapped through compatibility of their input and output characteristics. We use the well-known Djikstra's algorithm for finding the shortest path in the graph leading to the desired algorithm. This approach has the advantage of preventing the sequence of algorithms to loop, and allows for finding the optimal sequence.

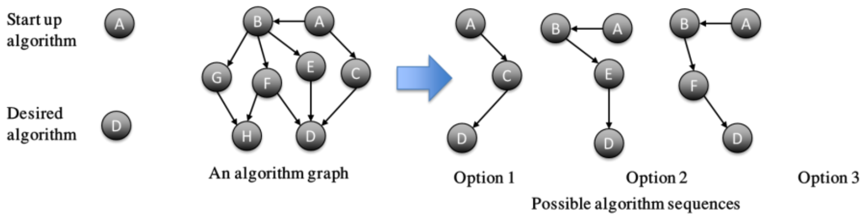

Fig. 4: Algorithm sequences extracted from the graph.

At any given iteration, the algorithms seek entities within the knowledge base which are either "identified", "unknown," or "ambiguous" and label them as such. Note that each label might (or might not) change with every new iteration. Based on the state of this information, the selection model (ASM) chooses an algorithm best suited to generate new characteristics, which will then help the next qualification step. This selection also integrates the choice of a best suited sequence out of several possible sequences (routes) of algorithms (or nodes).Many different knowledge components might have an impact in this context (data: noise, point density, point of view; object: size, shape, orientation; scene: possible objects, neighbourhood, etc.).

5.3 Qualification step

As seen in Section 4, the ontological schema holds the semantics of the objects such as the nature of its geometries and 3D spatial characteristics. This information helps to identify the nature of detected entities using the inference capacity of knowledge tools. The complexity of the rules needed depend directly on the existing complexity of the situation to be qualified. In simple cases, even very small rules are sufficient to produce a result. However, this concept also allows handling the inevitable complex situations.

A simple qualification of an entity (*GeneralizedGeometry*) based on a SWRL rule annotates an Electric pole, as found along railway tracks:

GeneralizedGeometry (?x) ^ hasHeight(?x, ?ht) ^ swrlb:greaterThan(?ht, 6)→ElectricPole(?x) (3)

A first extension of such simple geometric considerations is possible by the use of spatial relations (Metric, Topologic, and Directional) between the detected entities (BEN HMIDA, et al. 2012). It only requires having the appropriate algorithms available, and provides the result for the topological operation. ZLATANOVA, et al. 2002 gives a survey on different 3D models and relations. The spatial operators available for spatial query language consist of 3D topological operators (BORRMANN & RANK, 2008), 3D metric operators (BORRMANN, et al. 2009), 3D directional operators (BORRMANN & RANK, 2009) and finally 3D Boolean operators (BORRMANN, et al. 2006). In a simplified consideration, the following rule specifies that a "Building" defined in the ontology that overlaps a "Railway" (defined as well in the ontology), is a "RailwayStation".

Building(?b) ^ Railway(?r) ^ topo: overlaps(?b, ?r) → RailwayStation(?b) (4)

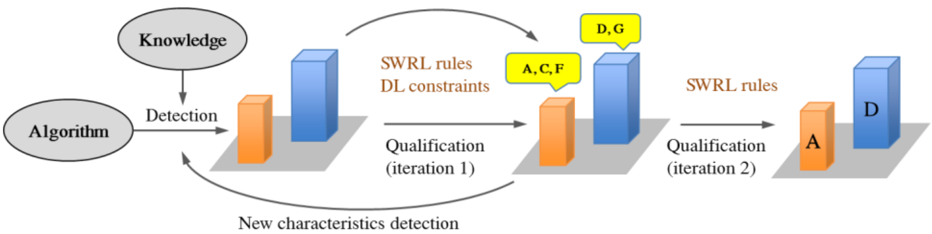

Fig. 5: Knowledge driven method for object detection and qualification process.

6 Case study

Two case studies illustrating our approach are presented in this section. These two case studies were taken from applying our system on scans from (DB) and from Frankfurt Airport (Fraport). In both cases, our goal was to detect and check relevant objects inside a defined work area. In the DB example, we deal with scans in the vicinity of the tracks. In the Fraport example, we use scans from an environment inside the airport buildings, typically a waiting area. Aspects of changes concerning the technical infrastructure are of main interest. Data comes from classical terrestrial laser scanning (Fraport) and from the LIMEZ III, a special train equipped with a laser scanner mounted at the head. The processing is shown for the initialization and refinement steps.

The origin of the whole process can be seen as collecting, structuring and modelling all available knowledge. This comprises the analysis of exisiting data bases, guidelines, rules or other information available from the user side. More general knowledge (for example, that a window is an opening inside a wall) also has to be collected and modelled. This knowledge is entered into the knowledge base and expressed in interrelated ontologies with rules, constraints and other components. Other relevant information (to probably exisiting individual objects) is also entered. The algorithm knowledge components are modeled, thus expressing relevant characteristics for the execution in context of the actual application (hasInput, hasOutput, isDesignedFor, etc.). The selection module uses the compatibility between input and output, and creates a graph based on the possible algorithm sequences.

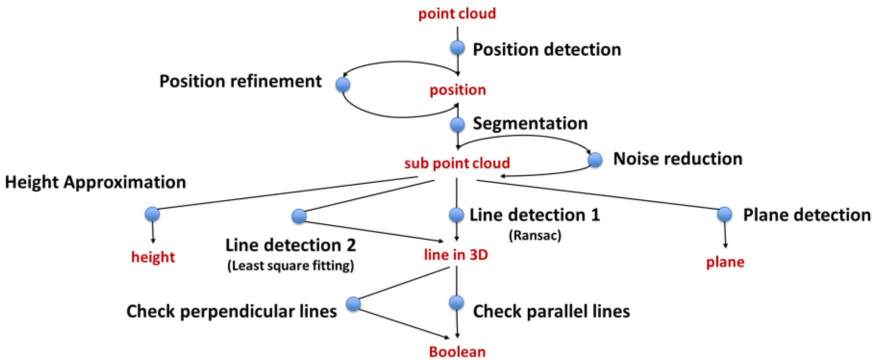

Fig. 6: Deployment of algorithms used for detecting object in the DB scene.

Although the DB possesses its own data base, the processing is handled as the most general case and does not use data of concrete existing objects. The whole processing therefore needs an initialisation in order to detect entities. Such an initialisation has to follow clear and prominent characteristics, allowing getting reliable candidates for a first qualification. In the DB case, such a prominent property can be found in the vertical structure of most objects (such as masts and signals). Such vertical structures are accessible by a vertical projection of the point cloud and based on analysis of the resulting feature values. In a second step, feature values are analysed in order to find evidence of sufficient characteristic entities. These entities are passed along with their coordinates and other feature values into the knowledge base as input for the first qualification step (Fig. 7a).

For the Fraport scenario, such an initialisation looks different. Here, objects are more complex and rather planar than vertical. Therefore a first step identifies the enclosing structure of walls in order to have first prominent entities (Fig. 7c).

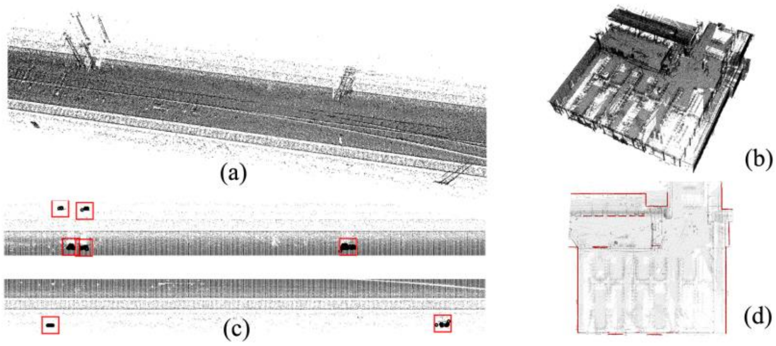

Figure 7. Point cloud representation of a section of a railroad (a).Results after executing the initialization step. Projecting the point cloud following the vertical direction, rectangles denote possible object positions (c). The scene of check-in area at Fraport (b), and the significant elements that are found, but not yet annotated (d).

In the case that the result of an initialisation was not able to allow a qualification - which only rarely occurs - further analysis is necessary. A refinement step attempts to detect additional characteristics of the entities found. The point cloud of an entity is therefore segmented into smaller parts (a “sub point cloud”), which are checked for additional features. This step follows again a strong to weak concept, in order to distinguish the most common properties. In the DB case, the knowledge base stores “hasHeight” as a prominent feature, while an appropriate algorithm (“Height approximation”) will be selected. Based on the different values of such features, knowledge classifies the entities as:

- Identified: as soon as a feature value is in the range of a class. This annotation has to be supported by subsequent qualifications and remains valid as long as no conflict is detected.
- Ambiguous: as soon as a feature value satisfies more than one class. Both annotations are stored and have to be separated by subsequent qualifications and remain doubtful as long as no separation is possible.
- Unknown: as soon as a feature value does not match an existing class. Further processing then requires the ASM to select other properties in order to continue the process.

Although a simple and evident example, this nevertheless shows the general logic, which can then be further extended with other considerations among entities. Comprehensibly, success is directly correlated to the ability to detect entities and the significance of the feature values chosen. Less characteristic features can also be used; however, they will need more iteration and more rules in order to come to a stable qualification.

The aspect of quality can also be incorporated into the concept. This might either be realised by thresholds modelling data noise, or by impacting the strategy of the algorithms. The latter case handles situations in which features are sensible to noise and corresponding algorithms might fail. In the DB case, for example, a mast of signals is represented by parallel vertical supports. As the point density and quality are not perfectly expressing this fact, a relevant pre-processing step is selected. The rule for the selection of the algorithms then looks as follows (Fig. 8a):

Position detection -> Segmentation -> Noise reduction -> Line detection 1 -> Check parallel

In the case of the Fraport scene, such a pre-processing is used for less prominent planar entities such as advertising panels. Here the algorithm sequence will appear as (Fig. 8b):

Position detection -> Segmentation -> Noise reduction -> Plane Detection

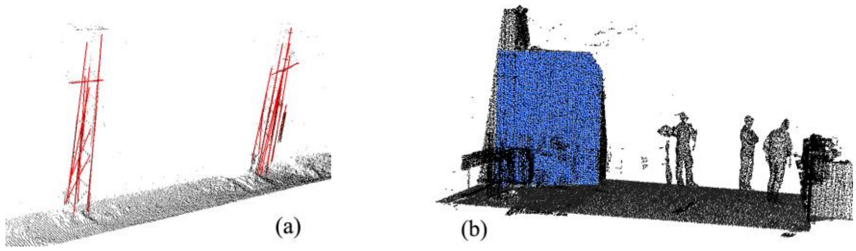

Figure 8.Results from detecting lines in 3D of a Signal in the rail road (a), and a plane of separated panel is detected at airport area (b).

These algorithms in the sequence then executed to recognize object features to update, and improved the quality and accuracy of the results (Fig. 9). The iterations are repeated to complete the annotation for all entities. The convergence conditions are applied to terminate the detection process for entities.

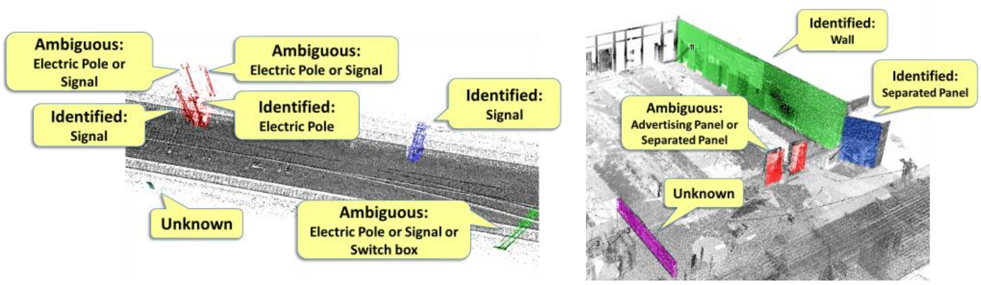

Figure 9. Every positions of object are given the first annotation result by knowledge. DB (left) and Fraport (right)

7 Conclusion

This paper presents a knowledge driven approach to detect objects in point clouds. It is based on semantics of different associated domains, which assist in detection and qualification of objects. Unlike other approaches, knowledge provides an overall base and is integrated into all steps of the processing, including the guidance of algorithms. This allows inter-relating the characteristics of algorithms with those of the objects in the domain of application. This provides the flexibility to infer the strategy from existing knowledge, and to adopt the processing to the needs of an application. The quality of results clearly depends on the robustness of the implemented algorithms and the selected strategy. However, the permanent iteration between algorithms and the knowledge base, as well as the hierarchical strategy following a strong to weak concept, allow for the smooth and careful construction of the knowledge base, which at the end will contain all entities which can be detected and identified. The approach is of a general nature and can be adapted for any field of application, any kind of data, and any kind of processing. Future work includes the expansion of the ontology, further implementation and testing of the rules, the improvement of the existing JAVA prototype application and the improvement and addition of 3D algorithms. We are also working towards the algorithmic learning mechanism in which the machine learns from various setups and situations, and adapts to new settings through learning.

Acknowledgements

The work presented in this paper is part of the research project WiDOP – Wissensbasierte Detektion von Objekten in Punktwolken für Anwendungen im Ingenieurbereich, funded by the German Federal Ministry for Research and Education (grant no. 1758X09). Partners in the project are Metronom

Automation GmbH and DB Netz AG (for the railway domain) and Fraport AG (for the facility management domain).

References

ANGUELOV, D., TASKARF, B., CHATALBASHEV, V., KOLLER, D., GUPTA, D., HEITZ, G. & NG, A., 2005: Discriminative learning of markov random fields for segmentation of 3d scan data. – Computer Vision and Pattern Recognition (CVPR), **2**, 169-176.

BECHHOFER, S., VAN HARMELEN, F., HENDLER, J., HORROCKS, I., MCGUINNESS, D. L., PATEL-SCHNEIDER, P. F. & STEIN, L.A., 2004: OWL web ontology language reference. – W3C recommendation, **10**.

BEN HMIDA, H., CRUZ, C., NICOLLE, C. & BOOCHS, F., 2011: From 3D Point Clouds To Semantic Objects An Ontology-Based Detection Approach, in Keod 2011. - International Conference on Knowledge Engineering and Ontology Development , Paris, France.

BEN HMIDA, H., CRUZ, C., NICOLLE, C. & BOOCHS, F., 2012: From Quantitative Spatial Operator to Qualitative Spatial Relation Using Constructive Solid Geometry, Logic Rules and Optimized 9-IM Model. - Proceeding 2012 ieee international conference on computer sciences and automation engeneering, Zhanjiajie China, **2**, 453.

BORRMANN, A. & RANK, E., 2009: Specification and implementation of directional operators in a 3D spatial query language for building information models. – Advanced Engineering Informatics, **23**, 32-44.

BORRMANN, A. & RANK, E., 2008: Topological operators in a 3D spatial query language for building information models. – The 12th Int. Conf. on Computing in Civil and Building Engineering (ICCCBE).

BORRMANN, A., SCHRAUFSTETTER, S. & RANK, E., 2009: Implementing metric operators of a spatial query language for 3D building models: octree and B-Rep approaches.– Journal of Computing in Civil Engineering , **23**, 34.

BORRMANN, A., VAN TREECK, C. & RANK, E., 2006: Towards a 3D spatial query language for building information models. – Int. Conf. for Computing and Decision Making in Civil and Building Engineering.

CANTZLER, H., FISHER, R. & DEVY, M., 2002: Quality enhancement of reconstructed 3D models using coplanarity and constraints. – Pattern Recognition, 34-41.

DUAN, Y., CRUZ, C. & NICOLLE, C., 2010: Managing semantics knowledge for 3D architectural reconstruction of building objects. – Eighth ACIS International Conference on Software Engineering Research, Management and Applications (SERA), 121-128.

DURAND, N., DERIVAUX, S., FORESTIER, G., WEMMERT, C., GANCARSKI, P., BOUSSAID, O. & PUISSANT A., 2007: Ontology-based Object Recognition for Remote Sensing Image Interpretation. – Tools with Artificial Intelligence,ICTAI 2007, 472-479.

HORROCKS, I., PATEL-SCHNEIDER, P. F., BOLEY, H., TABET, S., GROSOF, B. & DEAN, M., 2004: SWRL: A semantic web rule language combining OWL and RuleML. – W3C Member submission, **21**, 79.

MAILLOT, N. & THONNAT, M., 2008: Ontology Based Complex Object Recognition. – Image and Vision Computing. **26**, (1), 102-113.

NÜCHTER, A., WULF, O., LINGEMANN, K., HERTZBERG, J., WAGNER, B. & SURMANN, H.,2006: 3D mapping with semantic knowledge. – RoboCup 2005: Robot Soccer World Cup IX, 335-346.

PU, S. & VOSSELMAN, G., 2006: Automatic extraction of building features from terrestrial laser scanning. – International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences , **36**, 25-27.

PU, S. & VOSSELMAN, G., 2009: Knowledge based reconstruction of building models from terrestrial laser scanning data. – ISPRS Journal of Photogrammetry and Remote Sensing, **64**, (6), 575-584.

RIPPERDA, N. & BRENNER, C., 2006: Reconstruction of Facade Structures Using a Formal Grammar and RjMCMC. – DAGM'06 proceedings of the 28th conference on Pattern Recognition, 750-759.

SCHOLZE, S., MOONS, T. & VAN GOOL, L., 2002: A probabilistic approach to building roof reconstruction using semantic labelling. – Pattern Recognition, 257-264.

TEBOUL, O., SIMON, L., KOUTSOURAKIS P. & PARAGIOS, N., 2010: Segmentation of Building Facades Using Procedural Shape Priors. – Computer Vision and Pattern Recognition (CVPR), 3105-3112.

TRIEBEL, R., SCHMIDT, R., MOZOS, O. & BURGARD, W., 2007: Instance-based AMN classification for improved object recognition in 2D and 3D laser range data. – Proceedings of the 20th international joint conference on Artificial intelligence, 2225-2230.

VOSSELMAN, G. & DIJKMAN, S., 2001:3D building model reconstruction from point clouds and ground plans. – International Archives of Photogrammetry Remote Sensing and Spatial Information Sciences, Athens, Georgia, USA , **34**, (3/W4), 37-44.

ZLATANOVA, S., RAHMAN, A. A. & SHI, W., 2002: Topology for 3D spatial objects. – International Symposium and Exhibition on Geoinformation, 22-24.

Address of the Author:
Hung Truong, M.Sc., Fachhochschule Mainz, Institut für Raumbezogene Informations- und Messtechnik, Lucy-Hillebrand-Str. 2, 55128 Mainz
Tel.: +49-6131-628-1489, Fax: +49-6131-628-91489, e-mail: truong@geoinform.fh-mainz.de