


Article

Ontology for BIM-Based Robotic Navigation and Inspection Tasks

Fardin Bahreini ¹, Majid Nasrollahi ², Alhusain Taher ³ and Amin Hammad ^{1,*}

¹ Concordia Institute for Information Systems Engineering, Concordia University, Montréal, QC H3G 1M8, Canada; fardin.bahreini@concordia.ca

² AIRM Consulting Co., Winnipeg, MB R3C 4G1, Canada

³ eStructure Data Centers, Montréal, QC H4Z 1A1, Canada

* Correspondence: amin.hammad@concordia.ca

Abstract: The availability of inspection robots in the construction and operation phases of buildings has led to expanding the scope of applications and increasing technological challenges. Furthermore, the building information modeling (BIM)-based approach for robotic inspection is expected to improve the inspection process as the BIM models contain accurate geometry and relevant information at different phases of the lifecycle of a building. Several studies have used BIM for navigation purposes. Also, some studies focused on developing a knowledge-based ontology to perform activities in a robotic environment (e.g., CRAM). However, the research in this area is still limited and fragmented, and there is a need to develop an integrated ontology to be used as a first step towards logic-based inspection. This paper aims to develop an ontology for BIM-based robotic navigation and inspection tasks (OBRNIT). This ontology can help system engineers involved in developing robotic inspection systems by identifying the different concepts and relationships between robotic inspection and navigation tasks based on BIM information. The developed ontology covers four main types of concepts: (1) robot concepts, (2) building concepts, (3) navigation task concepts, and (4) inspection task concepts. The ontology is developed using Protégé. The following steps are taken to reach the objectives: (1) the available literature is reviewed to identify the concepts, (2) the steps for developing OBRNIT are identified, (3) the basic components of the ontology are developed, and (4) the evaluation process is performed for the developed ontology. The semantic representation of OBRNIT was evaluated through a case study and a survey. The evaluation confirms that OBRNIT covers the domain's concepts and relationships, and can be applied to develop robotic inspection systems. In a case study conducted in a building at Concordia University, OBRNIT was used to support an inspection robot in navigating to identify a ceiling leakage. Survey results from 33 experts indicate that 28.13% strongly agreed and 65.63% agreed on the usage of OBRNIT for the development of robotic navigation and inspection systems. This highlights its potential in enhancing inspection reliability and repeatability, addressing the complexity of interactions within the inspection environment, and supporting the development of more autonomous and efficient robotic inspection systems.



Citation: Bahreini, F.; Nasrollahi, M.; Taher, A.; Hammad, A. Ontology for BIM-Based Robotic Navigation and Inspection Tasks. *Buildings* **2024**, *14*, 2274. <https://doi.org/10.3390/buildings14082274>

Academic Editor: Heap-Yih Chong

Received: 27 June 2024

Revised: 16 July 2024

Accepted: 19 July 2024

Published: 23 July 2024

Keywords: ontology; BIM; robotics; navigation; inspection



Copyright: © 2024 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 (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Inspection is indispensable in the construction industry. Robots are used to automate the process of inspection during the construction and operation phases. The use of advanced technologies (e.g., scanners and sensors) has made the inspection process more accurate and reliable [1]. The complexity of the interactions with the surrounding building environment is the main challenge for inspection robots [2]. To overcome this challenge, an ontology can be used as a basis for the robot's task planning and execution. The robotic system utilizes and processes the ontology as the robot's central data store. In this regard,

there is a need for an interface between the Robot Operating System (ROS) [3] and different ontologies for robotic inspection [4]. To accomplish the tasks correctly, the autonomous robot needs to deal with high-level semantic data along with low-level sensory/motor data. Therefore, a variety of knowledge, including the robot's low-level data related to perception and high-level data about the environment, objects, and tasks, needs to be integrated [5].

Building information modeling (BIM) is an approach to model all the information related to buildings by representing the geometrical and spatial characteristics, and is supported by the international standard *Industry Foundation Classes* (IFC) [6]. BIM models comprise useful information about the building environment, which can help the inspection robot overcome task complexity. On the other hand, the ROS [3] uses several navigation methods, such as Lidar Odometry and Mapping (LOAM) and Simultaneous Localization and Mapping (SLAM), which help the robot build its map based on the collected data about the environment [7]. Regarding the different lifecycle phases, the BIM models of a building include as-designed at the design phase, as-built at the construction phase [8], and as-is at the operation and maintenance (O&M) phase. These models should be considered in the navigation and inspection processes. It should be noted that each of these models has several versions and should be continuously updated to reflect design, construction, renovation, and repair changes in the different phases of the lifecycle. Mismatches between the as-designed BIM model (or as-built BIM model) and the as-is state of the surrounding environment can create problems during the building's navigation and inspection tasks.

In the area of BIM-related ontologies, many minimal ontologies were created for the linked building data purpose [9]. The purpose of creating minimal ontologies (e.g., Building Topology Ontology (BOT) [9] and Damage Topology Ontology (DOT) [10]) is to solve interoperability issues by creating small, lightweight ontologies that reduce the complexity of IFC. Minimal ontologies are based on the hypothesis that these ontologies are easier to understand and use by software developers. However, application-level integration and developing an interface is difficult because application developers have to map the different concepts in different ontologies. Furthermore, it is difficult to update and modify the domain ontology after mapping from distributed heterogeneous ontologies to the computer software that uses them [11]. On the other hand, many studies focused on integrating ontologies with BIM information. Therefore, in order to perform the desired task by a robot in an efficient way (e.g., [12]), the required concepts must be integrated into an integrated ontology.

The navigation concepts in this paper are based on using the semantic knowledge of the BIM concepts for navigation tasks. The BIM-based approach is also expected to improve the inspection process. The robotic task must be performed in such a way that the process considers reliability, repeatability, and safety. Therefore, it is necessary to enhance operational consistency in the inspection environment [13]. Robotic systems' capabilities have progressed over time, and these systems have become dependent on multiple components with diverse functions. In most developed systems, the modules are created independently by different individuals with different technical expertise. Thus, a clear definition of the relationships between the system's various components is needed. The system's structure and related components must have a straightforward design and documentation to solve this problem [4]. A clear and accurate description of the environment and the task can help the robot to achieve the tasks more autonomously [14]. The robot's declarative knowledge represents the task's objects, properties, and objects' relationships in a semantic model [15]. The robot can use this declarative knowledge to perform the task more accurately. However, the research in this area is still limited and fragmented, and there is a need to develop an integrated ontology to be used as a knowledge model for the logic-based inspection of building defects. The objective of this paper is to develop a BIM-based ontology to cover the different types of information and concepts related to robot navigation and inspection tasks. The aim of this ontology is to help system engineers involved in developing robotic inspection systems by identifying the different concepts and relationships between robotic inspection and navigation tasks based on BIM information. This paper is an extension of

our previous work [16]. The ontology is called OBRNIT (ontology for BIM-based robotic navigation and inspection tasks). OBRNIT covers the high-level knowledge of the robot comprising robotic and building concepts, and navigation and inspection information. Unlike previous studies, OBRNIT integrates comprehensive robot characteristics, building features, and specific inspection and navigation tasks, enhancing robotic inspection processes. Leveraging BIM data improves inspection accuracy, repeatability, and efficiency in complex environments, providing a foundational model for future advancements in robotic inspection systems.

The use case context is an inspection robot that is navigating in a building with partial knowledge of the environment because of changes in the available information due to construction and renovation scheduling issues, unexpected obstacles in the building, etc. As shown in Figure 1, to define the requirements of OBRNIT, a UML (Unified Modeling Language) use case diagram is presented. The actor is a robot, and the associations between the actor and the use cases are shown with solid lines. The dependency relationships are shown with dotted lines. *Includes* relationships indicate that the involved use case is a part of the base use case. *Extends* relationships indicate that the base use case does not depend on the extending use case, and specific criteria are needed for the occurrence of the extending use case.

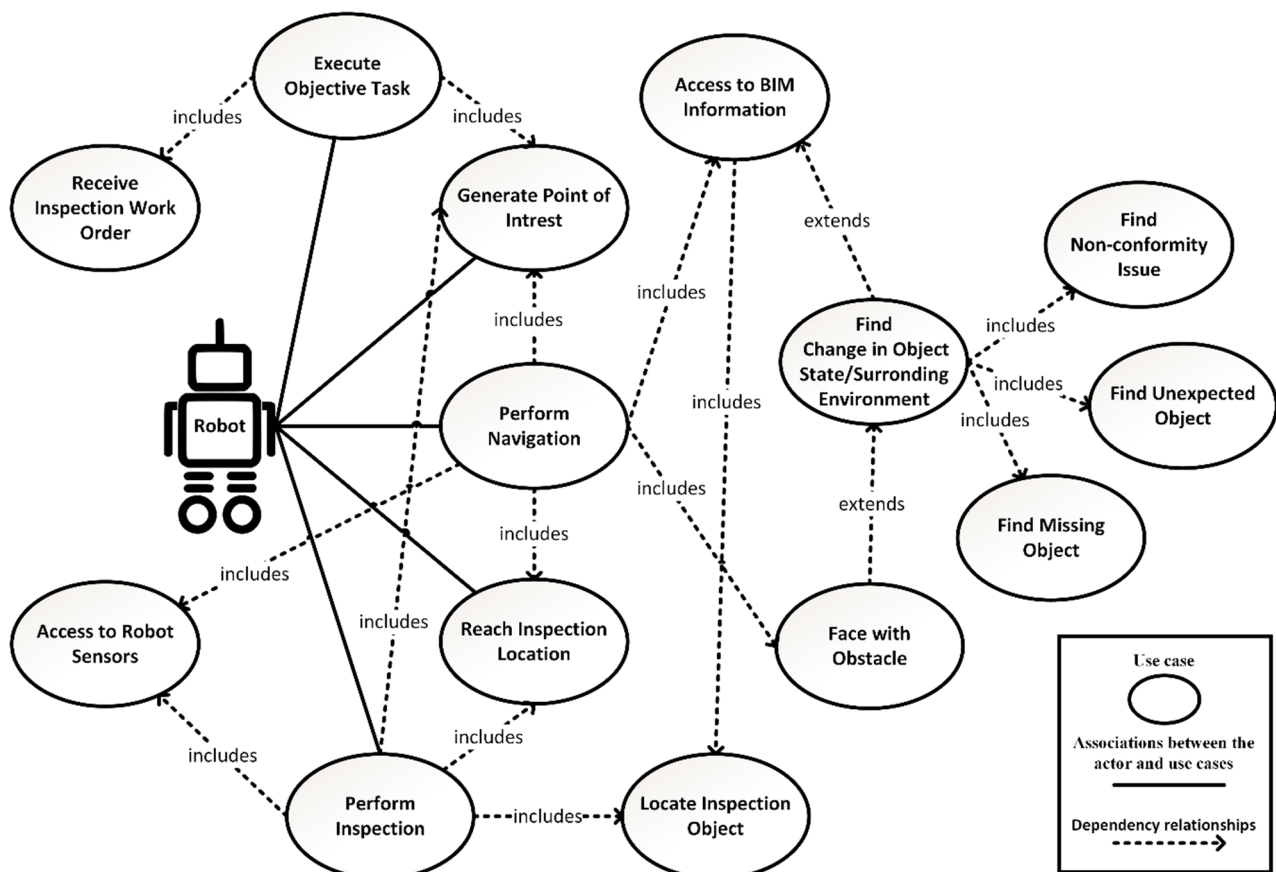


Figure 1. Use case diagram of OBRNIT.

The rest of the paper is organized as follows: The next section provides the related literature. Then, OBRNIT development is discussed. The following section focuses on the evaluation of OBRNIT using a case study and a survey. Finally, the final section presents the conclusions and future work.

2. Literature Review

2.1. Inspection Tasks during the Construction and Operation Phases

2.1.1. Construction Inspection

Inspection during the construction phase is an important task in the construction industry. A lack of proper inspection will increase the cost of maintenance in the operation phase [17]. Based on [18], the causes of construction defects include the misinterpretation of design and inaccurate measurement. Kim et al. proposed a framework for the dimensional and surface quality assessment of precast concrete elements using BIM and 3D laser scanning [19]. Park et al. proposed a framework for construction defect management using BIM and an ontology-based data collection template to represent the inspected defects during the construction phase [20]. Tekin et al. identified that concrete works, reinforcement works, and HVAC are the top three areas where problems are frequently observed during inspections, emphasizing the critical need for rigorous quality control in these areas [21].

Recent technologies (e.g., light detection and ranging (LiDAR) scanner) are integrated with BIM to enhance the capabilities of construction inspection [22]. These technologies can capture real-time data from the site [23]. The use of computer vision techniques can help to inspect most of the surface defects [24]. Bolourian and Hammad considered the potential locations of the defects on the inspected surfaces and proposed a path-planning method for LiDAR-equipped unmanned aerial vehicles (UAVs) [25]. Lundeen et al. developed an adaptive inspection framework for construction robots to detect the location and geometry of joints and fill these joints [26]. Freimuth et al. used BIM for UAV flight path planning for construction inspection [27]. Bahreini and Hammad introduced a semi-automated process for as-inspected modeling, integrating the results of defect semantic segmentation with BIM, enhancing the efficiency of managing and visualizing detected defects [28].

2.1.2. Inspection during Operation Phase

The other area of inspection is inspection during the operation phase. Facilities need regular inspections to satisfy their predetermined functions. Imperfections in the facilities are described as defects, errors, faults, failures, quality deviations, nonconformances, anomalies, snags, reworks, etc. [29]. Valença et al. introduced 'Feeling-BIM' integrating automated facade inspection with residents' sentiments, enhancing maintenance decision making by combining technical assessments with user feedback [30]. Metni and Hamel used the visual inspection of the structures in the operation phase and discussed the challenges of considering the orientation limits for UAVs to make sure that the inspected object is within the field of view of the sensor [31].

To reflect the changes in the BIM model related to inspection during the operation phase, Chen et al. focused on defect modeling [32]. Motamedi et al. proposed a defect/degradation model that includes various defect types, relationships between elements and defects, and the processes related to the inspection, evaluation, and repair of defects [33]. Hammad et al. developed an inspection ontology using BIM for lifecycle inspection and repair information modeling [34]. Kasireddy and Akinci proposed integrating inspection data with IFC to support condition assessment [35]. Ekba et al. proposed a systematic approach to technical inspection, which includes detailed visual and instrumental inspection, as well as the identification and mapping of defects and damages typical for the different stages of construction [36]. Choi et al. introduced a visualized semi-automated approach for extracting building condition data and integrating them with BIM using 3D point clouds. This method improves the efficiency of maintenance strategies by providing accurate and reliable inspection data [37]. Mohamed and Tran developed an approach for estimating inspection staffing needs for construction projects, emphasizing the importance of adequate and experienced inspection staff for ensuring project quality [38]. Furtner and O'Brien described an automated process for creating an IFC-compliant damage model from digital inspections [39]. Chen et al. reviewed advancements in integrating BIM and Life Cycle Assessment (LCA), highlighting how BIM can streamline LCA processes. The study enhances efficiency in environmental performance assessments [40]. Tan et al.

introduced a BIM-based defect data management (DDM) platform that integrates real-time inspection data, providing a comprehensive management system for lifecycle inspection and repair [41].

2.1.3. Post-Disaster Inspection

Post-disaster inspection is the third inspection type which should be conducted in the event of a disaster and before re-occupying the building to evaluate potential health and safety hazards. Search and rescue inspection is also carried out after disasters, which is a time-sensitive task, and it needs quick action to reduce the potential injuries and damages [42]. Shin and Cha proposed a new quality inspection process model that integrates advanced technologies, which significantly enhances inspection practices during post-disaster scenarios [43]. Liang et al. discussed the use of UAVs for efficient and accurate monitoring and inspection in post-disaster scenarios [44].

2.2. Using Robots for Inspection

With the advancement of technology, autonomous robots have evolved and are equipped with advanced capabilities [45]. Service robots can be used for different purposes in the architecture, engineering and construction, and facilities management (AEC/FM) industry. Mobile robots are designed for sensing, navigation, inspection, and remote operation in dangerous situations. Industrial applications show the variety in the types, capabilities, and uses of robots. Autonomous unmanned systems, including UAVs, unmanned ground vehicles (UGVs), and autonomous underwater vehicles (AUVs), can be used for quality inspection [46]. For instance, Doxel is a LiDAR-equipped robot that scans construction sites to monitor work [47].

Lin et al. introduced GLEWBOT, a bioinspired climbing robot for inspecting exterior wall tiles, integrating the leech, gecko, and woodpecker techniques for enhanced accuracy and efficiency [48]. Halder and Afsari reviewed various robotic systems, highlighting UAVs and UGVs as the most common types used for inspection and monitoring tasks [49]. Patil et al. demonstrated the use of UAVs for construction site inspection, which significantly enhances efficiency and reduces the cost of inspection processes [50]. Pu et al. introduced a framework named AutoRepo for the automated generation of construction inspection reports using unmanned vehicles and multimodal large language models, which can expedite the inspection process and improve report quality [51].

An autonomous robot control system enables robots to perform human activities in a building [52]. Cognitive Robot Abstract Machine (CRAM) is a software toolbox for designing and implementing cognitive-enabled autonomous robots, which is built using the Robot Operating System (ROS) [3] framework. The two main parts of CRAM are (1) CRAM Plan Language (CPL) and (2) knowledge processing system (KnowRob) [12].

2.3. IFC-Based Navigation

Path planning in 3D spaces needs information related to spaces and their functions, geometry, locations, and obstacle and accessibility information. BIM can help extract precise and up-to-date semantic and geometrical data from the building model [53]. As an example, IfcWall is used to define general walls.

IFC schema represents objects and their semantic relationships [54]. For example, Lin et al. used an IFC file as the input for path planning. They extracted all the geometric and semantic information from the IFC file and mapped them onto a planar grid [55]. The logical network is a representation of the full 3D building model and a detailed navigable network (e.g., spatial relations between floors, rooms with shared walls, etc.) [56].

In another effort, Rasmussen et al. stated that topological relationships between zones, elements, or zones and elements of a BIM model can be described as *interface* class in their proposed minimal BOT [9]. Furthermore, they stated that topological relationships can be used to specify restricted zones in the navigation. BOT can be used in combination with different ontologies to define the building products such as walls and windows. Hou and

Ju proposed a method for robot indoor path planning using BIM integrated with the A^* algorithm, improving the efficiency and accuracy of navigation in reconstructed building environments [57]. Liu et al. developed a hybrid map-based path planning method for robot navigation in unstructured environments, combining 2D grid and 2.5D digital elevation maps to enhance the safety and efficiency of navigation [58]. Zhai et al. enriched the BIM data schema (IFC) with IndoorGML, combining geometric information with spatial data to establish an indoor navigation model. This model enhances the capabilities of quadruped robots for automated 3D scanning by optimizing scan routes and improving computational efficiency, coverage, and scan point count [59].

2.4. Review of Related Ontologies

Ontology is defined as the explicit shared knowledge and conceptualization of the domain [60]. The ontology has two main elements: the related vocabulary to the domain of interest, and the knowledge representation using this vocabulary to describe the domain [61]. Different tools and languages are used to build ontologies. Protégé [62] and OntoEdit [63] are two examples of ontology editing environments. The web ontology language (OWL) and resource description framework (RDF) are examples of the languages used to represent ontologies in the human- and machine-readable formats [64].

2.4.1. AEC/FM Ontologies

The entire IFC schema is available in a large ifcOWL ontology, representing building data using semantic web [65]. In addition to ifcOWL ontology, many ontologies have been developed in the AEC/FM industry. For example, Cacciotti et al. presented an ontology for the diagnosis of damage in order to process and manage cultural heritage damage information [66].

Moreover, many minimal ontologies were created for the linked building data purpose. Linked building data principle means using a web-compatible standard for the exchange of web-based information [9]. BEO (Building Element Ontology) [67] and MEP (Mechanical, Electrical, and Plumbing) ontology [68] are two ontologies extracted from the IFC schema. These two ontologies do not include any relations, which can be used based on user requirements in different domains. As explained in Section 1, BOT and DOT are extensible minimal ontologies to describe building spaces and damages, respectively. Rasmussen et al. presented an Ontology for Property Management (OPM), which is a minimal high-level ontology for managing changes and property valuation over time [69]. Wagner et al. developed Ontology for Managing Geometry (OMG) to connect the geometric description to the building element [70]. Bonduel et al. developed Ontology for Geometry Formats (FOG) to exchange descriptive geometric data [71].

On the other hand, as explained in Section 1, many studies focused on integrating ontologies with BIM information. For example, Niknam and Karshenas proposed BIM Shared Ontology (BIMSO) to be extended with different building domain ontologies [11]. Table 1 shows the ontology metrics of some of the publicly available BIM-based ontologies.

2.4.2. Ontologies for Robots

In the robotic area, ontologies can be used for different applications such as general robotic purposes (e.g., standardization [72,73]) and ontologies for autonomous robots (e.g., description/reasoning about the environment and tasks) [14,74]. Robotic ontologies embody the real-world description of objects, properties, and relationships in the domain [75]. KnowRob is an OWL-based robotic ontology that contains a small core system and a large set of optional modules which are developed to perform human activities in a building. The KnowRob ontology v.1.0 has 742 classes, 176 relations, 119 attributes, and 23 individuals [12].

Table 1. Ontology metrics of some of the publicly available BIM-based ontologies (adapted from [76]).

Ontology	Metrics			
	Classes	Relations	Attributes	Individuals
BOT v.0.3.2 [9]	10	16	1	5
FOG v.0.4 [71]	3	14	119	4
ifcOWL v.4.1 [65]	1360	1644	5	1171
DOT v.0.8 [10]	13	13	3	1
MEP v0.1.0 [68]	484	0	1	1
OMG v.0.3 [70]	8	17	2	0
BEO v.0.1.0 [67]	183	0	1	1
OPM v.0.1.0 [69]	17	8	4	1

2.5. Related Works

In this section, the research about integrating some aspects of robotics, inspection, and navigation with BIM is reviewed. Table 2 lists a summary of the most related papers including the following information: (1) the purpose of the inspection; (2) the building lifecycle phase (e.g., construction and operation) in which the inspection is conducted; (3) in the case of using a robot, the type of the robot (i.e., UAV and UGV); (3) the type of sensor (i.e., RGB camera, depth camera, thermal camera, or LiDAR); (4) using BIM model or IFC concepts in the process, and in the case of using BIM, considering mismatch between the actual structure and the model; and (5) using the knowledge-based method (i.e., ontology).

Some studies focused on using different sensors and robots for inspection purposes. All the studies considered at least one type of sensor (i.e., LiDAR or camera), which could be an element of the robot in an integrated platform or could be mounted on the robot. The target element of the inspection was building/infrastructure elements and specific defects, such as cracks on steel or concrete surfaces. In addition to the review of most related papers to robotic inspection in Table 2, Hamledari et al.'s [77] and Wang et al.'s [78] papers are added only because their works considered a mismatch between the actual structure and the BIM model.

Despite the great benefits of the reviewed papers, they have one or more of the following limitations: (1) Robot awareness about the environment and the accuracy of object information and interactions during the task can be improved by considering a semantic description (i.e., ontology). (2) A standard BIM model was not used as a reference for navigation and localizing the defects. (3) In the case of having a BIM model, the BIM elements were not updated based on mismatch considerations, and the functional properties of the BIM elements were not used for inspection. The BIM model is assumed to be complete and reliable. (4) Robotic navigation was not considered in some studies. Navigation here refers to using a path generation method and obstacle avoidance based on sensor data.

From the studied literature, the integration of the knowledge of robotic inspection and the construction domain is a key factor in an effective and efficient inspection, which needs more attention. In the next sections, our developed ontology (OBRNIT) aims to cover all the concepts related to BIM and robotic navigation and inspection as an integrated knowledge representation.

Table 2. Related works.

Paper	Inspected Elements	Phase	Navigation	Robot			Sensor			BIM		Ontology
				UAV	UGV	LiDAR	RGB Camera	Depth Camera	Thermal Camera	Model/IFC	Mismatch Consideration	
An autonomous thermal scanning system with which to obtain the 3D thermal models of buildings [79]	Indoor building elements	O	✓	-	✓	✓	✓	-	✓	-	-	-
A framework for the automated acquisition and processing of as-built data with autonomous unmanned aerial vehicles [80]	Building elements	C	✓	✓	-	✓	✓	✓	-	✓	✓	-
The automated robotic monitoring and inspection of steel structures and bridges [81]	Steel cracks	O	✓	-	✓	✓	✓	✓	-	-	-	-
Automatic wall defect detection using an autonomous robot: a focus on data collection [82]	Walls	O	✓	-	✓	✓	-	-	-	-	-	-
Autonomous robotic exploration by incremental road map construction [83]	Indoor building elements	O	✓	-	✓	✓	-	-	-	-	-	-
Planning and executing construction inspections with unmanned aerial vehicles [84]	Building roofs	C	✓	✓	-	-	✓	-	-	✓	-	-
Tunnel structural inspection and assessment using an autonomous robotic system [85]	Concrete cracks	O	✓	-	✓	✓	✓	✓	-	-	-	-
The design and development of an inspection robotic system for indoor applications [86]	Building elements (tested on walls)	O	✓	-	✓	-	✓	✓	✓	-	-	-
A semi-autonomous mobile robot for bridge inspection [87]	Concrete cracks (tested on columns)	O	-	-	✓	-	✓	-	-	-	-	-
The IFC-based development of as-built and as-is BIMs using construction and facility inspection data: site-to-BIM data transfer automation [77]	Building elements: walls, doors, outlets, and light fixtures	O	-	-	-	-	✓	-	-	✓	✓	-
The automated quality assessment of precast concrete elements with geometry irregularities using terrestrial laser scanning [78]	Precast concrete elements	C	-	-	-	✓	-	-	-	✓	✓	-
Infrared building inspection with unmanned aerial vehicles [88]	Building elements (tested on roofs and roof windows)	O	-	✓	-	-	-	-	✓	-	-	-
Efficient search for known objects in unknown environments using autonomous indoor robots [89]	Indoor building elements	O	✓	-	✓	-	✓	✓	-	-	-	✓
A robotic crack inspection and mapping system for bridge deck maintenance [90]	Concrete cracks	O	✓	-	✓	✓	✓	-	-	-	-	-
Low-cost aerial unit for the outdoor inspection of building façades [91]	Building facade and envelope elements (tested on facade openings)	O	-	✓	-	-	✓	✓	-	-	-	-
Auto inspection system using a mobile robot for detecting concrete cracks in a tunnel [92]	Concrete cracks (tested on walls)	O	-	-	✓	-	✓	-	-	-	-	-

Notes: C = construction phase and O = operation phase.

3. Developing OBRNIT

3.1. Methodology Workflow

The main steps for developing OBRNIT are explained in this section. The methodology for developing OBRNIT is METHONTOLOGY, which is clear, well-documented, mature, and based on the experience of other domains' ontology development [93]. OBRNIT development based on METHONTOLOGY includes the initial, development, and final stages as shown in Figure 2. The best practices and knowledge in the robotic inspection domain are used to develop OBRNIT. The initial stage involves steps to specify the scope, main concepts, and the taxonomies of OBRNIT. The scope of OBRNIT is defined based on the requirements. Research papers, textbooks, and online resources are used as sources for the requirements (e.g., properties). The ontology needs to cover all the concepts about the robot characteristics, building characteristics, and inspection and navigation tasks. The competency questions need to be defined as a part of the requirements of the robotic inspection domain. The competency questions are identified based on the use case diagram (Figure 1) and the reviewed literature to define the key challenges OBRNIT can address, as shown in Table 3.

Table 3. Competency questions.

Q1	How to locate the defect in the BIM model?
Q2	How to relate the mobility characteristics of the robot with the conditions of the building based on the BIM model?
Q3	How to benefit from the BIM model in defining the path of the robot?
Q4	How to use the sensors of the robot to find the mismatches with the BIM model for replanning the path of the robot?
Q5	How to select the suitable sensors for the robot for the specific inspection task?

Furthermore, this step helps to consider the size of the development and the level of detail that needs to be covered in OBRNIT. The next step is defining the concepts and taxonomies for OBRNIT. The data related to OBRNIT are gathered in this step. The list of requirements from the defining scope step helps the process of ontology development. Communication with experts and end-users along with receiving feedback from them is essential during the whole cycle of this stage.

The development stage is devoted to constructing and verifying the initial structure of OBRNIT. In the first step of the development stage, the conceptualization model is clearly represented and implemented in a formal language (e.g., OWL) to be later accessible by computers and used by different systems. The development of OBRNIT involves reusing and adapting BIM concepts. BEO v.0.1.0 [67], which is based on the IfcBuildingElement subtree in the IFC specification and ifcOWL ontology [65], is a good starting point for including the relevant BIM concepts to OBRNIT. BEO is available in the OWL format, which facilitates the integration process. Moreover, BOT v.0.3.2 [9] and DOT v.0.8 [10] ontologies are integrated and adapted in the development of OBRNIT to represent the required concepts related to damages and building topology, respectively.

The ontology integration in the METHONTOLOGY method can be carried out at the conceptualization level. The methods to reuse available ontologies are (1) ontology merging, (2) ontology alignment, and (3) ontology integration. Ontology merging refers to unifying two or more available ontologies by comparing the available ontologies and finding similarities between their domain information. Ontology alignment refers to mapping the concepts and relationships in two or more available ontologies to find equivalency between them. This method requires the smallest number of changes, and it is a simpler form of merging [94]. Ontology integration refers to integrating one or more available ontologies in the process of developing a new ontology by adapting, extending, specializing, or assembling. The ontology integration method is selected in this research as it saves the effort to reuse and adapt the components that are needed to complete OBRNIT.

The next step of the development stage is verifying the developed ontology. Based on the consistency rules and competency questions, this process examines the ontologies from the technical perspective.

The final stage is to add new, or modify the existing, relationships and evaluate OBRNIT with experts and end-users through evaluation questions. In this stage, the ontology is improved with the suggestions of the domain experts and end-users to fulfill real-world requirements. OBRNIT evaluation is performed through a case study and a criteria-based evaluation method [95]. Similar ontologies in the robotic inspection domain are not available to compare the developed ontology with a benchmark ontology or high-level standards in the domain. The final step is documenting the developed OBRNIT. Obtaining knowledge, evaluation, and documentation are involved throughout the whole life cycle of ontology development. Each step of the METHONTOLOGY method is presented using the IDEF5 (Integrated DEFINition) [96] ontology description method, which includes detailed information about the input, output, control, and mechanism. The next section explains in detail about the ontology development. The following section focuses on the verification and evaluation steps.

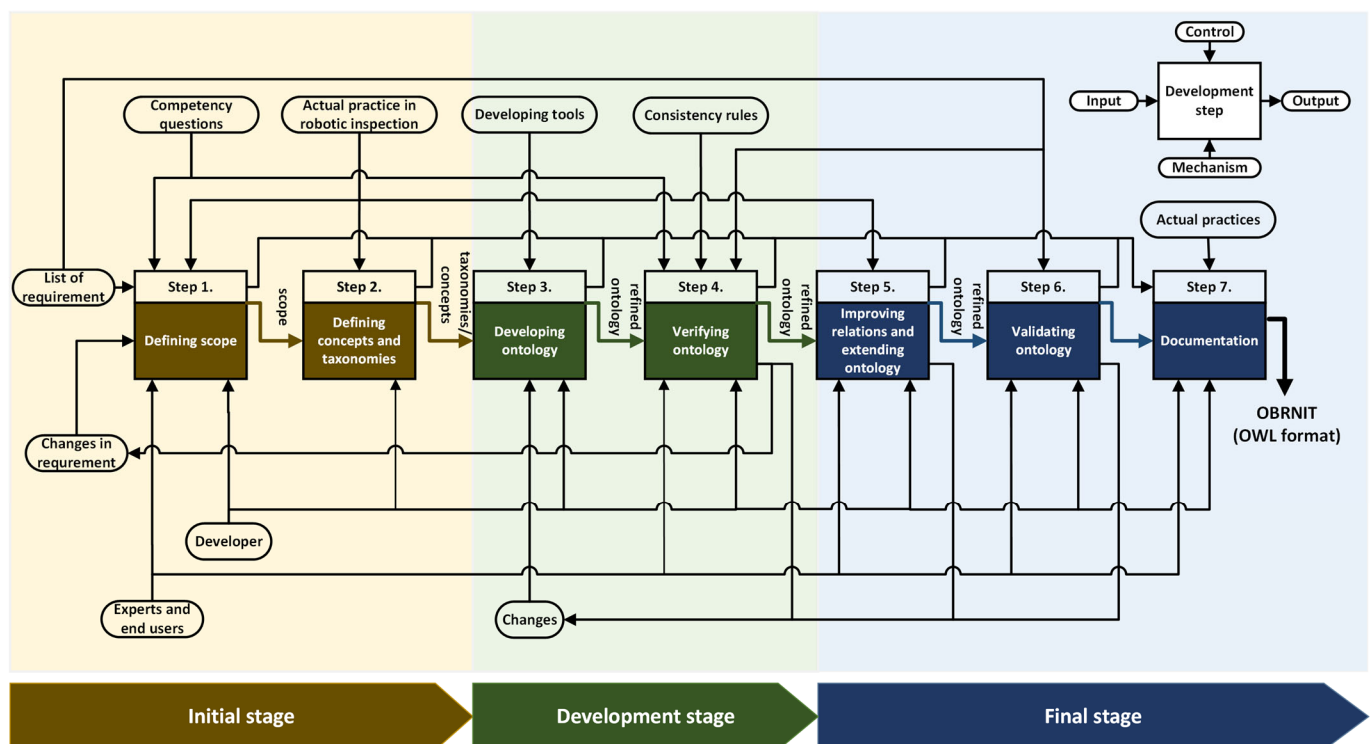


Figure 2. Development workflow of OBRNIT (adapted from [97]).

3.2. Components of OBRNIT

Some concepts from the BIM and KnowRob ontology [12] are used as parts of this study. Protégé [62] is used to develop OBRNIT and to integrate it with BEO, BOT, and DOT. OBRNIT has 386 classes, 45 relationships, 52 attributes, and 8 individuals. The current version of OBRNIT is available at <https://github.com/OBRNIT/OBRNIT> (accessed on 18 July 2024).

OBRNIT covers four main groups of concepts including (1) robot concepts, (2) building concepts, (3) navigation task concepts, and (4) inspection task concepts, which are explained in the following sections. Figure 3 shows the main types of OBRNIT concepts. Figure 4 shows the main concepts and relationships of OBRNIT. Figure 5 shows the inspection task's main concepts and relationships. Some concepts are duplicated in Figures 4 and 5 to improve the readability of the figures. Color coding is used to group the concepts pertaining

to each of the four groups. However, the figures are simplified by adding the colors only to the main concepts of the ontology. The relationships between the concepts show how the ontology components are semantically interrelated. The types of relations used in the developed ontology are as follows: is, has, uses, affects, performs, causes, captures, has state, has time, has target, and measures (e.g., thermal camera measures temperature).

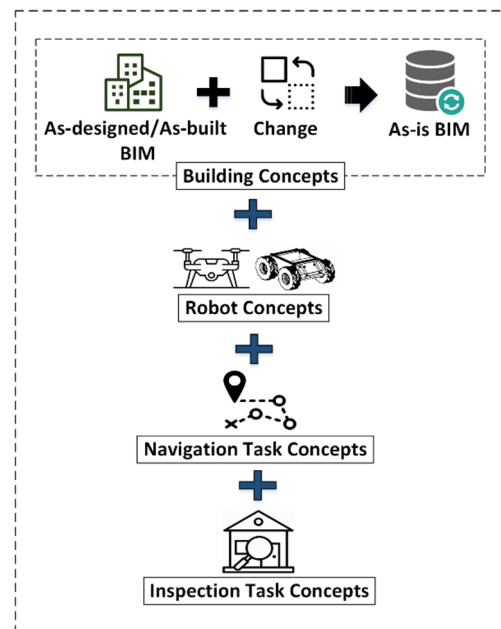


Figure 3. OBRNIT main types of concepts.

3.2.1. Robot Concepts

The robot concepts of OBRNIT cover the main functions of a robot along with the related knowledge of the inspection and navigation tasks. Declarative abstract knowledge about the tasks and environment should be encoded in the robot controller and used to determine proper actions for a specific task.

KnowRob ontology represents semantic models using object detection applied to the acquired point clouds enriched by encyclopedic, common-sense, and action-related knowledge [12]. From the BIM point of view, this ontology is primitive and does not provide full support for building elements. For example, the concept of a wall is only mentioned as a part of the edges of a region's surface and does not have dimensions, material, connectivity, type, etc. Walls may play a major role in inspection and navigation tasks because they define the boundaries of robots' movements or can be obstacles, or the main target of inspection. Other building elements, such as ceilings, columns, and windows, are not covered in KnowRob.

As shown in Figure 4, mobility and sensing are the two main functions of robots. The mismatches between the path found based on the non-updated BIM model (Section 3.2.2) and the as-is state of the surrounding environment (Section 3.2.3) will cause an obstacle for the robot's movement, and consequently its performance. Robot concepts cover basic attributes (e.g., type and size), robots' performance (e.g., movements and degrees of freedom (DOF)), robots' constraints (e.g., safety distance), and sensors for navigation and inspection tasks. The DOF defines the modes for the motion capability of the robot. The types of robots considered in OBRNIT are UAV and UGV. UGV refers to any type of crawling, climbing, and other ground-based robots. The movement of UAVs is in the 3D spaces of the building. However, UGVs move following the floors and may be able to climb the stairs. In this case, there are some constraints on the movement, such as the maximum height of a stair step that they can climb. Also, the flying movement of a UAV has constraints, which mainly depend on the size of the UAV.

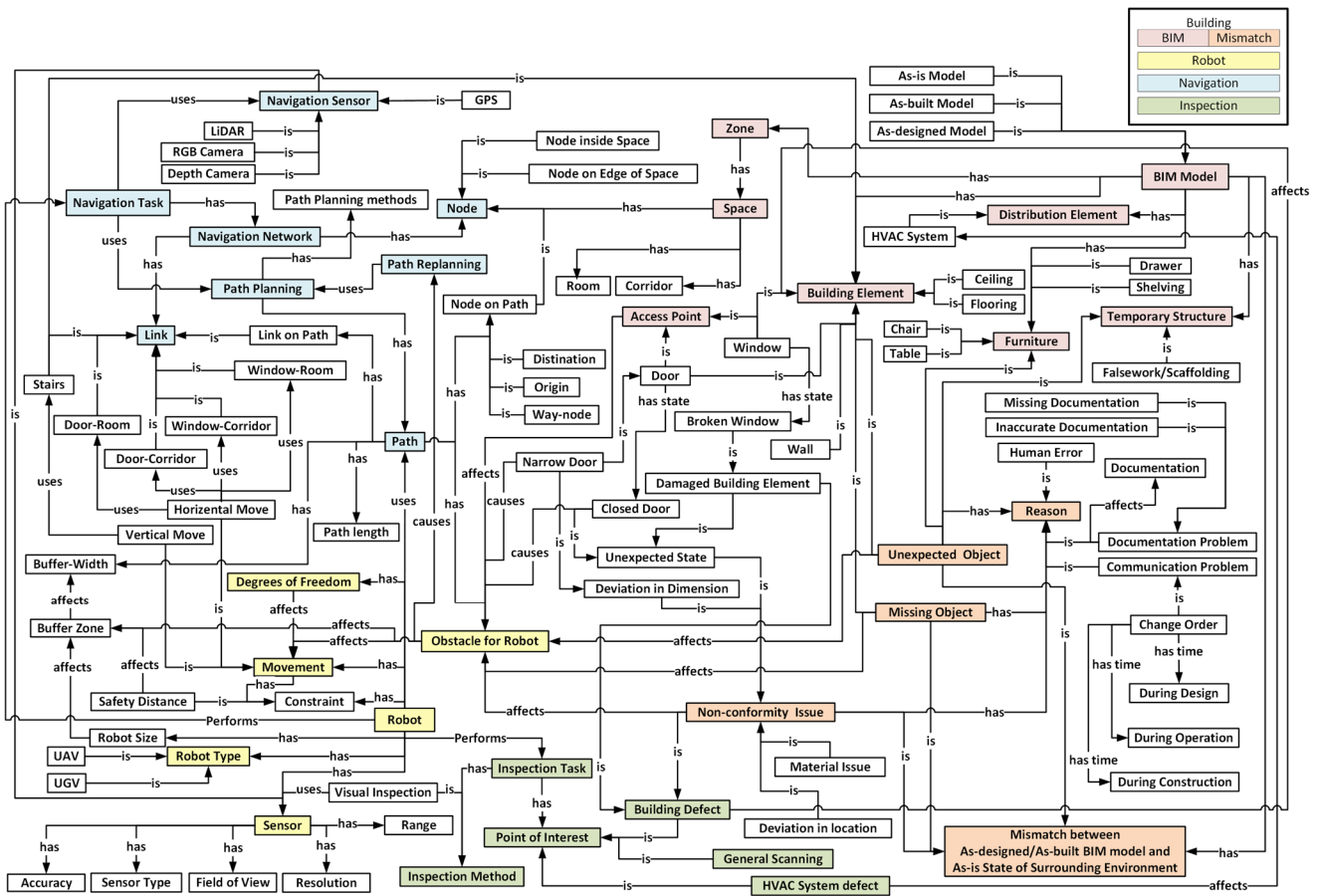


Figure 4. Main concepts and relationships of OBRNIT.

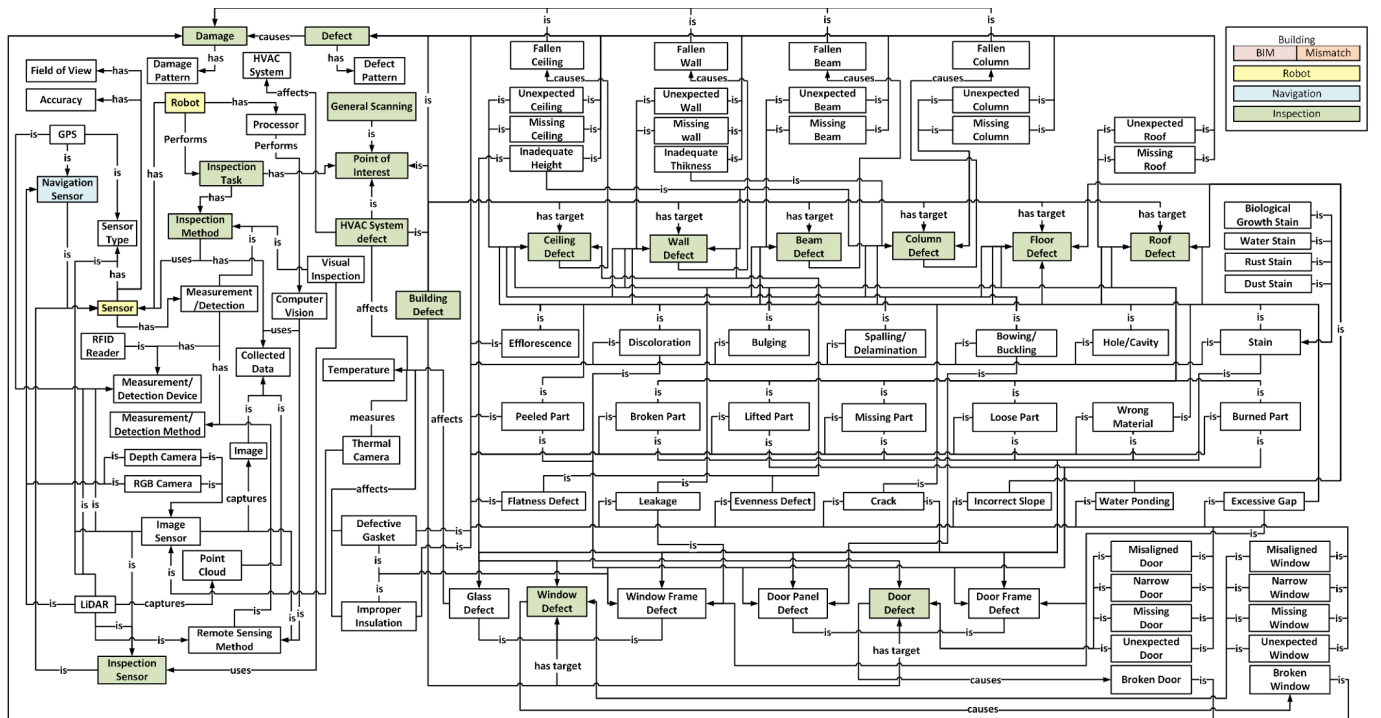


Figure 5. Inspection task main concepts and relationships.

Sensors can be used for inspection (e.g., RGB camera and thermal camera) and navigation purposes (e.g., depth camera and GPS). LiDAR and cameras are two different types of sensors. Cameras collect images, which can be RGB/depth/thermal images. LiDAR scanner is a remote sensing method which collects point clouds from the environment. The accuracy and field of the view of the robot's sensor, as well as its type, affect the robot's inspection performance. The concepts related to inspection tasks are explained in Section 3.2.4.

3.2.2. Building Concepts

The BIM model can provide information about the environment of the robotic inspection. Every building element that affects the robot navigation and inspection processes should be included in OBRNIT. As explained in Section 3, the integration process starts with integrating BEO. The required concepts, which are not included in BEO, are added from the ifcOWL ontology or defined based on the required concepts for robotic navigation and inspection. The process of integrating BIM concepts with OBRNIT aims to link the available BIM concepts with the developed OBRNIT concepts, including the related building concepts (e.g., BIM mismatch concepts), robot concepts, and inspection and navigation task concepts. Some research focused on robots that can open a closed door with specific access control or use a handle, knob, or button [98]. For example, Cobalt Access can open locked doors by using the door's access control reader [99]. However, passing through locked doors without human intervention is still one of the main issues for most of the robots. Figure 6 shows robot access control concepts in OBRNIT. The state of the door can be open or closed, locked or unlocked, mechanically locked, or electronically locked.

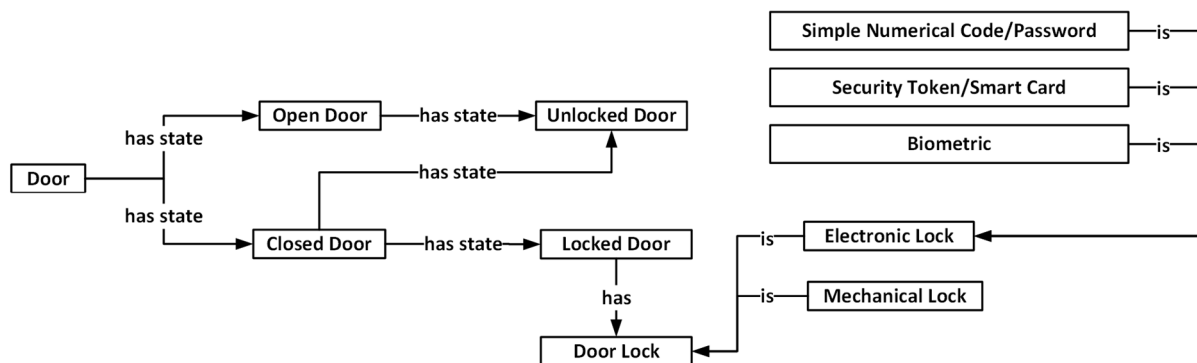


Figure 6. Robot access control concepts.

Table 4 shows examples of building concepts reused from BEO, BOT, and ifcOWL, and new concepts defined in OBRNIT. The building concepts of OBRNIT include the following: (1) Concepts reused from BEO ontology. (2) Concepts reused from BOT. (3) Concepts reused from ifcOWL: Some necessary concepts, which are not included in BEO (e.g., the furniture concept), are added from the ifcOWL ontology. HVAC elements are also added from the ifcOWL ontology in order to consider the HVAC system defects. (4) Concepts adopted from Building Management Systems (BMSs): Some concepts related to the state of the door are required for navigation purposes. These concepts are adopted from BMS. (5) New building concepts defined based on OBRNIT needs: These concepts include BIM mismatch concepts. In addition, the following relationships are defined to link the building-related concepts to the navigation and inspection concepts: (1) relationships to define the links between spaces for navigation paths (e.g., door/corridor), (2) relationships to define a BIM object as the point of interest of inspection, and (3) relationships to define obstacles or constraints for the robot's movement (e.g., a narrow door).

Table 4. Examples of reused, adopted, and new building concepts in OBRNIT.

Concept's Source	Example Concepts
Concepts reused from BEO	Beam, column, covering (ceiling and flooring), door, stair, wall, and window
Concepts reused from BOT	Space and zone
Concepts reused from ifcOWL	HVAC system
	Room and corridor
	Furniture (e.g., table and shelving)
Concepts adopted from BMS	Open door, closed door, locked door and unlocked door
	Access point
	Temporary structure (e.g., falsework/scaffolding)
	BIM model (as-designed, as-built, and as-is)
New building concepts	A mismatch between as-designed/as-built BIM and as-is state of the surrounding environment (missing objects, unexpected objects, and non-conformity issues), deviation in dimension, deviation in location, material issue, unexpected state, and damaged building element
	Mismatch reason (communication problem, documentation problem, or human error), change order, inaccurate documentation, and missing documentation

Furthermore, the mismatches between the as-designed or as-built BIM model and the as-is state of the surrounding environment should be semantically represented in OBRNIT. By implementing a BIM model of a building, all the information about the elements is available through this model. Identifying the potential types of mismatches is the first step to defining a logic-based robotic inspection system that can reduce delays and reworks. Having a rich semantic database about the spaces and building components can enhance the overall efficiency of the robot. Also, the information about the path has a major role when the goal is finding the optimal route and avoiding collisions with the existing barriers. Different spaces in the building can form different zones. Spaces (e.g., rooms) can be used to generate nodes for generating the path of the robot, which is explained in Section 3.2.3. The dimensions of a space can be used to define these nodes inside or on the edges of the space. The main building spaces for robot path planning are rooms, corridors, and stairs. The functionality of rooms and specifications of spaces can be different (e.g., security level for access to public/restricted rooms).

The mismatches between the information in the available BIM model and the reality cause navigation problems for robots. The preliminary model of the BIM at the design phase is called as-designed BIM. As-built BIM includes all the changes during the construction phase. As-is BIM includes the updated information of the facility and all the changes (e.g., repair, replacement, etc.) at the time of data collection. In some cases, the lack of adequate communication in the design phase, insufficient documentation, or the errors of the contractor can turn into unexpected results including information mismatches between the as-designed BIM model and the as-is state of the building. The same problem can occur in the operation phase, where renovation issues can cause mismatches between the non-updated as-built BIM and the as-is state of the building. The assumption in OBRNIT is that the path planning is based on a reference BIM model, but this model is not as-is and reliable. The semantic mismatch between the as-designed BIM model (or the as-built BIM model) and the as-is state of the surrounding environment could be caused by one of the following problems: (1) There is an object in the BIM model, which does not exist in reality. This problem can be the result of design changes during the construction phase (e.g., removing a door) where the changes are not applied in the BIM model. (2) There is an object in the building which is not included in the last updated BIM model. (3) There is a discrepancy between the BIM model and the actual building with respect to objects' attributes, such as location or dimensions. As shown in Figure 4, these problems that the robot can face

in a building are classified as missing objects, unexpected objects, and non-conformity issues. Each of these issues could be linked to fixed or mobile objects. For instance, building elements (e.g., access points) can be missing objects, and furniture and temporary structures (e.g., falsework) can be unexpected objects. Also, classes related to non-conformity should cover material issues, unexpected states (e.g., damaged building element or a closed door which is expected to be open), deviation in location or deviation in dimensions (e.g., narrow door), etc. As shown in Figure 4, each of the main mismatch entities has one or more causes and effects. For instance, some of the causes are human errors, documentation problems (e.g., change request was not documented), and communication problems during the different phases of AEC/FM. Each of these reasons causes a problem that can be described as an effect (e.g., obstacles for a robot). A narrow door (i.e., deviation in dimensions) or a closed door (i.e., different states from what is expected) are examples of non-conformity that can cause problems for a robot during its operation.

3.2.3. Navigation Concepts

The navigation task in OBRNIT refers to the act of performing navigation by the robot. As shown in Figure 4, navigation concepts cover the main information related to the path of the robot including nodes and links, which can be used for path planning. The navigation task has a network, and it uses the information of this network for path planning. Different types of navigation sensors can be used including GPS, LiDAR scanner, and depth camera. A LiDAR scanner can be used to support both the inspection task (Section 3.2.4) and the navigation task. The robot uses the path for performing the navigation task. A path has attributes including the length, direction, and buffer-width.

A node can be the origin or destination of a path, or a way-node on the path. Spaces (e.g., room and corridor) and access point elements of a building (e.g., doors and windows) can be the nodes of a path. For example, if a robot must move from a corridor to a room, the center point of the corridor is the origin node, the center point of the room is the destination node, and the door of the room is a way-node. The positions of the way-nodes vary based on the obstacles in the way of the robot. These obstacles may be unexpected objects detected by the robot as explained in Section 3.2.2. New links on the path connect these way-nodes to the origin and the destination nodes and each other [55]. Links connect the nodes and define the direction of the path. Examples of links are the links connecting a window to a room (in the case of UAV), a door to a corridor, or a door to a room based on the defined building elements and spaces as explained in Section 3.2.2. Links can be horizontal or vertical (e.g., stairs' links are vertical). Figure 7 shows a simple example, where Node 1 at the center of Room 1 is the origin node and Node 2 at the center of Room 2 is the destination node. Link 1 is the shortest link to connect the origin to the destination nodes; but it crosses two obstacles (i.e., the walls of the rooms). Nodes 3 to 7, which are way-nodes on the path, and the links between them are added to create an obstacle-free path (path A). As explained in Section 3.2.2, the states and dimensions of the access points (e.g., doors and windows) are important to enable the robot's movement over the path. For example, a closed or narrow door can be an obstacle to the robot's navigation.

3.2.4. Inspection Concepts

The inspection is the main task of the robot in OBRNIT and is mostly performed using vision sensors (e.g., LiDAR scanners and cameras). As explained in Section 3, the DOT concepts are integrated to link with the building defect concepts of OBRNIT. Examples of concepts reused from DOT are damage, damage pattern, documentation, and defect. In this section, the attributes of the inspection-related tasks of OBRNIT are defined based on common defects in buildings [100]. OBRNIT covers only the major types of defects related to ceiling, beam, column, wall, floor, roof, door, and window elements. However, it does not cover all the types of building defects. Building elements can have different types of defects that the robot can inspect based on their material. For example, concrete surfaces can have defects such as cracks, spalling, and efflorescence. Moreover, some

types of defects such as missing roofs can be detected after a disaster occurrence. As shown in Figure 5, the inspection task has an inspection method, which can be a visual inspection or a method for the measurement/detection of physical conditions (e.g., broken glass) or environmental conditions (e.g., temperature). The method of inspection is based on the sensor's measurement/detection and acquired datasets. Measurement/detection devices for inspection are radio-frequency ID (RFID) readers, image sensors (i.e., RGB and thermal cameras), and LiDAR scanners. RFID is a technology that uses radio frequencies to detect objects. RFID tags can be attached to separate object instances and linked with BIM information. Inspection using cameras produces images while inspection using LiDAR scanners produces point clouds. These images and point clouds can be used to detect surface defects, deformations, non-conforming elements, etc. The quality of LiDAR data are defined based on the two main parameters of density and accuracy [101]. The density of a point cloud is represented by the number of points in a specific area. The distance between the two points which are next to each other defines the point spacing. Computer vision methods can be used for anomaly detection on the collected data. Also, the information on computer vision methods can be used for obstacle detection and navigation tasks (Section 3.2.3).

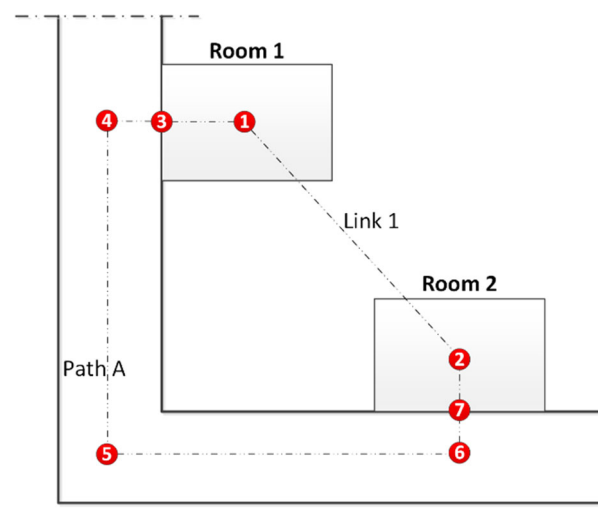


Figure 7. Example of using BIM for path planning.

Defects can cause damage to the building elements. Damage occurs when a defective element loses its function. For example, water leakage from the ceiling is a defect, which can cause damage to the ceiling elements over time. There are several causes for defect formation and damage occurrence. Defects and damages have various patterns and characteristics.

OBRNIT covers two main types of defects, including building defects and HVAC system defects. The point of interest of the inspection task is defined by the inspection purpose, which can be general scanning, inspecting mechanical systems (e.g., HVAC), or detecting building defects. General robotic scanning aims to update the BIM model or to collect data on a hazardous building, which is unsafe to inspect by human inspectors. The malfunctions of the HVAC system affect the environment's temperature and air quality. Defected HVAC elements or related building elements (e.g., improper insulation) can be evaluated by thermal cameras. In the case of inspecting building defects (e.g., surface/material defects), specific building elements (e.g., doors, walls, floors, etc.) are the points of interest, and each of them can be a target for the inspection task. For example, defective gaskets and improper insulation are some types of window frame defects, and the ceiling can be inspected for different types of defects such as leakage, stain, discoloration, bulging, spalling, delamination, and efflorescence. Some issues related to non-conformity can be considered as building defects, as discussed in Section 3.2.2. Furthermore, the

detected defects can be used to update the available BIM model to create an up-to-date as-is BIM model.

4. Evaluation

Ontology evaluation is a main step in ontology development, which refers to the process of evaluating if the developed ontology is correct and if it represents the main concepts and relationships [102]. Two evaluation methods are used for evaluating the usefulness of OBRNIT: (1) application-based evaluation and (2) qualitative criteria-based evaluation. The application-based evaluation is the evaluation of a developed ontology using a case study. This approach judges whether the ontology is suitable to perform the task and meets the objectives. However, it is not used to evaluate the design or the contents of the ontology [102]. On the other hand, the qualitative criteria-based evaluation approach is used to evaluate the ontology based on criteria such as clarity, coherence, consistency, correctness, and expandability. Consistency criteria are tested using the HermiT OWL Reasoner in the verification process [103]. The HermiT OWL Reasoner, which is based on the hypertableau algorithm, is used for identifying subsumption relationships and consistency evaluation [104]. The reasoner clarified some inconsistencies in the ontology. As described in Section 3.1, these results were utilized as feedback and input to Step 3 to fix the problems before going to the final step.

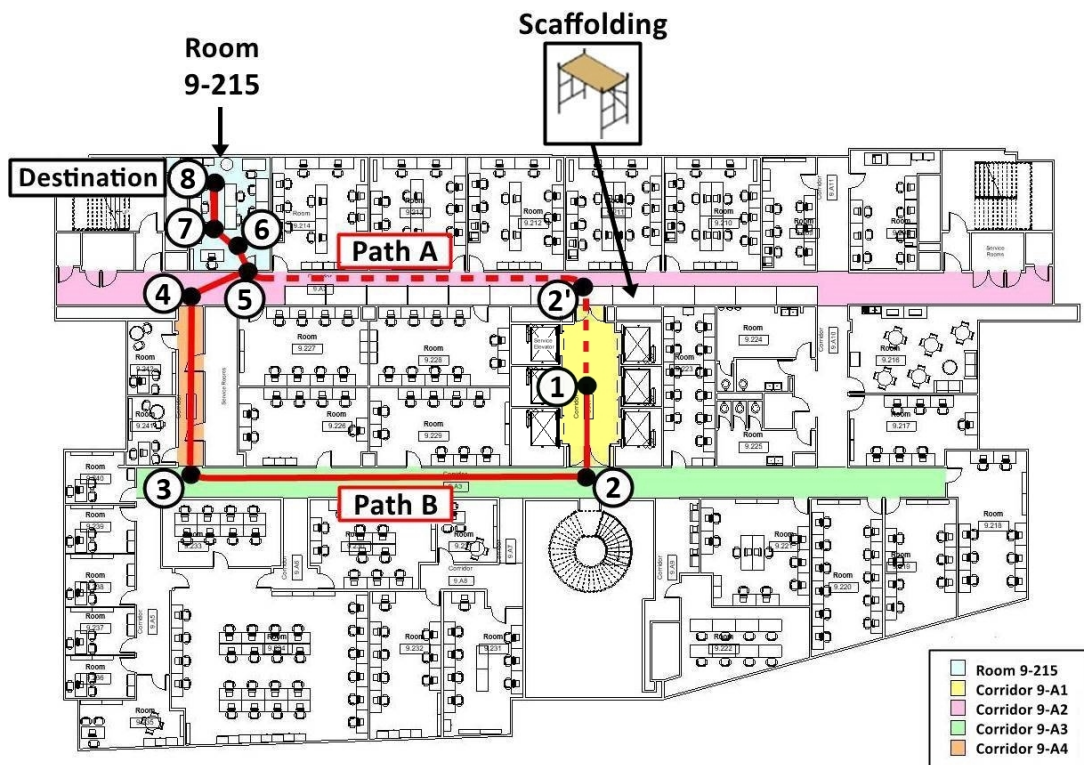
4.1. Case Study

This case study investigates the use of an inspection robot to locate a leakage in Room 9-215, situated on the ninth floor of a building at Concordia University (Figure 8). The primary objective is to demonstrate the applicability of the OBRNIT by utilizing specific information extracted from the BIM model of the building in conjunction with the capabilities of the employed inspection robot. For this case study, the robot has partial knowledge of the environment based on a non-updated BIM model. After defining the inspection point of interest as the leakage in the ceiling of Room 9-215, the robot navigates to reach this location to perform the inspection task. Path planning is conducted based on a reference as-built BIM model. Once at the inspection location, the robot utilizes an RGB camera to capture images of the ceiling. The specifications of the inspection task are outlined in Table 5.

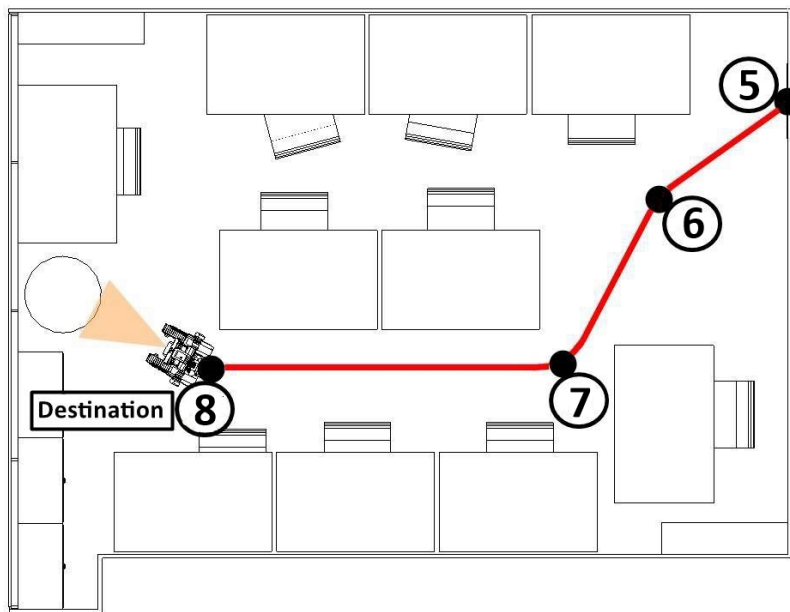
Table 5. Inspection task specifications based on concepts in OBRNIT.

Concept in OBRNIT	
Point of interest	Ceiling defect
Type of defect	Leakage
Inspection method	Measurement/detection
Measurement/detection device	RGB camera

The Mecabot Pro robot [105], an unmanned ground vehicle (UGV) with horizontal mobility, has been chosen for this case study as shown in Figure 9. The detailed specifications of the inspection robot are provided in Table 6. This robot is equipped with a LiDAR sensor, enabling comprehensive 360-degree scanning for object avoidance. The seamless integration of this LiDAR into the Mecabot platform ensures efficient mapping and navigation functionalities. Furthermore, the robot is equipped with a depth camera, providing detailed visual information. To explore immersive environments and capture rich visual experiences, the robot is enhanced with a VR (Virtual Reality) camera. This additional camera enables the robot to record and stream immersive 360-degree videos. To facilitate the robot's autonomous navigation and mapping, SLAM was used to continuously detect obstacles not available in the BIM model. The implementation of the robotic inspection case study was performed using the ROS [3].

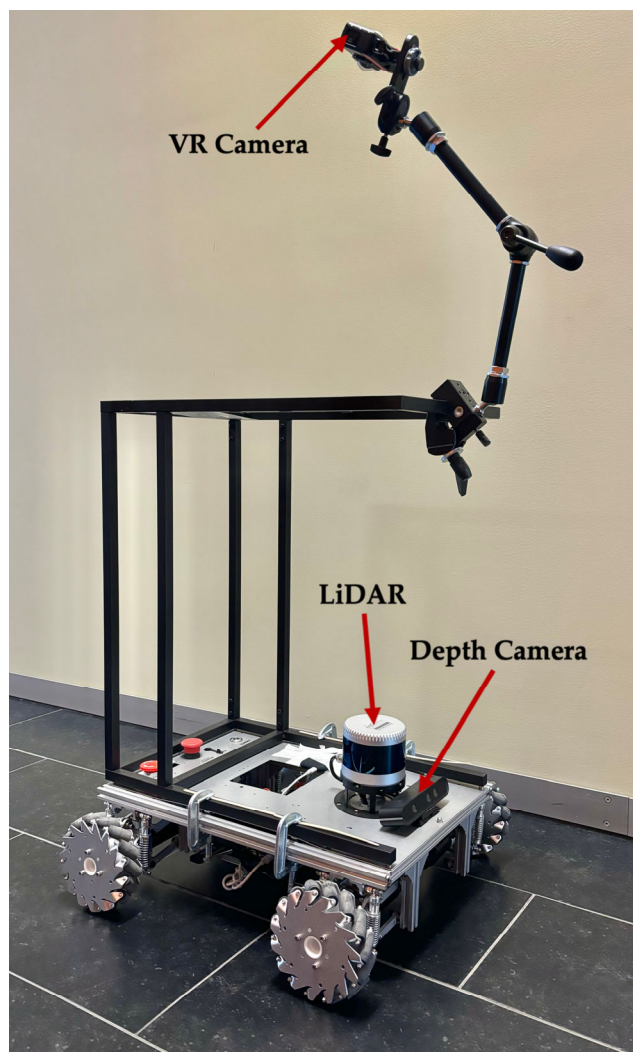


(a) BIM model of the 9th floor and corridor spaces for the path from the elevator to Room 9-215

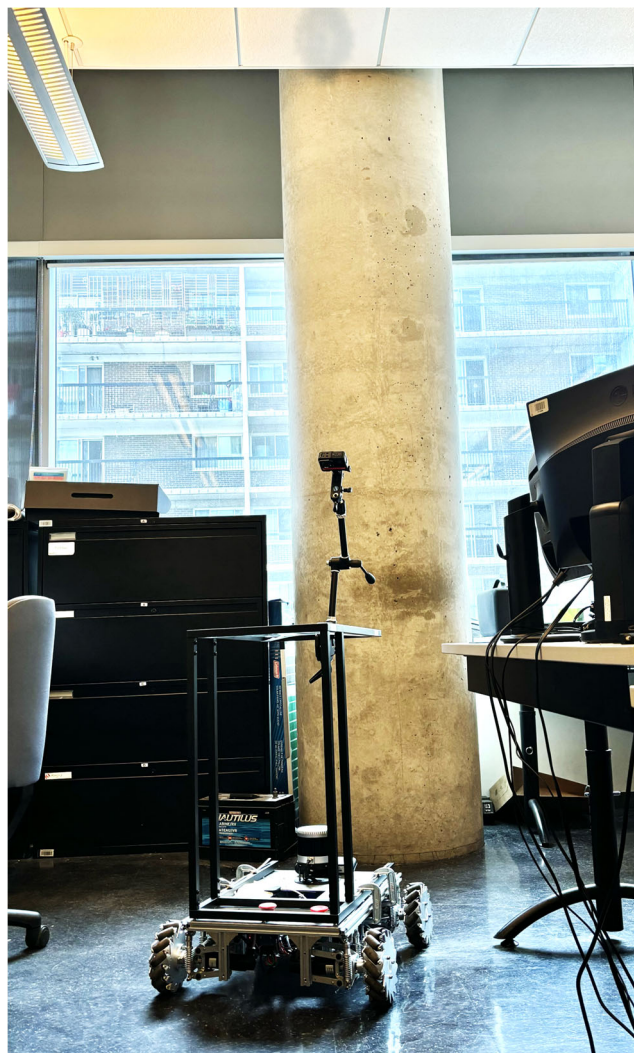


(b) Path of inspection robot in Room 9-215

Figure 8. Case study of using an inspection robot.



(a) Robot equipped with LiDAR, depth and VR cameras



(b) Robot collecting an image of leakage in the room

Figure 9. Mecabot Pro robot.

Table 6. Inspection robot main specifications.

Concept in OBRNIT		Specifications
Robot type (UGV)		Mecabot Pro
Movement		Horizontal movement
Sensor type	LS LiDAR (Leishen C16 3D)	360-degree scanning range and surroundings perception
	Depth camera (Orbbec Astra)	RGBD image capturing for a range of uses including gesture control, skeleton tracking, 3D scanning, and point cloud development
	VR camera (Insta360 ONE RS)	360-degree view for photos and video capturing
Degrees of freedom		3 degrees of freedom
Size	Length	58.1 cm
	Width	54.1 cm
	Height	22.5 cm

Table 7 presents examples of BIM-based information, including the objects in Room 9-215, the inspection point of interest, and the spaces/objects from the elevator on the ninth

floor to the door of Room 9-215. It is important to note that this table only contains the walls of Room 9-215 and does not show the other walls of the entire ninth floor.

Table 7. Examples of the IFC-based information of Room 9.215 and the spaces/objects outside the room.

	IfcEntity	Name	Tag	Concept in OBRNIT
Room 9-215	IfcColumn	M_Round Column: 610 mm Diameter	364991	Column
	IfcCovering	Compound Ceiling: 600 × 600 mm grid 2, white	378778	Ceiling (point of interest for leakage inspection)
	IfcCurtainWall	Curtain Wall: Storefront	363008	Curtain wall
	IfcDoor	M_Single-Flush: 0915 × 2134 mm: 379291	379291	Door
	IfcFurniture	M_Furniture_System-Standing_Desk-Rectangular:	372571	Table
	IfcFurniture	1500 × 750 mm	373006	
	IfcFurniture		373129	
	IfcFurniture		373192	
	IfcFurniture		373239	
	IfcFurniture		373486	
	IfcFurniture		373630	
	IfcFurniture		374087	
	IfcFurniture		374640	
	IfcFurniture		374723	
	IfcFurniture		376992	
	IfcFurniture	M_Chair—Executive	377394	
	IfcFurniture		377583	
	IfcFurniture		377646	
	IfcFurniture		377711	
	IfcFurniture		377776	
	IfcFurniture		377859	
	IfcFurniture		377916	
	IfcFurniture		377983	
	IfcFurniture		378050	
	IfcFurniture	M_Shelving: 1240 × 0305 × 1500 mm	368134	Shelving
	IfcFurniture		370460	
	IfcFurniture	M_Cabinet-File 4 Drawer: 1000 × 0457 mm	367042	Drawer
	IfcFurniture		367118	
	IfcFurniture		368542	
	IfcSlab	Floor: Generic Floor—400 mm	359802	Flooring
	IfcSpace	Room—9-215		Room
	IfcWallStandardCase	Basic Wall: Interior—138 mm Partition	360817	Wall
	IfcWallStandardCase		360875	
IfcWallStandardCase	360745			
IfcWallStandardCase	361005			
IfcWallStandardCase	361035			
IfcWallStandardCase	Basic Wall: steel—200 mm concrete masonry unit (CMU)		361214	
IfcBuildingElementProxy	Elevator: 1300 × 950 mm	263782	Transport element -elevator	
IfcBuildingElementProxy		263642		
IfcBuildingElementProxy		263853		
IfcBuildingElementProxy	Site_Scaffolding	321511	Falsework/scaffolding	
IfcSpace	Corridor—9-A1	-	Corridor	
IfcSpace	Corridor—9-A2	-		
IfcSpace	Corridor—9-A3	-		
IfcSpace	Corridor—9-A4	-		
IfcStair	Assembled Stair: 7" max riser 11" tread	258349	Stair	

Table 8 shows the navigation network and path planning concepts for the desired path. The origin node is in front of the elevators on the ninth floor, and the destination node is inside Room 9-215. The path is divided into two parts: (1) Horizontal movement from the ninth-floor elevator hall to Room 9-215 (Figure 8a). The shortest path (Path A) would involve using Corridors 9-A1 and 9-A2 (Nodes 1, 2, and 5). However, this path is obstructed by scaffoldings, which are used for a renovation project, creating an obstacle for the robot. As a result, the robot must follow a longer path (Path B) to reach the room. The robot obtains information about the scaffoldings from its sensing ability using LiDAR. After detecting the obstacle, the robot replans a new path (Path B). The corridors involved in Path B to reach Room 9-215 are Corridor 9-A3, Corridor 9-A4, and Corridor 9-A2, which contain Nodes 1, 2, 3, 4, and 5. (2) The second part of the path involves horizontal movement inside the room, from the door to the destination node (i.e., the inspection point of interest), as shown in Figure 8b. Sub-nodes 6 and 7 in the navigation path distinctly represent the robot's behavior in avoiding obstacles (i.e., a chair) using the LiDAR data. The total distance the robot traveled, considering both parts of the path, is 40.56 m.

Table 8. Navigation network and path-planning concepts.

Concept in OBRNIT			
Parts of the Path to Reach Inspection Point of Interest		Links Connecting Nodes	Obstacles for Robot
Horizontal path in corridors on the 9th floor	Path A	1-2'-5	Scaffoldings, walls, and door
	Path B	1-2-3-4-5	Walls and door
Horizontal path inside Room 9-215		5-6-7-8	Chairs and tables

The case study demonstrates that OBRNIT can answer all the competency questions (Table 3) and it covers all the concepts necessary for the planning of the robotic building navigation and inspection. Integrating the mobility characteristics of the robot and the knowledge about the surrounding environment have been used to help the robot define the appropriate path based on the robot type and constraints and meet the requirements for the inspection task. The ontology has been used to help select a robot with suitable sensors for the navigation and inspection tasks. Moreover, the robot benefits from the BIM model to define the path based on defining the nodes and links of the path. In addition, the BIM model helps the robot locate the inspection object. The case study shows how several concepts are extracted from OBRNIT.

4.2. Criteria-Based Evaluation

A survey was conducted to evaluate the adequacy of the semantic representation of the concepts and relationships of OBRNIT. The survey includes eight questions, which are related to the different components of OBRNIT. These questions reflect the coverage of the concepts and semantic relationships between the classes, and aim to measure the clarity and comprehensiveness of OBRNIT. The first question was about the respondents' information. The second question was about BIM and its benefits for inspection robots. The third and fourth questions considered the clarity and comprehensiveness of the main concepts of OBRNIT. The fifth and sixth questions were about the clarity and comprehensiveness of the inspection part of OBRNIT. The seventh question was about a statement related to the complexity of interactions between components in OBRNIT. Finally, the last question considered OBRNIT's capability for system development. Figures 3–5 and 8 were provided in the survey to present the background and some details of OBRNIT. A five-point Likert scale is used to obtain the quantitative values of the answers.

The survey was sent to 105 internationally recognized experts selected based on their knowledge of BIM, construction, and robotic inspection. A total of 33 individuals participated in the survey (response rate of 31%). The 33 respondents have a total of

117 years of experience in robotics, computer science, and information systems, and a total of 225 years of experience in BIM, construction, and inspection research.

Table 9 lists the results of the survey answers. For Q2, the respondents strongly agreed (36.36%) or agreed (51.52%) that BIM extends the declarative knowledge of the environment for the cognitive robot's performance during navigation tasks. The answers for Q3 indicate that the main concepts and the relationships of OBRNIT are very clear (9.38%), clear (56.25%), somewhat clear (25%), and not so clear (6.25%). The answers to Q4 about the comprehensiveness of the main concepts and the relationships in OBRNIT indicate that they are very comprehensive (15.63%), comprehensive (59.38%), and somewhat comprehensive (21.88%). Q5 indicates that the inspection task concepts are very clear (15.63%), clear (59.38%), and somewhat clear (15.63%). The answers to Q6 about the comprehensiveness of the inspection task concepts were very comprehensive (18.75%), comprehensive (53.13%), somewhat comprehensive (21.88%), and not comprehensive (3.13%). For Q7, the respondents strongly agreed (21.88%) or agreed (68.75%) that complex declarative knowledge of OBRNIT is useful in clarifying the context of knowledge. Q8, which is related to the usage of OBRNIT for the development of robotic navigation and inspection systems, gained responses that strongly agreed (28.13%), agreed (65.63%), and neither agreed nor disagreed (6.25%).

Table 9. Distribution of the responses.

Q No	Ave.	SD	Results					
Q2	1.75	0.65	Strongly agree	Agree	Neither agree nor disagree	Disagree	Strongly disagree	No answer
			36.36%	51.52%	12.12%	0%	0%	0%
Q3	2.29	0.72	Very clear	Clear	Somewhat clear	Not so clear	Not clear at all	No answer
			9.38%	56.25%	25%	6.25%	0%	3.13%
Q4	2.06	0.61	Very comprehensive	Comprehensive	Somewhat comprehensive	Not comprehensive	Missing lots of concepts	No answer
			15.63%	59.38%	21.88%	0%	0%	3.13%
Q5	2	0.58	Very clear	Clear	Somewhat clear	Not so clear	Not clear at all	No answer
			15.63%	59.38%	15.63%	0%	0%	9.38%
Q6	2.09	0.73	Very comprehensive	Comprehensive	Somewhat comprehensive	Not comprehensive	Missing lots of concepts	No answer
			18.75%	53.13%	21.88%	3.13%	0%	3.13%
Q7	1.87	0.54	Strongly agree	Agree	Neither agree nor disagree	Disagree	Strongly disagree	No answer
			21.88%	68.75%	9.38%	0%	0%	0%
Q8	1.78	0.54	Strongly agree	Agree	Neither agree nor disagree	Disagree	Strongly disagree	No answer
			28.13%	65.63%	6.25%	0%	0%	0%

The feedback from the survey is largely on the positive side. For Q3 about the clarity of OBRNIT, the explanation of the partially negative evaluation could be that this question only provided the overall condensed figure of OBRNIT (Figure 4) without any explanation. The reason for not providing the explanation is to keep the time needed for answering the survey within a reasonable limit (less than 30 min). This was a compromise aiming to increase the number of responses to the survey.

The respondents' comments were also very positive in general, and some suggestions for improvement were provided. Some examples of comments are as follows: for Q2, "BIM

can extend the knowledge of path optimality, safety, and feasibility as it can provide the basic geometric information necessary for localization and pathfinding"; for Q6, "Other types of defects could be added". These comments will be considered in our future work.

5. Results and Discussion

This paper developed an integrated ontology, called OBRNIT, to extend BIM applications for robotic navigation and inspection tasks. There are 386 classes, 45 relationships, 52 attributes, and 8 individuals in OBRNIT. OBRNIT comprises high-level knowledge of the concepts and relationships related to buildings, robots, and navigation and inspection tasks. BIM is considered as a reference that is integrated with the knowledge model. The application of OBRNIT was investigated in a case study. In addition, a survey was designed and conducted to evaluate the semantic representation of OBRNIT. The evaluation demonstrates that OBRNIT covers the domain's concepts and relationships up to the point that satisfies the domain experts. Based on the evaluation, OBRNIT was able to give a clear understanding of the concepts and relationships in the domain, and it can be applied for developing robotic inspection systems. OBRNIT can be used as a first step towards logic-based inspection, which can help robots perform inspection tasks autonomously without the help of human judgment. It is difficult to prove that an ontology enables additional capabilities for systems that would not be possible without it [4]. However, using a central ontological knowledgebase can facilitate the development of robotic inspection systems. The integration of abstract knowledge with robot action-related procedural knowledge, as suggested by [15], could make tasks more easily executable. Furthermore, developing a planning language system for reasoning over the ontological knowledgebase for plan execution, as studied by [52,89], could enhance the practical application of OBRNIT. Linking OBRNIT with other available ontologies, such as the Sensor Ontology [106], could further extend its capabilities.

6. Contributions, Conclusions, and Future Work

This paper has successfully developed and evaluated OBRNIT, an integrated ontology for robotic navigation and inspection tasks based on BIM. The ontology provides a comprehensive framework for understanding and representing the complex relationships between buildings, robots, and inspection tasks. OBRNIT's potential to enhance the development of autonomous robotic inspection systems is significant, offering a foundation for future advancements in this field. OBRNIT is expected to provide the following benefits: (1) OBRNIT can help system engineers involved in developing robotic inspection systems by identifying the different concepts and relationships about robotic inspection and navigation tasks based on BIM information; (2) capturing the essential information from BIM can help to develop a seamless knowledge model to cover the missing parts of BIM; (3) using ontological knowledge can help overcome the complexity in interactions between the components in the robotic inspection system.

This study has several limitations: (1) This paper focused on developing declarative knowledge, including conceptual and geometric information related to inspection and navigation tasks, and does not address the low-level path planning and the problem of SLAM. (2) The development of as-is BIM models, which are necessary for robotic inspection and obstacle avoidance, remains challenging for existing buildings. (3) Additional studies are needed to demonstrate how the ontology enhances robotic inspection system capabilities beyond the current methods. Future work will focus on further development and implementation of OBRNIT to integrate it with low-level robotic capabilities to make the robot more autonomous. This includes (1) combining abstract knowledge with robot action-related procedural knowledge to make tasks executable; (2) developing a planning language system for reasoning over the ontological knowledgebase for plan execution; (3) extending OBRNIT by linking it with other available ontologies and expanding the concepts to other types of defects; (4) developing methods for the automatic updating of BIM models throughout the different stages of building lifecycles, addressing the scarcity of

as-built or reliable as-is BIM models in existing buildings. These directions aim to enhance the practical applicability of OBRNIT and address the current limitations in the field of robotic inspection and navigation.

Author Contributions: Conceptualization, F.B., M.N. and A.H.; methodology, F.B. and A.T.; software, F.B.; validation, F.B. and A.H.; formal analysis, F.B. and A.H.; investigation, F.B.; resources, F.B., M.N., A.T. and A.H.; data curation, F.B.; writing—original draft preparation, F.B.; writing—review and editing, F.B. and A.H.; visualization, F.B.; supervision, A.H.; project administration, A.H.; funding acquisition, A.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original data presented in the study are openly available at [<https://github.com/OBRNIT/OBRNIT>] (accessed on 18 July 2024).

Conflicts of Interest: Author Majid Nasrollahi was employed by the company AIRM Consulting Co, author Alhusain Taher was employed by the company eStructure Data Centers. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

List of Abbreviations

Abbreviation	Description
AEC/FM	Architecture, engineering, construction, and facilities management
AUV	Autonomous underwater vehicles
BEO	Building Element Ontology
BIM	Building information modeling
BIMSO	BIM Shared Ontology
BMS	Building Management System
BOT	Building Topology Ontology
CPL	CRAM Plan Language
CRAM	Cognitive Robot Abstract Machine
DDM	Defect data management
DOF	Degrees of freedom
DOT	Damage Topology Ontology
FOG	Ontology for Geometry Formats
GPS	Global Positioning System
HVAC	Heating, Ventilation, and Air Conditioning
IDEF	Integrated Definition
IFC	Industry Foundation Classes
LCA	Life Cycle Assessment
LiDAR	Light detection and ranging
LOAM	Lidar Odometry and Mapping
MEP	Mechanical, Electrical, and Plumbing
O&M	Operation and maintenance
OBRNIT	Ontology for BIM-based robotic navigation and inspection tasks
OMG	Ontology for Managing Geometry
OPM	Ontology for Property Management
OWL	Web ontology language
RDF	Resource description framework
RFID	Radio Frequency Identification
RGB	Red, Green, and Blue
ROS	Robot Operating System
SLAM	Simultaneous Localization and Mapping
UAV	Unmanned aerial vehicle
UGV	Unmanned ground vehicle
UML	Unified Modeling Language
VR	Virtual Reality

References

1. Balaguer, C.; Gimenez, A.; Abderrahim, C.M. ROMA robots for inspection of steel based infrastructures. *Ind. Robot.* **2002**, *29*, 246–251. [[CrossRef](#)]
2. Kim, D.; Goyal, A.; Newell, A.; Lee, S.; Deng, J.; Kamat, V.R. Semantic relation detection between construction entities to support safe human-robot collaboration in construction. In Proceedings of the ASCE International Conference on Computing in Civil Engineering: Data, Sensing, and Analytics, Atlanta, GA, USA, 17–19 June 2019; ASCE: Reston, VA, USA, 2019; pp. 265–272.
3. Open Robotics, ROS. 2020. Available online: <https://www.ros.org/> (accessed on 7 July 2020).
4. Saïgol, Z.; Wang, M.; Ridder, B.; Lane, D.M. The Benefits of Explicit Ontological Knowledge-Bases for Robotic Systems. In Proceedings of the 16th Annual Conference on Towards Autonomous Robotic Systems, Liverpool, UK, 8–10 September 2015; Springer: Liverpool, UK, 2015; pp. 229–235.
5. Lim, G.H.; Suh, I.H.; Suh, H. Ontology-based unified robot knowledge for service robots in indoor environments. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2010**, *41*, 492–502. [[CrossRef](#)]
6. BuildingSMART. Industry Foundation Classes Release 4.1. 2018. Available online: https://standards.buildingsmart.org/IFC/RELEASE/IFC4_1/FINAL/HTML/ (accessed on 1 July 2020).
7. Zhang, J.; Singh, S. LOAM: Lidar Odometry and Mapping in Real-time. *Robot. Sci. Syst.* **2014**, *2*, 401–416.
8. Akinci, B.; Boukamp, F. Representation and integration of as-built information to IFC based product and process models for automated assessment of as-built conditions. In Proceedings of the 19th ISARC International Symposium on Automation and Robotics in Construction, Washington, DC, USA, 23–25 September 2002; pp. 543–550.
9. Rasmussen, M.H.; Lefrançois, M.; Schneider, G.F.; Pauwels, P. BOT: The building topology ontology of the W3C linked building data group. *Semant. Web* **2020**, *12*, 143–161. [[CrossRef](#)]
10. Hamdan, A.H.; Bonduel, M.; Scherer, R.J. An ontological model for the representation of damage to constructions. In Proceedings of the 7th Linked Data in Architecture and Construction Workshop, London, UK, 19–21 June 2019.
11. Niknam, M.; Karshenas, S. A shared ontology approach to semantic representation of BIM data. *Autom. Constr.* **2017**, *80*, 22–36. [[CrossRef](#)]
12. Tenorth, M.; Beetz, M. KnowRob—Knowledge processing for autonomous personal robots. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10–15 October 2009.
13. Lattanzi, D.; Miller, G. Review of robotic infrastructure inspection systems. *J. Infrastruct. Syst.* **2017**, *23*, 04017004. [[CrossRef](#)]
14. Brunner, S.; Kucera, M.; Waas, T. Ontologies used in robotics: A survey with an outlook for automated driving. In Proceedings of the IEEE International Conference on Vehicular Electronics and Safety (ICVES), Vienna, Austria, 27–28 June 2017; pp. 81–84.
15. Stulp, F.; Beetz, M. Combining declarative, procedural, and predictive knowledge to generate, execute, and optimize robot plans. *Robot. Auton. Syst.* **2008**, *56*, 967–979. [[CrossRef](#)]
16. Bahreini, F.; Hammad, A. Towards an ontology for BIM-based robotic navigation and inspection tasks. In Proceedings of the 38th International Symposium on Automation and Robotics in Construction, Dubai, United Arab Emirates, 2–4 November 2021.
17. Artus, M.; Koch, C. State of the art in damage information modeling for RC bridges—A literature review. *Adv. Eng. Inform.* **2020**, *46*, 101171. [[CrossRef](#)]
18. Tayeh, B.A.; Maqsoom, A.; Aisheh, Y.I.A.; Almanassra, M.; Salahuddin, H.; Qureshi, M.I. Factors affecting defects occurrence in the construction stage of residential buildings in Gaza Strip. *SN Appl. Sci.* **2020**, *2*, 167. [[CrossRef](#)]
19. Kim, M.K.; Cheng, J.C.; Sohn, H.; Chang, C.C. A framework for dimensional and surface quality assessment of precast concrete elements using BIM and 3D laser scanning. *Autom. Constr.* **2015**, *49*, 225–238. [[CrossRef](#)]
20. Park, C.S.; Lee, D.Y.; Kwon, O.S.; Wang, X. A framework for proactive construction defect management using BIM, augmented reality and ontology-based data collection template. *Autom. Constr.* **2013**, *33*, 61–71. [[CrossRef](#)]
21. Tekin, H.; Yilmaz, I.; Koc, K.; Atabay, Ş.; Gürgün, A. Identification of Defective Construction Works during Building Inspections. *Proc. Int. Struct. Eng. Constr.* **2023**, *10*, QUA-01-1–QUA-01-5. [[CrossRef](#)]
22. Ding, L.; Li, K.; Zhou, Y.; Love, P.E. An IFC-inspection process model for infrastructure projects: Enabling real-time quality monitoring and control. *Autom. Constr.* **2017**, *84*, 96–110. [[CrossRef](#)]
23. Melo, R.R.S.; Costa, D.B.; Alvares, J.S.; Irizarry, J. Applicability of unmanned aerial system (UAS) for safety inspection on construction sites. *Saf. Sci.* **2017**, *98*, 174–185. [[CrossRef](#)]
24. Phung, M.D.; Quach, C.H.; Dinh, T.H.; Ha, Q. Enhanced discrete particle swarm optimization path planning for UAV vision-based surface inspection. *Autom. Constr.* **2017**, *81*, 25–33. [[CrossRef](#)]
25. Bolourian, N.; Hammad, A. LiDAR-equipped UAV path planning considering potential locations of defects for bridge inspection. *Autom. Constr.* **2019**, *117*, 103250. [[CrossRef](#)]
26. Lundeen, K.M.; Kamat, V.R.; Menassa, C.C.; McGee, W. Autonomous motion planning and task execution in geometrically adaptive robotized construction work. *Autom. Constr.* **2019**, *100*, 24–45. [[CrossRef](#)]
27. Freimuth, H.; Müller, J.; König, M. Simulating and Executing UAV-Assisted Inspections on Construction Sites. In Proceedings of the 34th ISARC International Symposium on Automation and Robotics in Construction, Taipei, Taiwan, 28 June 2017.
28. Bahreini, F.; Hammad, A. Dynamic graph CNN based semantic segmentation of concrete defects and as-inspected modeling. *Autom. Constr.* **2024**, *159*, 105282. [[CrossRef](#)]
29. Bortolini, R.; Forcada, N. Building inspection system for evaluating the technical performance of existing buildings. *J. Perform. Constr. Facil.* **2018**, *32*, 04018073. [[CrossRef](#)]

30. Valença, J.; Morin, K.; Jouen, N.; Olivo, N.; Torres-Gonzalez, M.; Mendes, M.P.; Silva, A. Feeling-BIM: A digital model to support maintenance decisions, based on automatic inspection and dwellers' feelings. *J. Build. Eng.* **2024**, *87*, 108937. [[CrossRef](#)]
31. Metni, N.; Hamel, T. A UAV for bridge inspection: Visual servoing control law with orientation limits. *Autom. Constr.* **2007**, *17*, 3–10. [[CrossRef](#)]
32. Chen, W.; Yabuki, N.; Fukuda, T.; Michikawa, T.; Motamedi, A. Development of product model for harbor structures degradation. In Proceedings of the 2nd International Conference on Civil and Building Engineering Informatics (ICCBIE), Tokyo, Japan, 22–24 April 2015.
33. Motamedi, A.; Yabuki, N.; Fukuda, T. Extending BIM to include defects and degradations of buildings and infrastructure facilities. In Proceedings of the 3rd International Conference on Civil and Building Engineering Informatics, Taipei, Taiwan, 19–21 April 2017.
34. Hammad, A.; Motamedi, A.; Yabuki, N.; Taher, A.; Bahreini, F. Towards unified ontology for modeling lifecycle inspection and repair information of civil infrastructure systems. In Proceedings of the 17th ICCBE International Conference on Computing in Civil and Building Engineering, Tampere, Finland, 5–7 June 2018.
35. Kasireddy, V.; Akinci, B. Towards the integration of inspection data with bridge information models to support visual condition assessment. In Proceedings of the ASCE International Workshop on Computing in Civil Engineering, Austin, TX, USA, 21–23 June 2015.
36. Ekba, S.; Borovkova, A.; Nikolenko, D.; Koblyuk, D. A systematic approach to technical inspection of construction projects. *E3S Web Conf.* **2023**, *402*, 07003. [[CrossRef](#)]
37. Choi, M.; Kim, S.; Kim, S. Semi-automated visualization method for visual inspection of buildings on BIM using 3D point cloud. *J. Build. Eng.* **2024**, *81*, 108017. [[CrossRef](#)]
38. Mohamed, M.; Tran, D. Approach for Estimating Inspection Staffing Needs for Highway Construction Projects. *Transp. Res. Rec.* **2023**, *2677*, 697–707. [[CrossRef](#)]
39. Furtner, P.; O'Brien, P. Automated Creation of an IFC-4 Compliant Damage Model from a Digital Inspection Supported by AI. *ce/papers* **2023**, *6*, 1366–1372. [[CrossRef](#)]
40. Chen, Z.; Chen, L.; Zhou, X.; Huang, L.; Sandanayake, M.; Yap, P.S. Recent technological advancements in BIM and LCA integration for sustainable construction: A review. *Sustainability* **2024**, *16*, 1340. [[CrossRef](#)]
41. Tan, Y.; Xu, W.; Chen, P.; Zhang, S. Building defect inspection and data management using computer vision, augmented reality, and BIM technology. *Autom. Constr.* **2024**, *160*, 105318. [[CrossRef](#)]
42. Zverovich, V.; Mahdjoubi, L.; Boguslawski, P.; Fadli, F. Analytic prioritization of indoor routes for search and rescue operations in hazardous environments. *Comput. Aided Civ. Infrastruct. Eng.* **2017**, *32*, 727–747. [[CrossRef](#)]
43. Shin, H.; Cha, H. Proposing a Quality Inspection Process Model Using Advanced Technologies for the Transition to Smart Building Construction. *Sustainability* **2023**, *15*, 815. [[CrossRef](#)]
44. Liang, H.; Lee, S.; Bae, W.; Kim, J.; Seo, S. Towards UAVs in Construction: Advancements, Challenges, and Future Directions for Monitoring and Inspection. *Drones* **2023**, *7*, 202. [[CrossRef](#)]
45. Lundeen, K.M.; Kamat, V.R.; Menassa, C.C.; McGee, W. Scene understanding for adaptive manipulation in robotized construction work. *Autom. Constr.* **2017**, *82*, 16–30. [[CrossRef](#)]
46. Carra, G.; Argiolas, A.; Bellissima, A.; Nicolini, M.; Ragaglia, M. Robotics in the construction industry: State of the art and future opportunities. In Proceedings of the 35th ISARC on Automation and Robotics in Construction, Berlin, Germany, 20–25 July 2018; pp. 1–8.
47. Dormehl, L. 98 Percent of Construction Projects Go over Budget. These Robots Could Fix That. 2018. Available online: <https://www.digitaltrends.com/cool-tech/doxel-construction-monitoring-robots/> (accessed on 20 January 2020).
48. Lin, T.H.; Chiang, P.C.; Putranto, A. Multispecies hybrid bioinspired climbing robot for wall tile inspection. *Autom. Constr.* **2024**, *164*, 105446. [[CrossRef](#)]
49. Halder, S.; Afsari, K. Robots in Inspection and Monitoring of Buildings and Infrastructure: A Systematic Review. *Appl. Sci.* **2023**, *13*, 2304. [[CrossRef](#)]
50. Patil, S.; Admuthe, V.; Patil, R.; Bhokare, N.; Desai, A.; Chhachwale, S.; Nikam, P.; Desai, D. Construction Site Inspection by Using Drone or UAV. *Int. J. Eng. Appl. Sci. Technol.* **2023**, *7*, 101–103. [[CrossRef](#)]
51. Pu, H.; Yang, X.; Li, J.; Guo, R.; Li, H. AutoRepo: A general framework for multi-modal LLM-based automated construction reporting. *arXiv* **2023**, arXiv:2310.07944.
52. Beetz, M.; Mösenlechner, L.; Tenorth, M. CRAM—A Cognitive Robot Abstract Machine for everyday manipulation in human environments. In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, 18–22 October 2010.
53. Wong, M.O.; Lee, S. A Technical Review on Developing BIM-Oriented Indoor Route Planning. In Proceedings of the International Conference on In Computing in Civil Engineering: Visualization, Information Modeling, and Simulation, Atlanta, GA, USA, 17–19 June 2019; ASCE: Reston, VA, USA, 2019; pp. 336–342.
54. Belsky, M.; Sacks, R.; Brilakis, I. Semantic enrichment for building information modeling. *Comput. Aided Civ. Infrastruct. Eng.* **2016**, *31*, 261–274. [[CrossRef](#)]
55. Lin, Y.H.; Liu, Y.S.; Gao, G.; Han, X.G.; Lai, C.Y.; Gu, M. The IFC-based path planning for 3D indoor spaces. *Adv. Eng. Inform.* **2013**, *27*, 189–205. [[CrossRef](#)]

56. Boguslawski, P.; Mahdjoubi, L.; Zverovich, V.; Fadli, F. Automated construction of variable density navigable networks in a 3D indoor environment for emergency response. *Autom. Constr.* **2016**, *72*, 115–128. [CrossRef]
57. Hou, Y.; Ju, Y. A Method for Robot Indoor Path Planning Using BIM Based on A* Algorithm. In Proceedings of the International Conference on Electronic Technology, Communication, and Information (ICETCI), Changchun, China, 26–28 May 2023.
58. Liu, J.; Chen, X.; Xiao, J.; Lin, S.; Zheng, Z.; Lu, H. Hybrid Map-Based Path Planning for Robot Navigation in Unstructured Environments. In Proceedings of the International Conference on Intelligent Robots and Systems (IROS), Detroit, MI, USA, 1–5 October 2023; pp. 2216–2223.
59. Zhai, R.; Zou, J.; Gan, V.J.; Han, X.; Wang, Y.; Zhao, Y. Semantic enrichment of BIM with IndoorGML for quadruped robot navigation and automated 3D scanning. *Autom. Constr.* **2024**, *166*, 105605. [CrossRef]
60. Gruber, T.R. Toward principles for the design of ontologies used for knowledge sharing. *Int. J. Hum. Comput. Stud.* **1995**, *43*, 907–928. [CrossRef]
61. Gašević, D.; Djuric, D.; Devedžić, V. *Model Driven Engineering and Ontology Development*; Springer Science and Business Media: Berlin/Heidelberg, Germany, 2009.
62. Stanford University, Stanford Center for Biomedical Informatics Research. 2019. Available online: <https://protege.stanford.edu/about.php> (accessed on 2 January 2019).
63. Staab, S.; Maedche, A. Ontology engineering beyond the modeling of concepts and relations. In Proceedings of the ECAI Workshop on Ontologies and Problem-Solving Methods, Berlin, Germany, 21–22 August 2000.
64. McGuinness, D.L.; Harmelen, F. OWL Web Ontology Language Overview. 2004. Available online: https://www.researchgate.net/publication/200034408_OWL_Web_Ontology_Language---Overview (accessed on 10 June 2022).
65. BuildingSMART. ifcOWL. 2019. Available online: <https://technical.buildingsmart.org/standards/ifc/ifc-formats/ifcowl/> (accessed on 22 July 2020).
66. Cacciotti, R.; Blaško, M.; Valach, J. A diagnostic ontological model for damages to historical constructions. *J. Cult. Herit.* **2015**, *16*, 40–48. [CrossRef]
67. Pauwels, P. Building Element Ontology. 2018. Available online: <https://pi.pauwel.be/voc/buildingelement/index-en.html> (accessed on 26 February 2021).
68. Pauwels, P. Distribution Element Ontology. 2019. Available online: <https://pi.pauwel.be/voc/distributionelement/index-en.html> (accessed on 27 February 2021).
69. Rasmussen, M.; Lefrançois, M.; Bonduel, M.; Hviid, C.; Karlshøj, J. OPM: An ontology for describing properties that evolve over time. In Proceedings of the 6th Linked Data in Architecture and Construction Workshop, London, UK, 19–21 June 2018.
70. Wagner, A.; Bonduel, M.; Pauwels, P.; Uwe, R. Relating geometry descriptions to its derivatives on the web. In Proceedings of the European Conference on Computing in Construction, Crete, Greece, 10–12 July 2019.
71. Bonduel, M.; Wagner, A.; Pauwels, P.; Vergauwen, M.; Klein, R. Including widespread geometry formats in semantic graphs using RDF literals. In Proceedings of the European Conference on Computing in Construction, Crete, Greece, 10–12 July 2019.
72. Kokar, M.M.; Matheus, C.J.; Baclawski, K. Ontology-based situation awareness. *Inf. Fusion* **2009**, *10*, 83–98. [CrossRef]
73. Haidegger, T.; Barreto, M.; Gonçalves, P.; Habib, M.K.; Ragavan, S.K.V.; Li, H.; Vaccarella, A.; Perrone, R.; Prestes, E. Applied ontologies and standards for service robots. *Robot. Auton. Syst.* **2013**, *61*, 1215–1223. [CrossRef]
74. Barbera, T.; Albus, J.; Messina, E.; Schlenoff, C.; Horst, J. How task analysis can be used to derive and organize the knowledge for the control of autonomous vehicles. *Robot. Auton. Syst.* **2004**, *49*, 67–78. [CrossRef]
75. Kostavelis, I.; Gasteratos, A. Semantic mapping for mobile robotics tasks: A survey. *Robot. Auton. Syst.* **2015**, *66*, 86–103. [CrossRef]
76. Bourreau, P.; Charbel, N.; Werbrouck, J.; Senthilvel, M.; Pauwels, P.; Beetz, J. Multiple inheritance for a modular BIM. In *Le BIM et L'évolution des Pratiques: Ingénierie et Architecture, Enseignement et Recherche*; Eyrolles: Paris, France, 2020.
77. Hamledari, H.; Rezazadeh Azar, E.; McCabe, B. IFC-based development of as-built and as-is BIMs using construction and facility inspection data: Site-to-BIM data transfer automation. *J. Comput. Civ. Eng.* **2018**, *32*, 04017075. [CrossRef]
78. Wang, Q.; Kim, M.K.; Cheng, J.C.; Sohn, H. Automated quality assessment of precast concrete elements with geometry irregularities using terrestrial laser scanning. *Autom. Constr.* **2016**, *68*, 170–182. [CrossRef]
79. Adan, A.; Prieto, S.A.; Quintana, B.; Prado, T.; Garcia, J. An autonomous thermal scanning system with which to obtain 3D thermal models of buildings. In Proceedings of the 35th CIB W78 Conference on IT in Design, Construction, and Management, Chicago, IL, USA, 9 October 2018; Springer: Chicago, IL, USA, 2019; pp. 489–496.
80. Freimuth, H.; König, M. A framework for automated acquisition and processing of As-built data with autonomous unmanned aerial vehicles. *Sensors* **2019**, *19*, 4513. [CrossRef]
81. La, H.M.; Dinh, T.H.; Pham, N.H.; Ha, Q.P.; Pham, A.Q. Automated robotic monitoring and inspection of steel structures and bridges. *Robotica* **2019**, *37*, 947–967. [CrossRef]
82. Wang, J.; Luo, C. Automatic Wall Defect Detection Using an Autonomous Robot: A Focus on Data Collection. In Proceedings of the ASCE International Conference on Computing in Civil Engineering: Data, Sensing, and Analytics, Atlanta, GA, USA, 17–19 June 2019; American Society of Civil Engineers: Reston, VA, USA, 2019; pp. 312–319.
83. Wang, C.; Chi, W.; Sun, Y.; Meng, M.Q.H. Autonomous robotic exploration by incremental road map construction. *IEEE Trans. Autom. Sci. Eng.* **2019**, *16*, 1720–1731. [CrossRef]

84. Freimuth, H.; König, M. Planning and executing construction inspections with unmanned aerial vehicles. *Autom. Constr.* **2018**, *96*, 540–553. [CrossRef]
85. Menendez, E.; Victores, J.G.; Montero, R.; Martínez, S.; Balaguer, C. Tunnel structural inspection and assessment using an autonomous robotic system. *Autom. Constr.* **2018**, *87*, 117–126. [CrossRef]
86. Rea, P.; Ottaviano, E. Design and development of an Inspection Robotic System for indoor applications. *Robot. Comput. Integr. Manuf.* **2018**, *49*, 143–151. [CrossRef]
87. Sutter, B.; Lelevé, A.; Pham, M.T.; Gouin, O.; Jupille, N.; Kuhn, M.; Lulé, P.; Michaud, P.; Rémy, P. A semi-autonomous mobile robot for bridge inspection. *Autom. Constr.* **2018**, *91*, 111–119. [CrossRef]
88. Krawczyk, J.M.; Mazur, A.M.; Sasin, T.; Stokłosa, A.W. Infrared building inspection with unmanned aerial vehicles. *Pr. Inst. Lotnictwa* **2015**, *3*, 32–48. [CrossRef]
89. Saigol, Z.; Ridder, B.; Wang, M.; Dearden, R.; Fox, M.; Hawes, N.; Lane, D.M.; Long, D. Efficient search for known objects in unknown environments using autonomous indoor robots. In Proceedings of the IROS Workshop on Task Planning for Intelligent Robots in Service and Manufacturing, Hamburg, Germany, 2 October 2015.
90. Lim, R.S.; La, H.M.; Sheng, W. A robotic crack inspection and mapping system for bridge deck maintenance. *IEEE Trans. Autom. Sci. Eng.* **2014**, *11*, 367–378. [CrossRef]
91. Roca, D.; Laguela, S.; Diaz-Vilarino, L.; Armesto, J.; Arias, P. Low-cost aerial unit for outdoor inspection of building facades. *Autom. Constr.* **2013**, *36*, 128–135. [CrossRef]
92. Yu, S.N.; Jang, J.H.; Han, C.S. Auto inspection system using a mobile robot for detecting concrete cracks in a tunnel. *Autom. Constr.* **2007**, *16*, 255–261. [CrossRef]
93. Prestes, E.; Carbonera, J.L.; Fiorini, S.R.; Jorge, V.A.; Abel, M.; Madhavan, R.; Locoro, A.; Goncalves, P.; Barreto, M.E.; Habib, M.; et al. Towards a core ontology for robotics and automation. *Robot. Auton. Syst.* **2013**, *61*, 1193–1204. [CrossRef]
94. Noy, N.F.; Griffith, N.; Musen, M.A. Collecting community-based mappings in an ontology repository. In Proceedings of the 7th ISWC International Conference on Semantic Web, Karlsruhe, Germany, 16–30 October 2008.
95. Brank, J.; Grobelnik, M.; Mladenic, D. A survey of ontology evaluation techniques. In Proceedings of the Conference on Data Mining and Data Warehouses, Copenhagen, Denmark, 22–26 August 2005; Citeseer: Ljubljana, Slovenia, 2005; pp. 166–170.
96. KBSI. IDEF—Integrated DEFINition Methods (IDEF). 2020. Available online: <https://www.idef.com/> (accessed on 18 July 2020).
97. Taher, A.; Vahdatikhaki, F.; Hammad, A. Towards Developing an Ontology for Earthwork Operation. In Proceedings of the ASCE International Workshop on Computing in Civil Engineering, Washington, DC, USA, 25–27 June 2017.
98. Klingbeil, E.; Saxena, A.; Ng, A.Y. Learning to open new doors. In Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, 18–22 October 2010; IEEE: Taipei, Taiwan, 2010; pp. 2751–2757.
99. Cobalt Robotics. The First Security Industry Solution to Allow Robots to Open Doors. 2019. Available online: <https://www.cobaltai.com/cobalt-robotics-announces-door-integration-solution/> (accessed on 30 January 2020).
100. Richardson, B. *Defects and Deterioration in Buildings: A Practical Guide to the Science and Technology of Material Failure*, 2nd ed.; Routledge: London, UK, 2002.
101. Lohani, B.; Singh, R. Effect of data density, scan angle, and flying height on the accuracy of building extraction using LiDAR data. *Geocarto Int.* **2008**, *32*, 81–94. [CrossRef]
102. Haghighi, P.D.; Burstein, F.; Zaslavsky, A.; Arbon, P. Development and evaluation of ontology for intelligent decision support in medical emergency management for mass gatherings. *Decis. Support Syst.* **2013**, *54*, 1192–1204. [CrossRef]
103. Yu, J.; Thom, J.A.; Tam, A. Requirements-oriented methodology for evaluating ontologies. *Inf. Syst.* **2009**, *34*, 766–791. [CrossRef]
104. Oxford University, The Knowledge Representation and Reasoning Group, Hermit OWL Reason. 2019. Available online: <http://www.hermit-reasoner.com/> (accessed on 10 June 2020).
105. Hercz, T.; Liu, W. Mecabot User Manual. 2023. Available online: https://cdn.robotshop.com/rbm/815d1a40-62cd-43a6-8dad-e05323e8953a/e/e350701f-db45-4050-bb03-255ea0a04018/c71e4dfe_mecabot-user-manual-v.20230330.pdf (accessed on 1 March 2023).
106. Compton, M.; Barnaghi, P.; Bermudez, L.; Garcia-Castro, R.; Corcho, O.; Cox, S.; Graybeal, J.; Hauswirth, M.; Henson, C.; Herzog, A.; et al. The SSN ontology of the W3C semantic sensor network incubator group. *J. Web Semant.* **2012**, *17*, 25–32. [CrossRef]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.