

Article

OntoSLAM: An Ontology for Representing Location and Simultaneous Mapping Information for Autonomous Robots

Maria A. Cornejo-Lupa ^{1,†} , Yudith Cardinale ^{2,3,*,†} , Regina Ticona-Herrera ^{1,†}, Dennis Barrios-Aranibar ^{2,†} , Manoel Andrade ⁴ and Jose Diaz-Amado ^{2,4} 

¹ Computer Science Department, Universidad Católica San Pablo, Arequipa 04001, Peru; maria.cornejo.lupa@ucsp.edu.pe (M.A.C.-L.); rticona@ucsp.edu.pe (R.T.-H.)

² Electrical and Electronics Engineering Department, Universidad Católica San Pablo, Arequipa 04001, Peru; dbarrios@ucsp.edu.pe (D.B.-A.); jose_diaz@ifba.edu.br (J.D.-A.)

³ Department of Computer Science, Universidad Simón Bolívar, Caracas 1086, Venezuela

⁴ Instituto Federal da Bahia, Vitoria da Conquista 45078-300, Brazil; manoelpedro@ifba.edu.br

* Correspondence: ycardinale@usb.ve

† These authors contributed equally to this work.

Abstract: Autonomous robots are playing an important role to solve the Simultaneous Localization and Mapping (SLAM) problem in different domains. To generate flexible, intelligent, and interoperable solutions for SLAM, it is a must to model the complex knowledge managed in these scenarios (i.e., robots characteristics and capabilities, maps information, locations of robots and landmarks, etc.) with a standard and formal representation. Some studies have proposed ontologies as the standard representation of such knowledge; however, most of them only cover partial aspects of the information managed by SLAM solutions. In this context, the main contribution of this work is a complete ontology, called OntoSLAM, to model all aspects related to autonomous robots and the SLAM problem, towards the standardization needed in robotics, which is not reached until now with the existing SLAM ontologies. A comparative evaluation of OntoSLAM with state-of-the-art SLAM ontologies is performed, to show how OntoSLAM covers the gaps of the existing SLAM knowledge representation models. Results show the superiority of OntoSLAM at the *Domain Knowledge* level and similarities with other ontologies at *Lexical* and *Structural* levels. Additionally, OntoSLAM is integrated into the Robot Operating System (ROS) and Gazebo simulator to test it with Pepper robots and demonstrate its suitability, applicability, and flexibility. Experiments show how OntoSLAM provides semantic benefits to autonomous robots, such as the capability of inferring data from organized knowledge representation, without compromising the information for the application and becoming closer to the standardization needed in robotics.

Keywords: SLAM; autonomous and mobile robots; ontology; ontologies evaluation



Citation: Cornejo-Lupa, M.A.; Cardinale, Y.; Ticona-Herrera, R.; Barrios-Aranibar, D.; Andrade, M.; Diaz-Amado, J. OntoSLAM: An Ontology for Representing Location and Simultaneous Mapping Information for Autonomous Robots. *Robotics* **2021**, *10*, 125. <https://doi.org/10.3390/robotics10040125>

Academic Editor: Rui P. Rocha

Received: 9 October 2021

Accepted: 15 November 2021

Published: 21 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 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

The evolution of mobile technologies and sensors has increased the complexity of autonomous robot behaviors in many scenarios, including Simultaneous Localization and Mapping (SLAM) applications [1,2]. Naturally, these complex behaviors imply the use of more complex knowledge (i.e., robots characteristics and capabilities, maps information, locations of robots and landmarks, etc.) and require understanding the SLAM problem as a continuous and dynamic process, since the physical world that robots explore may constantly change. Although SLAM is a well-researched area and has reached a high level of maturity [3], there is still a lack of standardization to represent the information needed to propose efficient and interoperable solutions [4]. In this context, the need for a standard and well-defined model to capture the knowledge used by SLAM algorithms becomes evident. Ontologies, from the Semantic Web, seem to be suitable options, since they allow

standardizing and making the knowledge of a specific domain readable for both humans and machines [5].

In the context of SLAM, the use of an ontology leads to interoperability among robots. For example, despite using different techniques or sensors, robots can store and share the knowledge acquired with the same ontology: an aerial robot in a 3D spatial scenario could share the location of features with a terrestrial robot in a 2D spatial scenario. Some studies have formulated ontologies to partially model the information related to some aspects of SLAM, as shown in the studies presented in [6,7], that propose a categorization of the knowledge domain of SLAM and compare state-of-the-art SLAM ontologies. Those studies show that most SLAM ontologies are focused on the knowledge related to the SLAM final result (i.e., the maps) and considering the SLAM problem as a static process [8–10]. Nonetheless, to develop a complete ontology, it is necessary to consider not only the outcome of SLAM applications, but also to examine the inherent characteristics of the SLAM dynamics, such as uncertainty.

To overcome these limitations, in this work it is proposed OntoSLAM, an ontology that represents all knowledge related to autonomous robots and the SLAM problem considered to be a continuous process with the presence of uncertainty in robots and landmarks positions. Thus, the main contribution of this work is a complete ontology to model all aspects related to autonomous robots and the SLAM problem, towards the standardization needed in robotics, which is not reached until now with the existing SLAM ontologies. OntoSLAM is the result of an integration and extension of three basis ontologies taking the best from them to overcome their individual limitations [11–13]. The general design of OntoSLAM is presented, as well as a comparative evaluation with state-of-the-art SLAM ontologies. The comparative evaluation shows that OntoSLAM is superior in terms of *Quality* and *Correctness* at *Domain Knowledge* level and similar metrics at *Lexical* and *Structural* levels. Moreover, to demonstrate its suitability, applicability, and flexibility, OntoSLAM is integrated into Robot Operating System (ROS) and Gazebo [14] simulator to test it with Pepper robots. Results prove the functionality of OntoSLAM, its generality, maintainability, and re-usability towards the standardization needed in robotics, without losing any information but gaining semantic benefits. Experiments show how OntoSLAM provides autonomous robots the capability of inferring data from organized knowledge representation, without compromising the information for the application.

The remainder of this article is organized as follows. Related studies are described and compared in Section 2. The description of OntoSLAM is presented in Section 3. Results of validation and performance evaluation of OntoSLAM are described in Section 4. Finally, conclusions and future work is discussed in Section 5.

2. Related Work

In a previous study, it was proposed four categories of the knowledge managed by SLAM applications [6], each one consisting of several subcategories as:

1. **Robot Information (RI):** Conceptualizes the main characteristics of the robot, its physical and structural capabilities. It additionally considers the location, with its correlative uncertainty, of the robot in a map and its pose, because according to that the robot could act differently within its environment. It considers the following aspects:
 - (a) Robot kinematic information: It is related to the mobility capacity and degrees of freedom of each part of the robot.
 - (b) Robot sensory information: It refers to the different sensors that robots use to explore the world.
 - (c) Robot pose information: To model the information related to the robot's location and position and orientation associated with its degrees of freedom.
 - (d) Robot trajectory information: To represent information related to the association of a sequence of certain poses with respect to time.

- (e) **Robot position uncertainty:** There is an uncertainty related to a set of positions in which the robot could be. Therefore, it is necessary to model the possible positions and the actual positions of the robot.
- 2. **Environment Mapping (EM):** Represents the robot's ability to describe the environment in which it is located, including other objects than robots. This category contemplates objects main features such as color and dimensions, as well as position and uncertainty of that position. This modeling capability is what opens the possibility of a more complex SLAM, since if robots are able to differentiate objects from their environments, they have the ability to locate itself either quantitatively or qualitatively with respect to such objects. It includes the following subcategories:
 - (a) **Geographical information:** It refers to the modeling of physical spaces mapped by the robot, comprising simple areas (such as an office) and complex areas (such as a building with its interior offices).
 - (b) **Landmark basic information (position):** It models the objects and their position with respect to the map generated by the robot, while dealing with the SLAM problem.
 - (c) **Landmark shape information:** It refers to the characteristics of each object, related to its size, shape, and composition. In some environments, the robot could have the ability to decompose landmarks into simpler parts and the ontology would allow it to model this.
 - (d) **Landmark position uncertainty:** Analogous to the uncertainty of the robot's position in the first category, this subcategory seeks to model the uncertainty of the position of each of the landmarks found in the environment.
- 3. **Timely Information (TI):** Related to the capability of modeling a path of the robot, representing its movements—i.e., where it has moved and for how long it has remained in that movement or position. The aspects considered in this category are:
 - (a) **Time information of robots and objects:** To consider the space-time relationship of the robot's positions.
 - (b) **Mobile objects:** It models objects that may be in one position at one instant in time and the next instant no longer be present in that position, either because it moved (e.g., a bicycle) or because someone else moved it (e.g., a box).
- 4. **Workspace Information (WI):** Models the general characteristics of the environment being mapped, such as its dimensional space, as well as the capacity of modeling entities that belong only to a specific domain. This category includes the following two subcategories:
 - (a) **Dimensions of mapping and localization:** It refers to the number of dimensions (2D, 3D) in which the robot determines its localization and performs the mapping of the environment.
 - (b) **Specific domain information:** Since it is necessary to solve the SLAM problem in varied environments, it is necessary to be able to model a high-level knowledge of the environment in which the robot is located, also considering the knowledge domain, where SLAM is being applied. Examples of specific knowledge that can be modeled could be related to objects in a museum (for a tourism application) or objects in an office (for a workspace application).

In total, from the categorization of SLAM knowledge, there exist 13 subcategories that represent the aspects that might be considered when modeling the SLAM problem. In a previous work [7], the most popular and recent SLAM ontologies up to 2020 are revised, classifying them according to the proposed categorization. In this section, that review is updated up to 2021 and it is presented a brief description on how the existing ontologies model partial aspects of the knowledge associated with SLAM, according to the categorization considered. In Table 1, a black circle (●) means that the corresponding ontology conceptualizes the respective subcategory; a gray circle (●) represents that the on-

ontology partially models the corresponding subcategory; and an empty circle (○) designates ontologies that do not conceptualize the subcategory.

Table 1. Summary of evaluation of ontologies for SLAM.

Name	Ref.	Robot Information					Environment Mapping				Time Information		Workspace Information	
		a	b	c	d	e	a	b	c	d	a	b	a	b
Robot Ontology, 2005	[15]	●	●	○	○	○	○	○	○	○	○	○	○	○
Martinez et al., 2007	[16]	●	○	●	○	○	●	●	○	○	○	○	○	●
OMRKE, 2007	[17]	○	●	○	○	○	●	●	○	○	●	○	●	●
SUMO, 2007	[18]	○	●	○	○	○	○	○	○	○	○	○	○	○
Space Ontology, 2010	[8]	○	○	○	○	○	●	○	○	○	○	○	○	○
OUR-K, 2011	[19]	●	●	●	○	○	●	●	●	○	●	○	●	○
PROTEUS, 2011	[20]	●	●	●	●	●	●	●	○	○	○	○	○	●
Uncertain Ontology, 2011	[21]	○	●	○	○	○	●	●	●	●	○	○	●	●
Wang and Chen, 2011	[22]	○	○	○	○	○	●	●	●	○	○	○	●	●
KnowRob, 2012	[13]	●	●	●	○	○	○	●	○	○	○	○	●	●
Hotz et al., 2012	[23]	○	○	●	○	○	●	●	○	○	○	○	○	●
OASys, 2012	[24]	●	●	○	○	○	○	○	○	○	○	○	○	○
Core Ontology, 2013	[10]	●	○	○	○	○	●	●	○	○	○	○	○	○
Li et al., 2013	[25]	○	○	●	○	○	●	●	○	○	○	○	●	●
POS, 2013	[26]	●	○	●	○	○	●	●	○	○	○	○	●	○
V. Fortes, 2013	[12]	●	○	●	●	○	●	●	○	○	●	●	●	○
Wu et al., 2014	[27]	●	●	●	○	○	●	●	●	●	○	○	●	○
RoboEarth, 2015	[28]	●	●	●	○	○	●	●	●	●	●	○	●	●
ROSPlan, 2015	[9]	○	●	●	○	○	●	●	○	○	○	○	○	○
Burroughes and Gao, 2017	[29]	●	●	●	○	○	●	●	○	○	●	●	●	○
ADRON, 2018	[30]	●	●	○	○	○	○	○	○	○	○	○	○	○
Deeken et al., 2018	[31]	○	●	●	○	○	●	●	○	○	○	○	●	●
Sun et al., 2019	[32]	●	●	●	●	○	●	●	○	●	○	○	●	●
ISRO, 2020	[11]	●	○	●	●	○	●	●	●	○	●	●	○	●
Crespo et al., 2020	[33]	●	●	●	○	○	●	●	●	○	○	○	●	●
Sung-Hyeon et al., 2020	[34]	●	●	●	○	○	●	●	●	○	○	●	●	●
BIRS, 2021	[35]	●	●	●	○	○	●	●	●	○	○	○	●	●
Shchekotov et al., 2021	[36]	○	○	○	○	○	●	●	●	○	●	●	○	●
OntoSLAM		●	●	●	●	●	●	●	●	●	●	●	●	●

Almost all analyzed ontologies represent partial knowledge of **Robot Information**, only PROTEUS [20] covers all subcategories. Robot Ontology [15], SUMO [18], ADRON [30], and OASys [24] only model partial knowledge for this category, neglecting all other categories.

Regarding **Environment Mapping**, Space Ontology [8] models only the geographical information and nothing from all other categories. All other ontologies, but Robot Ontology [15], SUMO [18], ADRON [30], and OASys [24], partially represent this category. Only Core Ontology for Robots & Automation (CORA) [10], POS [26], and ROSPlan [9] are focused on the two first categories.

Few of the revised ontologies partially model the knowledge of **Timely Information** [11,12,17,19,28,29,34,36], also these analyzed ontologies partially model aspects in all categories.

Concerning **Workspace Information**, some ontologies allow representing specific domain objects, such as the ontologies proposed in [22,25,31], which represent specific objects of an office (e.g., monitor, desk, printer) to describe the robot's environment; KnowRob [13] and the ontology proposed by Hotz et al. in [23] allow representing objects of restaurant environments, such as cup, chair, and kitchen; and the one proposed by Sun et al. in [32] related to Search and Rescue (SAR) scenarios that model concepts such as search and rescue. The remain works [16,21,27,33–36] are designed for a non-specific indoor environments with concepts such as cabinet, sink, sofa, and beds.

Table 1 shows that few ontologies consider **Timely Information**, thus, most of them disregard dynamic environments for SLAM solutions; none of the ontologies analyzed, with the exception of the proposed OntoSLAM, models all 13 aspects of SLAM knowledge, presenting limitations to solve the SLAM problem. Although there exist several ontologies to represent such knowledge, it is evident that there is a lack of a standard arrangement and generic ontology covering the full aspects of the SLAM knowledge.

In this sense, OntoSLAM represents a novel development of an ontology, which is a global solution that covers all the proposed subcategories. In particular, it models the dynamics of the SLAM process by including uncertainty of robot and landmarks positions. The following section explains the proposal in detail.

3. OntoSLAM: The Proposal

To be able of representing all knowledge related to SLAM and overcome the limitations of existing ontologies, in this work it is proposed OntoSLAM, an extensible and complete SLAM ontology, freely available (<https://github.com/Alex23013/ontoSLAM> accessed on 16 November 2021). For the design of OntoSLAM, the following ontologies are used as a basis:

- ISRO [11]: it is a recent developed ontology in the field of service robotics, with the aim of improving human-robot interactions; therefore, it includes robotic and human agents in its models.
- The ontology proposed by V. Fortes [12]: It will hereafter be referred as FR2013 ontology; it is an ontology aimed at solving the problem of mixing maps when two robots collaboratively map a space; it integrates and extends POS [26] and CORA [10] ontologies (developed by the IEEE-RAS working group) [15], which in turn inherit general concepts from the SUMO ontology [18], that has been highly referenced.
- KnowRob ontology [13]: it is a framework developed for teleoperation environments, designed around a robotic agent, whose main mission is to fetch things and it must perform SLAM to fulfill this mission; therefore, the ontology allows describing the place where it is; this ontology is already developed and tested in ROS, which gives free access to the packages and ontologies developed in this framework.

These ontologies have the following common characteristics that make them suitable for the purpose of this work:

- They cover at least three of the four SLAM knowledge categories.
- They cover at least one category completely.
- They provide open source or a detailed explanation of the ontology structure, to facilitate the integration and extension of the ontological concepts.

For the developing process of OntoSLAM, it is followed a three-step methodological process, consisting of: *Context Familiarization*, *Implementation*, and *Validation*, as shown in Figure 1.

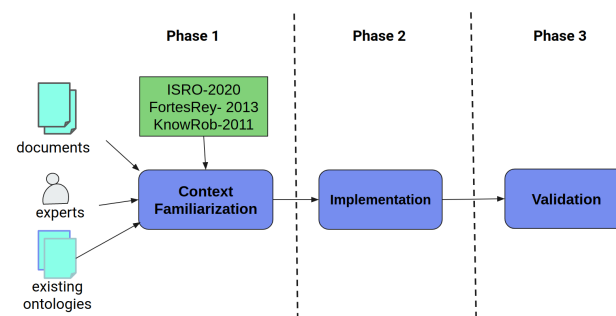


Figure 1. OntoSLAM development flow.

3.1. Context Familiarization

This phase comprises the research and review of related studies to become familiar with the terminology, knowledge, and existing works in the context of the SLAM problem. Documents such as articles, technical reports, and books serve as a source of information for the familiarization of the SLAM problem and the knowledge to be represented in an ontology. Existing ontologies are selected, evaluated, and finally fully or partially reused, paying attention to the level of granularity (whether the existing ontology covers the same level of detail as the ontology under development). SLAM domain experts also act as a source and support for conceptualization, since they provide their terminology. Section 2 and the previous work presented in [7], reflect some results of this familiarization phase.

3.2. Implementation

During this phase, OntoSLAM is developed as a result of extending and reusing some concepts from the selected ontologies. To distinguish entities (e.g., classes, relations, properties) taken from the basis ontologies and the new added entities, it is used the following format $\langle \text{prefix} \rangle : \langle \text{entityName} \rangle$, where $\langle \text{prefix} \rangle$ is an abbreviation of the name of the ontology to which the entity belongs to and $\langle \text{entityName} \rangle$ is the name of the entity. For example, `cora:Robot` refers to the entity Robot of the CORA ontology. The ontology prefixes used in this work are:

- `isro`: for ISRO ontology;
- `kn`: for entities taken from the KnowRob framework;
- `fr`: refers to the FR2013 ontology;
- `cora`: is the prefix for CORA ontology;
- `os`: refers to OntoSLAM (the proposal in this work).

As most ontologies, the base class of OntoSLAM is `os:Thing`, which defines anything that exists. This class has two subclasses, as shown in Figure 2:

- `os:PhysicalThing`, that denotes all things that occupy a physical space in the environment. It can be (see Figure 3):
 - `isro:Agent`, that denotes an entity that perceives and acts on its environment. This class can be extended to model both robotic and human agents.
 - `os:Part`, that represents the basic building block for modeling an object. A part can be composed of other parts but can also be atomic.
 - `os:Joint`, that models the connection between two parts. It defines the pose of the parts to which it is connected. Every joint must have a connection with two parts.
 - `cora:Environment`, that refers to a region that occupies a physical location in a space.
- `os:AbstractThing`, that describes things that exist but do not occupy a physical place in the space. It has the following subclasses:
 - `os:StructuralModel`, which represents a set of `os:Part` and `os:Joint`. A model describes the whole structure of a physical thing. It is used to describe agents,

parts, and environments. All `os:PhysicalThing` have an `os:StructuralModel`, linked with the relation `os:hasModel`.

- `kn:MathematicalThing`, that denotes all mathematical concepts used during the formalization of the knowledge obtained while solving the SLAM problem. Examples of this class are vectors and matrices.
- `os:FeatureThing`, which represents the characteristics that a physical thing can have; for example, color or shape.
- `isro:TemporalThing`, which represents all the entities necessary to model the time associated with the events that occur during the SLAM process. Its main subclasses are: `isro:TimePoint` and `isro:TimeInterval`.
- `os:PositionalThing`, used for concepts related to the positioning of both robots and objects in the working environment.

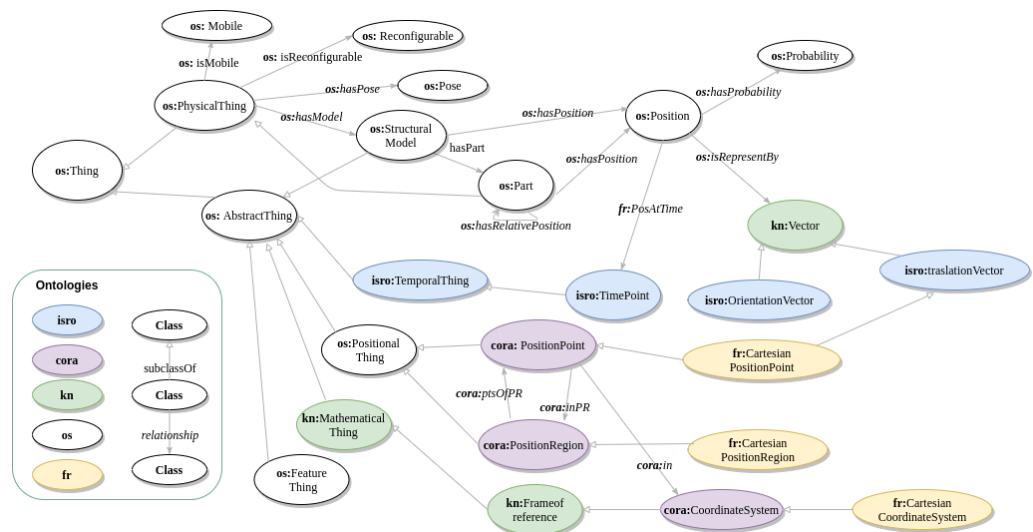


Figure 2. Main concepts and relationships of OntoSLAM related to Positioning.

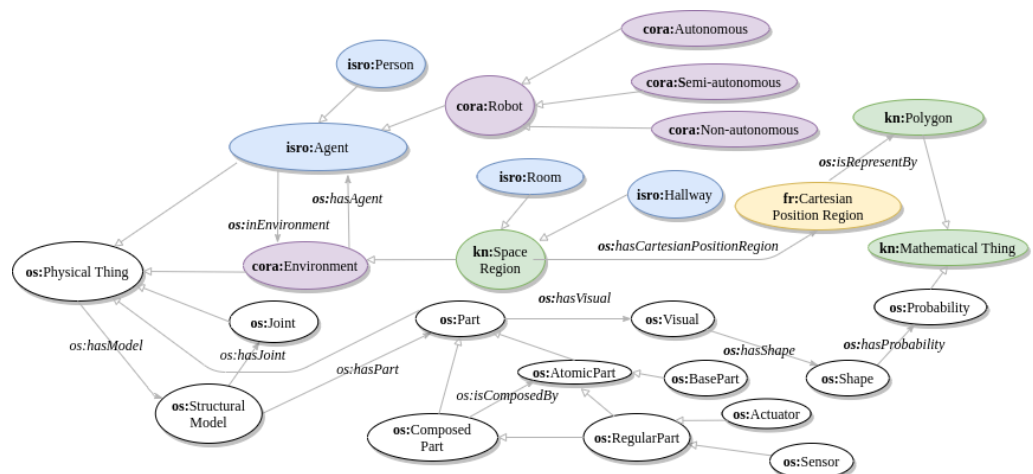


Figure 3. OntoSLAM main classes of Robot Information and Environment Mapping.

Figure 2 additionally shows classes of OntoSLAM related to positioning. To represent dynamic positions and uncertainty, class `os:Position` is related to `isro:TimePoint` class, through the relation `fr:PosAtTime`, and to the probability (`os:Probability`) of being in that position, through the relation `os:hasProbability`. Furthermore, the `os:Mobile` class is used to model mobile objects and the `os:Reconfigurable` class is used to model objects that can change their pose but not their position.

Figure 3 shows the main classes that model **Robot Information** and **Environment Mapping**. One of the main aspects is the class hierarchy to model the parts. The `os:Compo-`

sedPart class represents the set of several `os:AtomicPart`, which can be the `os:BasePart`, that determines the position of the robot, or `os:RegularPart`, which can be `os:Actuator` or `os:Sensor` type. In addition, an `os:Part` has associated visual characteristics, such as shape (`os:Shape`) that also has a value of uncertainty (`os:Probability`), which can be updated as the robot performs the SLAM. This `os:Shape` can be a known geometric figure, such as `os:Cylinder`, `os:Plane`, `os:Sphere`, `os:Box`. However, in case it is not certain to which figure it belongs, it can be modeled as `os:Undefined`, a class specialized in two types: `os:HeightMap` and `os:OccupancyGrid`, which are two formats used in robotics to save maps without losing information. Other features that can be modeled are colors (`os:Color`) and the dimensions (`os:Dimension`) of the visual component of the `os:Part`. These last two features and the `os:Shape` are subclasses of `os:AbstractThing`.

Figure 4 shows the main classes that model **Temporal Information**. For this module the ISRO ontology has been taken as a base, starting from its base concept `isro:Temporal-Thing`, which in turns is specialized in two subclasses: `isro:TimePoint` and `isro:TimeInterval`. The first one is associated with the position (`os:Position`) attributed to each `os:Part`, through the relation `os:atTime`. With this concept it is possible to model the trajectory of the robot over the time. On the other hand, with the `isro:TimeInterval` class, it is possible to model processes that have a certain duration. For example, the time in which the SLAM process was performed. To determine this duration, the subclasses `isro:StartInterval` and `isro:EndInterval` are used. In addition, the class `os:State`, refers to whether the object was moved or not at the time being evaluated, with the following four values: Reconfigured, Moved, Not reconfigured, or Not moved. These values are set through the `os:isMobile` and `os:isReconfigurable` relationships.

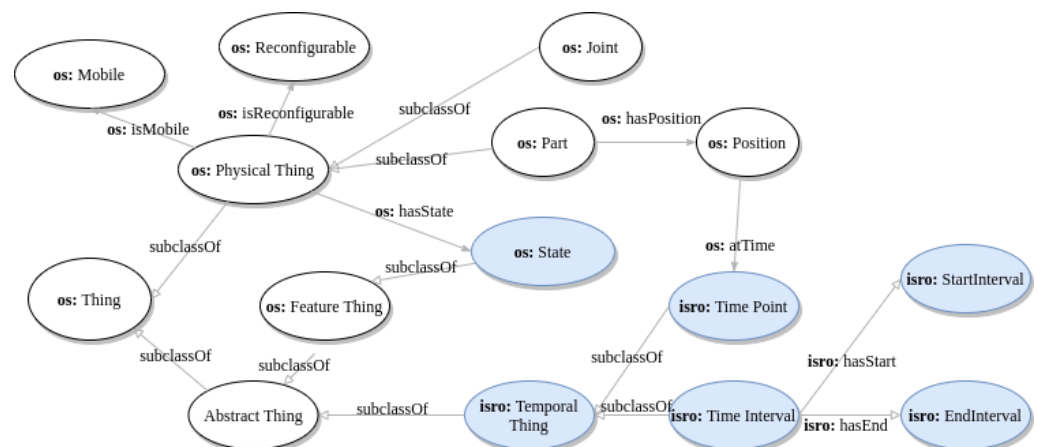


Figure 4. OntoSLAM main classes of Temporal Information.

Regarding the last category, **Workspace Information**, OntoSLAM does not offer specific concepts of any specific domain. However, it can be easily extended from the `os:Thing`, `os:PhysicalThing`, or `os:AbstractThing` classes to integrate a specific domain ontology or classes that represents elements of specific environments, such as *chair*, *table*, *plates* (for a restaurant application), *artworks* (for a museum application).

Once the concepts that constitute the proposed ontology are defined, a validation process must be conducted to ensure compliance with the requirements of OntoSLAM.

3.3. Validation

To evaluate OntoSLAM, the methodology for evaluating and comparing ontologies proposed in [37] is used. This methodology bases the evaluation on a *golden-standard* to measure the *Lexical*, *Structural*, and *Domain Knowledge* levels of ontologies, from two perspectives: *Quality* and *Correctness*.

The *Lexical* level includes linguistic, vocabulary, and syntactic aspects; the *Structural* level considers aspects related to taxonomy, hierarchy, relationships, architecture, and

design that define the ontology; and the *Domain Knowledge* level considers how well the knowledge is covered and how the application results are improved using the ontology. *Quality* refers to the way the ontology is structured in terms of lexemes and relations between entities. The correctness perspective seeks to review the correctness of the ontology at the level of syntax, architecture, and design. For applying this evaluation methodology, it is necessary to define a *golden-standard*, as the best reference of the SLAM knowledge representation, and select ontologies to compare with, which have available their entire code. The *golden-standard* can be a referential ontology, a corpus of documents in the domain, or a categorization of the knowledge of the domain done by experts. The previously proposed SLAM knowledge categorization (see Section 2) becomes the *golden-standard* to apply the methodology to evaluate OntoSLAM.

Based on this evaluative methodology, OntoSLAM is compared with two of its three base ontologies: FR2013 and KnowRob, since ISRO has not its code available for free use. With this comparative methodology, the improvement between OntoSLAM and two of its predecessors can be quantitatively measured. Next section presents the comparative evaluation and an illustrative case of study to show the suitability of OntoSLAM.

4. OntoSLAM Evaluation

In this section, the evaluation process of the OntoSLAM is detailed, its suitability is shown in a case of study, and the results and perspectives are discussed.

4.1. Ontology Evaluation

A comparative evaluation of OntoSLAM is performed, following the methodology proposed in [37]. The *golden-standard* is defined by the categorization of the SLAM knowledge presented in [6] and OntoSLAM is compared with KnowRob [13] and FR2013 ontology [12], since they are publicly available. In the following, the metrics used to evaluate *Quality* and *Correctness* on each level are shown.

4.1.1. Lexical Level

To evaluate this level, the *Linguistic Similarity* (*LS*) among the evaluated ontologies is calculated. For that, it is required to compute: (i) *String Similarity* (*StringSim*), based on the edit distance [38], between strings representing the names of the ontology entities (e.g., classes, properties, relations); to do so, it is developed a script in Python able to compute the edit distance among each pair of terms from two source code of the ontologies being compared (https://github.com/Alex23013/ontoSLAM/blob/main/formal-validation/lexical_level/LinguisticSim.py accessed on 16 November 2021); and (ii) *Document Similarity* (*DocSim*), which is related to the occurrence of entities in the ontologies; to calculate *DocSim*, the TFIDF Vectorizer class of the Python scikit-learn library [39] is used.

There is not *Lexical* comparison of the ontologies against a *golden-standard* ontology, since it is represented as a categorization of the SLAM knowledge. This comparison is only possible among actual available ontologies in some RDF format. Table 2 shows the results of comparing FR2013, KnowRob, and OntoSLAM, which source codes are public available. For both similarities (*StringSim* and *DocSim*), OntoSLAM is more similar to FR2013 than KnowRob, due to KnowRob has been developed to solve the task of teleoperation in the context of service robotics, while OntoSLAM and FR2013 have a more general SLAM-oriented scope.

Table 2. Lexical Comparison.

Pair	StringSim	DocSim	LS
FR2013/OntoSLAM	0.43	0.65	0.54
KnowRob/OntoSLAM	0.16	0.57	0.36
FR3013/KnowRob	0.15	0.55	0.35

4.1.2. Structural Level

Following the evaluation methodology, at this level, ontologies are evaluated in terms of number of classes, relations, properties, and annotations. Similar to the *Lexical* level, the *Structural* comparison is only possible among ontologies that have the source code available. Thus, there is not *Structural* comparison against a *golden-standard* ontology. Table 3 shows the analysis of the components of the three ontologies: (i) all ontologies mostly relate their classes as subclasses, with *is-a* relations; (ii) KnowRob shows the highest cohesion, because it has the highest number of relations; and (iii) the best readability would be attributed to the OntoSLAM, since it has the highest number of annotations.

Table 3. Structural Comparison.

Ontology	Classes	Relations			Properties	Annotations
		is-a	has-*	other		
FR2013	46	41	0	16	2	0
KnowRob	252	182	117	76	62	3
OntoSLAM	69	86	34	13	14	33

At this level, the relationships of the ontology and how the resources are related to each other are more relevant. Since ontologies are knowledge graphs, graph similarity techniques can be used. Table 4 presents the results obtained by Falcon-AO [40], a tool focused on ontology alignment, which evaluates linguistic and structural similarity together. Like in the *Lexical* level, FR2013 and OntoSLAM are the most similar to each other.

Table 4. Linguistic/structural similarities (from Falcon-AO).

Pair	SimLingStruc
FR2013/OntoSLAM	0.29
OntoSLAM/KnowRob	0.11
FR2013/KnowRob	0.08

4.1.3. Domain Knowledge Level

The *golden-standard* considered for the comparative evaluation is applied only to this level, since it is based on a categorization of the SLAM knowledge. Following the methodological process to evaluate this level, it should be considered the *Application Results* and *Knowledge Coverage*. *Application Results* can be evaluated with the support of domain experts, through the development of questionnaires and SPARQL queries based on the *golden-standard*. The questionnaires related to each category of the *golden-standard* are shown below:

1. Robot Information:

- (a1) Does the ontology store the geometry of the robot?
- (a2) Does the ontology define a referential system for each robot joint?
- (a3) Does the ontology recognize types of articulations?
- (a4) Does the ontology allow transformations between referential systems?
- (b1) Does the ontology define an own reference systems for each sensor?
- (c1) Does the ontology represent the pose of a robot?
- (c2) Can represent the relative position of a robot to the objects around it?
- (d1) Does it allow storage of a path of the robot and query it?
- (e1) Does the ontology conceptualizes the uncertainty of the robot position?

2. Environment Mapping:

- (a1) Does it allow storage of empty spaces and their coordinates?

- (b1) Does it differentiate objects around the robot in terms of their name and characteristics?
 - (b2) Does it allow the representation of the pose of an object in the robot environment?
 - (b3) Does it allow knowledge of the relative position between objects?
 - (c1) Does it allow storing the geometry of objects in the environment?
 - (c2) Does it allow storage of sub-objects of interest in larger objects?
 - (c3) Does it register objects (other than robots) with joints?
 - (d1) Does it model the uncertainty of objects position?
3. **Timely information:**
- (a1) Does it allow storage of the different poses of a robot in time?
 - (b1) Does it allow storage of the different poses of objects in time?
4. **Workspace:**
- (a1) Does it clearly indicate the dimensions of the workspace?
 - (b1) Does it allow the modeling of specific information of the application domain?

All these questions were translated into SPARQL queries to be answered by the ontology. Table 5 shows the results of the application of the questionnaires on the ontologies.

According to these results, FR2013 ontology performs worse with only 35% of questions answered; KnowRob has a better performance than FR2013, since it was able to answer almost all the questions of the Environment Mapping questionnaire and all the questions of the Workspace questionnaire, achieving 87.5% of the questions answered. However, OntoSLAM outperforms its predecessors by modeling 100% of all categories of the *golden-standard*, showing its superiority at the *Domain Knowledge* level.

Table 5. Domain Knowledge level—questionnaire.

Ontologies	Robot Information								Environment Mapping							Timely Inform.		Workspace Inform.		Questions Answered	
	a1	a2	a3	b1	c1	c2	d1	e1	a1	b1	b2	b3	c1	c2	c3	d1	a1	b1	a1	b1	%
FR2013	✓	✓			✓		✓		✓								✓		✓		35 %
KnowRob	✓	✓	✓		✓	✓	✓		✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	85 %
OntoSLAM	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	100 %

The result of the *Knowledge Coverage* evaluation is shown in Figure 5, which presents the three OntoSLAM basis ontologies (FR2013, KnowRob, and ISRO) and OntoSLAM itself, evaluated with respect to the defined *golden-standard* (the 13 subcategories of the SLAM knowledge). Table 1 shows the comparison at this level of OntoSLAM with all revised ontologies. This evaluation is the one that shows the best suitability of the ontology for the SLAM domain. With OntoSLAM, it is possible to cover all the categories proposed by the *golden-standard*. Once again, it is demonstrated that OntoSLAM is superior to existing SLAM ontologies in *Domain Knowledge* covering.

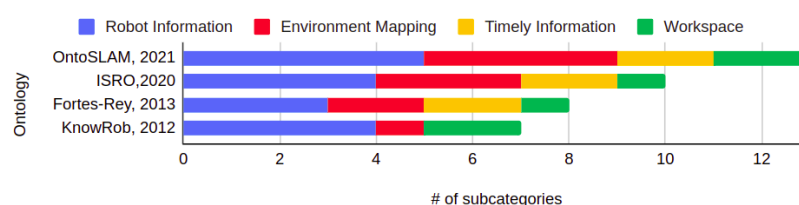


Figure 5. Comparing *Knowledge Coverage*.

4.1.4. OQuaRE Quality Metrics

The methodological comparison of ontologies proposes to complement the evaluation performed with the OQuaRE metrics [41]. They evaluate the *Quality* of the ontology based on SQuaRE (SQuaRE: SO/IEC 25000:2005 standard for Software product Quality

Requirements and Evaluation), a Software Engineering standard. The Quality Model considers the following categories: Structural, Functional Adequacy, Reliability, Operability, Compatibility, Transferability, and Maintainability. In each category, subcategories are specified to specialize the measures.

Since each OQuaRE category is evaluated with different metrics, they are assessed separately. Figure 6 shows subcategories of Functional Adequacy, in which OntoSLAM is equal or superior to its predecessors. In particular, OntoSLAM overcomes for more than 22% its predecessors in the sub-characteristic of Knowledge Reuse; it means OntoSLAM can be reused to further specialize the use of ontologies in the field of robotics and SLAM. Additionally, the three ontologies exceed 50% in the Functional Adequacy category.

The evaluation on Compatibility, Operability, and Transferability categories is shown in Figure 7. Like in the Functional Adequacy category, OntoSLAM is superior to its predecessors. Moreover, in these characteristics the three evaluated ontologies present behaviors above 80%. The highest score (97%) was obtained by OntoSLAM in the Operability category, which guarantees that OntoSLAM can be easily learned by new users.

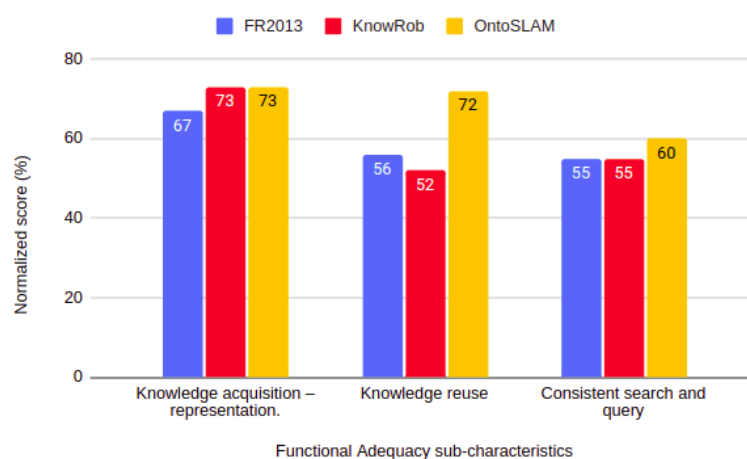


Figure 6. Quality Model: Functional Adequacy.

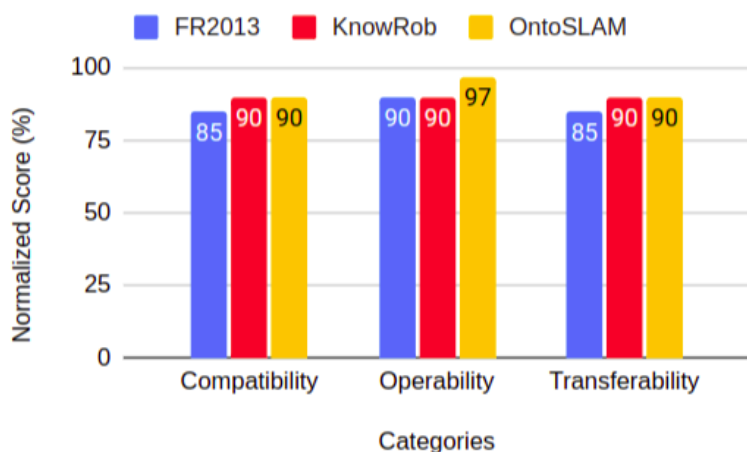


Figure 7. Quality Model: Operability, Transferability, Maintainability.

Results of the Maintainability category are shown in Figure 8. Once again, OntoSLAM shows the best performance. Moreover, the evaluated ontologies show the best results, reaching 100% in some sub-characteristics, such as Modularity and Modification Stability. Results are above 80% on average for this category, which reveals that all the ontologies evaluated are maintainable.

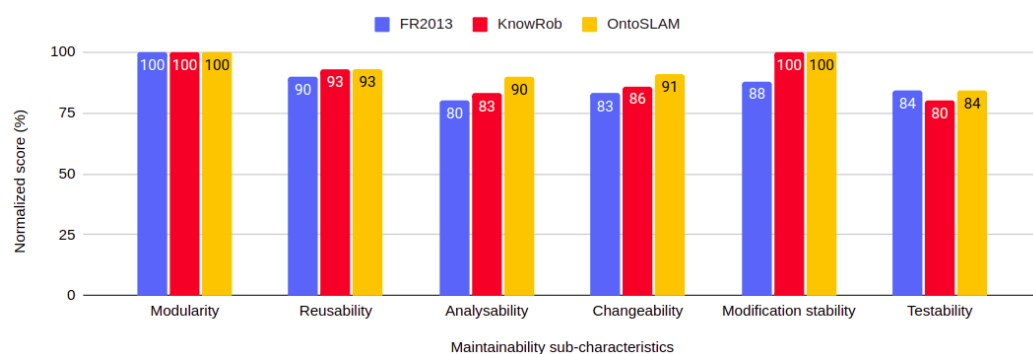


Figure 8. Quality Model: Maintainability.

All these results from the OQuARE metrics, demonstrate that the *Quality* at *Lexical* and *Structural* levels of OntoSLAM is similar or slightly superior compared with its predecessor ontologies.

4.2. Applying OntoSLAM in ROS: Case of Study

To empirically evaluate and demonstrate the suitability of OntoSLAM, it was incorporated into ROS and a set of experiments with simulated robots were performed. The simulated scenarios and their validation are designed into four phases, as shown in Figure 9. The scenario consists of two robots: Robot “A” executes a SLAM algorithm, by collecting environment information through its sensors and generates ontology instances, which are stored and published on the OntoSLAM web repository, and Robot “B” performs queries on the web repository, thus, it is able to obtain the semantic information published by Robot “A” and use it for its needs (e.g., continue the SLAM process, navigate). The simulation is as follows:

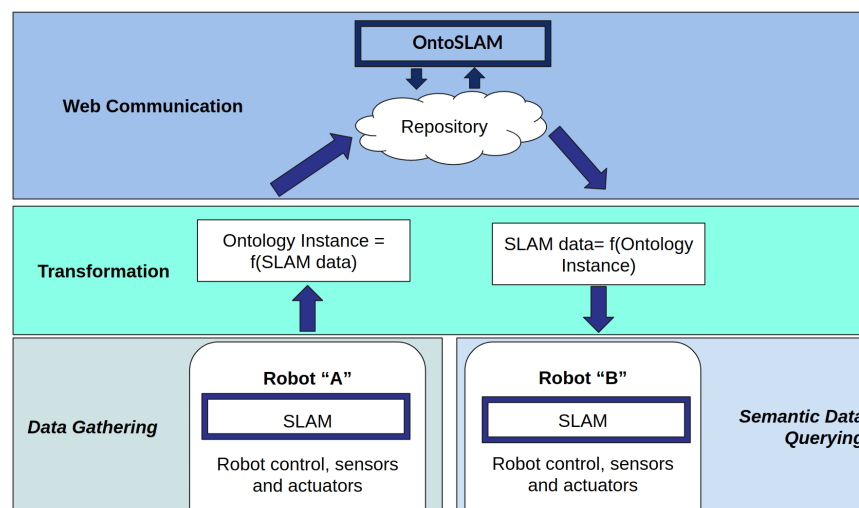


Figure 9. Data flow for the case of study.

4.2.1. Data Gathering

This phase deals with the collection of the data to perform SLAM (robot and map information). For this purpose, the well-known ROS and the simulator Gazebo are used. The Pepper robot is simulated in Gazebo and scripts subscribed to the ROS nodes, fed by the internal sensors of Robot “A” are generated. With this information obtained in real time, it is possible to move on to the transformation phase.

4.2.2. Transformation

This phase deals with the transformation of the raw data taken from the Robot “A” sensors to instances in the ontology (publish the data in the semantic repository) and the

transformation of instances of the ontology to SLAM information for Robot “B” or the same Robot “A”, during the mapping process or in another time.

To do so, the following functions are implemented:

- F1 SlamToOntology:** to convert the raw data collected by the robot’s sensors in the previous phase into instances of OntoSLAM. Information such as the name of the robot, its position, and the time at which the information was recorded, will be transformed into a set of RDF triplets that can be seen as a graph. In these experiments, Robot “A” uses this function.
- F2 OntologyToSlam:** to transform ontology instances into SLAM information in ROS format. This function is used by the Robot “B”.

Figure 10 shows an example of the use of F1 and F2, the SLAM information box represents the data collected by Robot “A” and the graph represents the OntoSLAM instance, which is the data recovered by Robot “B”. To develop both transformation functions, it is used RDFLib [42], which is a pure Python package that works with RDF. This library contains parsers and serializers for RDF/XML, N3, N-Quads, and Turtle formats.

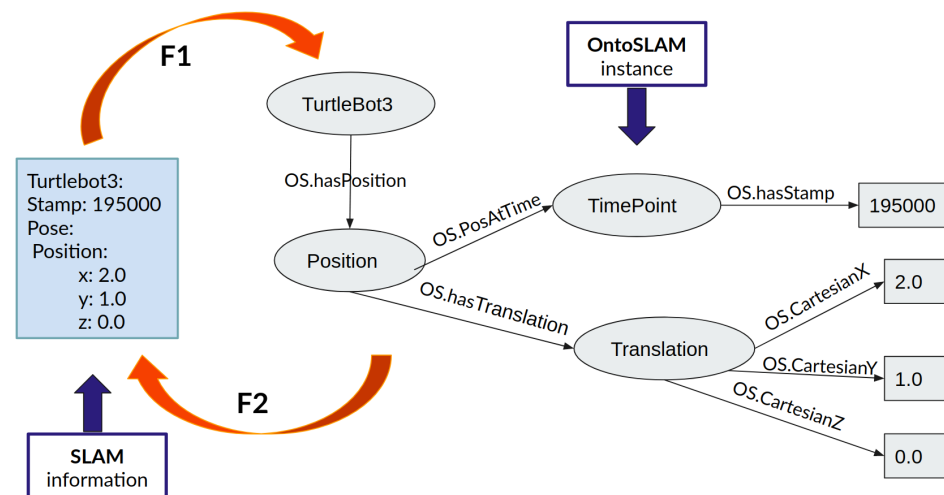


Figure 10. Transformation diagram.

4.2.3. Web Communication

This phase deals with the communication between two or more robots. For a beneficial exchange of information, there must be communication protocols and the information must be organized and modeled in a format understandable for both parties (receiver and sender). In this work, ontologies, and specifically OntoSLAM, fulfill this role of moderator and knowledge organizer. Data obtained in the *Data Gathering* phase, through the sensors of Robot “A”, which in turn are converted in a semantic format at the *Transformation* phase, also by Robot “A”, are stored and published in a web semantic repository, populated with OntoSLAM entities.

4.2.4. Semantic Data Querying

Once the OntoSLAM repository is populated by Robot “A”, Robot “B” or the same Robot “A” later in time can use this information after passed for the inverse transformation function, where the ontology instances are converted into data that the robot can understand and use for its own purposes.

To show the suitability and flexibility of OntoSLAM, two different SLAM algorithms are executed, with different scenarios, in a desktop with 256GB SSD disk, 8GB of RAM, an Nvidia® GTX 950 SC, and an Intel® Xeon® E3-1230 v2, with Ubuntu 16.04 and the Kinetic distribution of ROS and the Gazebo simulator. Figure 11 shows a scenario in a room with three landmarks: (i) Figure 11a, shows the view of the room scenario in Gazebo, where the Robot “A” (a Pepper robot in this scenario) performs the *Data Gathering*

phase; (ii) Figure 11b shows the resulting map on a 2D occupancy grids after performing SLAM with the Pepper robot and the Gmapping algorithm [43]; this map was built based on information from the laser_scan sensors of Robot “A”; (iii) Figure 11c presents the map recovered from the ontology instance, developed by the Robot “B” (another Pepper robot), showing the result of the *Semantic Data Querying* phase presented on the Rviz visualizer; (iv) Figure 11d shows the 3D map constructed by the same Robot “A” and in the same scenario, but with the octomap mapping algorithm [44], which uses the point cloud generated by the depth sensor of Robot “A”; and (v) Figure 11e, presents the recovered map by the Robot “B” from OntoSLAM.

The adaptability and compatibility of the ontology can be noticed in these experiments, since both Figure 11c,e are results of the knowledge modeled by OntoSLAM, which were generated with two different sensors (laser_scan and depth sensor) and two different SLAM algorithms (Gmapping and octomap mapping).

Figure 12 shows the same experiment but in a larger scenario with five landmarks and presence of people. In both scenarios, it is visually observed that no information is lost during the flow explained in Figure 9. All these results can be reproduced with Python scripts developed during this work, which are in a public repository on GitHub (<https://github.com/Alex23013/ontoSLAM> accessed on 16 November 2021).

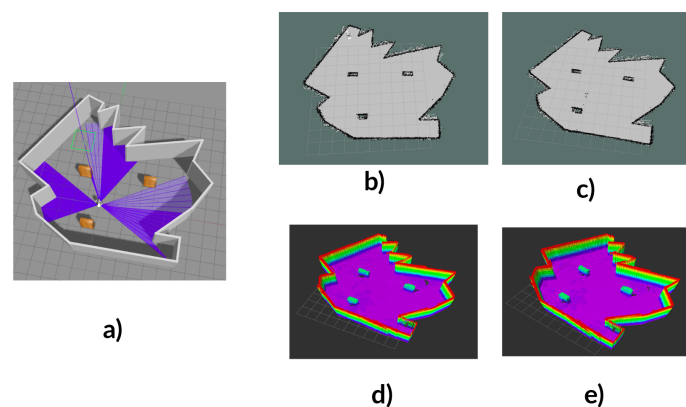


Figure 11. Experiments with Pepper in one room scenario. (a) the view of the room scenario in Gazebo, (b) the resulting map on a 2D occupancy grids after performing SLAM with the Pepper robot and the Gmapping algorithm, (c) the map recovered from the ontology instance, developed by the Robot “B”, (d) 3D map constructed by the same Robot “A” and in the same scenario, (e) recovered map by the Robot “B” from OntoSLAM.

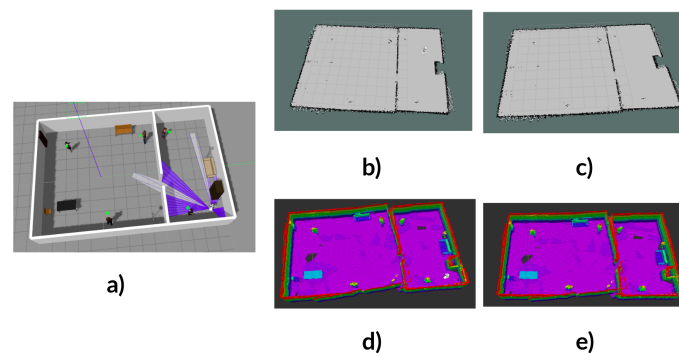


Figure 12. Experiments with Pepper in an office scenario. (a) the view of the room scenario in Gazebo, (b) the resulting map on a 2D occupancy grids after performing SLAM with the Pepper robot and the Gmapping algorithm, (c) the map recovered from the ontology instance, developed by the Robot “B”, (d) 3D map constructed by the same Robot “A” and in the same scenario, (e) recovered map by the Robot “B” from OntoSLAM.

4.3. Discussion

Results of the comparative evaluation, demonstrate that OntoSLAM is able to answer 100% of the questions of the *Domain Knowledge* questionnaire, maintaining a percentage of *Lexical* and *Structural* similarity of 54% and 29%, respectively, with its predecessor FR2013.

Moreover, OntoSLAM manages to comply with all the categories proposed by the *golden-standard*, including the subcategories relative to uncertainty and temporality that many existing ontologies do not consider. With this capability, OntoSLAM is able to model the SLAM problem as a dynamic process; therefore, more real-life scenarios are covered.

OntoSLAM outperforms its predecessors in terms of the number of annotations, which results in a higher readability of the ontology. This superiority is also reflected in the OQuARE Quality model, where OntoSLAM beats in features such as Knowledge Reuse, Consistent Search and Query, Operability, Analyzability, Testability, and Modifiability. For the rest of the characteristics, it performs the same as the predecessor ontologies with which it was compared.

From the simulated scenarios with ROS and Gazebo, it was demonstrated that no information is lost while transforming the information to the ontology instance and querying it afterwards. This achieves several benefits, such as: (i) the map can be partially constructed at certain moment, the partial map can be stored in the ontology, and continue the map construction in another later time; (ii) the map can be constructed by two different robots, at different times since the ontology takes over as the moderator; and (iii) a complete map can be recovered by other robots to do not repeat the SLAM process, and used it for other purposes (e.g., navigation).

5. Conclusions

In this work it is presented OntoSLAM, an ontology for modeling all aspects related to SLAM knowledge, in contrast of existing ontologies that only represent partially that knowledge, mainly focusing on the result of the SLAM process and neglecting the dynamic nature of the SLAM process. To be able to represent the SLAM knowledge considering all aspects, the model should include Robot Information, Environment Mapping, Time Information, and Workspace Information. The analysis performed in this work reveals that there is no a complete ontology covering these aspects of the SLAM knowledge. Therefore, OntoSLAM is proposed to solve this gap in the state-of-the-art. From the comparative evaluation of OntoSLAM with other ontologies, it is concluded that this approach outperforms its predecessors in all OQuARE Quality Metrics evaluated, without losing important information of Robotics. In particular, OntoSLAM overcomes in more than 22% its predecessors in the sub-characteristic of Knowledge Reuse; it is superior to its predecessors on Compatibility, Operability, and Transferability categories with a score of 97%; and it shows the best performance results in the Maintainability category. From the empirical evaluation using ROS, it is demonstrated that OntoSLAM is adaptable and compatible with any SLAM algorithm, which permits that this ontology can be released as a ROS package in the future to be used with any robot and SLAM algorithm.

The results of this work show that the semantic web is a way to standardize and formalize knowledge in Robotics, helping to improve the interconnection and interoperability among different robotic systems. It is possible to fit this semantic layer in the navigation stack of any robot that performs SLAM. The next steps of this work include the test of OntoSLAM together with other processes, such as perception and navigation, and the application of OntoSLAM into a wider variety of SLAM algorithms and robots.

Author Contributions: Conceptualization, M.A.C.-L., Y.C., R.T.-H. and D.B.-A.; Formal analysis, M.A.C.-L. and Y.C.; Funding acquisition, Y.C. and R.T.-H.; Investigation, M.A.C.-L., Y.C. and D.B.-A.; Methodology, M.A.C.-L. and Y.C.; Project administration, Y.C.; Software, M.A.C.-L., M.A. and J.D.-A.; Supervision, Y.C., R.T.-H. and D.B.-A.; Validation, M.A.C.-L., Y.C., D.B.-A., M.A. and J.D.-A.; Visualization, M.A. and J.D.-A.; Writing—original draft, M.A.C.-L. and Y.C.; Writing—review and editing, M.A.C.-L., Y.C. and R.T.-H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by FONDO NACIONAL DE DESARROLLO CIENTÍFICO, TECNOLÓGICO Y DE INNOVACIÓN TECNOLÓGICA-FONDECYT as executing entity of CONCYTEC under grant agreement no. 01-2019-FONDECYT-BM-INC.INV in the project RUTAS: Robots for Urban Tourism Centers, Autonomous and Semantic-based.

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

References

1. Coeckelbergh, M.; Pop, C.; Simut, R.; Peca, A.; Pintea, S.; David, D.; Vanderborght, B. A survey of expectations about the role of robots in robot-assisted therapy for children with ASD: Ethical acceptability, trust, sociability, appearance, and attachment. *Sci. Eng. Ethics* **2016**, *22*, 47–65. [[CrossRef](#)] [[PubMed](#)]
2. Ingrand, F.; Ghallab, M. Deliberation for autonomous robots: A survey. *Artif. Intell.* **2017**, *247*, 10–44. [[CrossRef](#)]
3. Thrun, S. Robotic Mapping: A Survey. In *Exploring Artificial Intelligence in the New Millennium*; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2003; pp. 1–35.
4. Manzoor, S.; Rocha, Y.G.; Joo, S.H.; Bae, S.H.; Kim, E.J.; Joo, K.J.; Kuc, T.Y. Ontology-Based Knowledge Representation in Robotic Systems: A Survey Oriented toward Applications. *Appl. Sci.* **2021**, *11*, 4324. [[CrossRef](#)]
5. 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](#)]
6. Cornejo-Lupa, M.; Ticona-Herrera, R.; Cardinale, Y.; Barrios-Aranibar, D. A categorization of simultaneous localization and mapping knowledge for mobile robots. In Proceedings of the ACM Symposium on Applied Computing, Brno, Czech Republic, 30 March–3 April 2020; pp. 956–963.
7. Cornejo-Lupa, M.A.; Ticona-Herrera, R.P.; Cardinale, Y.; Barrios-Aranibar, D. A Survey of Ontologies for Simultaneous Localization and Mapping in Mobile Robots. *ACM Comput. Surv. (CSUR)* **2020**, *53*, 1–26. [[CrossRef](#)]
8. Belouaer, L.; Bouzid, M.; Mouaddib, A.I. Ontology Based Spatial Planning for Human-Robot Interaction. In Proceedings of the Symposium on Temporal Representation and Reasoning, Paris, France, 6–8 September 2010; pp. 103–110.
9. Cashmore, M.; Fox, M.; Long, D.; Magazzeni, D.; Ridder, B.; Carrera, A.; Palomeras, N.; Hurtos, N.; Carreras, M. Rosplan: Planning in the robot operating system. In Proceedings of the International Conference on Automated Planning and Scheduling, Jerusalem, Israel, 7–11 June 2015; Volume 25, pp. 333–341.
10. 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](#)]
11. Chang, D.S.; Cho, G.H.; Choi, Y.S. Ontology-based knowledge model for human-robot interactive services. In Proceedings of the 35th Annual ACM Symposium on Applied Computing, Brno, Czech Republic, 30 March–3 April 2020; pp. 2029–2038.
12. Fortes, V. *A Positioning Ontology for C-SLAM*; Monograph; UFRGS—Curso de Bacharelado em Ciencia da Computacao: Rio Grande do Sul, Brazil 2013; pp. 23–33.
13. Tenorth, M.; Beetz, M. KNOWROB: Knowledge processing for autonomous personal robots. In Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St Louis, MO, USA, 10–15 October 2009; pp. 4261–4266.
14. Koenig, N.; Howard, A. Design and use paradigms for gazebo, an open-source multi-robot simulator. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No. 04CH37566), Sendai, Japan, 28 September–2 October 2004; IEEE: Piscataway, NJ, USA, 2004; Volume 3, pp. 2149–2154.
15. Schlenoff, C.; Messina, E. A Robot Ontology for Urban Search and Rescue. In Proceedings of the 2005 ACM Workshop on Research in Knowledge Representation for Autonomous Systems, Bremen, Germany, 4 November 2005; pp. 27–34.
16. Mozos, O.M.; Jensfelt, P.; Zender, H.; Kruijff, G.J.M.; Burgard, W. From Labels to Semantics: An Integrated System for Conceptual Spatial Representations of Indoor Environments for Mobile Robots. In Proceedings of the IROS 2007 Workshop: From Sensors to Human Spatial Concepts (FS2HSC), San Diego, CA, USA, 2 November 2007; pp. 25–32.
17. Suh, I.H.; Lim, G.H.; Hwang, W.; Suh, H.; Choi, J.H.; Park, Y.T. Ontology-based multi-layered robot knowledge framework (OMRKF) for robot intelligence. In Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, USA, 29 October–2 November 2007; IEEE: Piscataway, NJ, USA, 2007; pp. 429–436.
18. Eid, M.; Liscano, R.; El Saddik, A. A Universal Ontology for Sensor Networks Data. In Proceedings of the Computational Intelligence for Measurement Systems and Applications, Ostuni, Italy, 27–29 June 2007; pp. 59–62.
19. Lim, G.H.; Suh, I.H.; Suh, H. Ontology-Based Unified Robot Knowledge for Service Robots in Indoor Environments. *Syst. Man Cybern.* **2011**, *41*, 492–509. [[CrossRef](#)]
20. Dhoubi, S.; Du Lac, N.; Farges, J.L.; Gerard, S.; Hemaissia-Jeannin, M.; Lahera-Perez, J.; Millet, S.; Patin, B.; Stinckwich, S. Control architecture concepts and properties of an ontology devoted to exchanges in mobile robotics. In Proceedings of the 6th National Conference on Control Architectures of Robot, Grenoble, France, 24–25 May 2011; pp. 24–30.
21. Pronobis, A.; Jensfelt, P. Multi-modal semantic mapping. In Proceedings of the RSS Workshop on Grounding Human-Robot Dialog for Spatial Tasks, Los Angeles, CA, USA, 1 July 2011.
22. Wang, T.; Chen, Q. Object semantic map representation for indoor mobile robots. In Proceedings of the 2011 International Conference on System Science and Engineering, Macau, China, 8–10 June 2011; pp. 309–313.

23. Hotz, L.; Rost, P.; von Riegen, S. Combining qualitative spatial reasoning and ontological reasoning for supporting robot tasks. In Proceedings of the International Conference on Knowledge Engineering and Ontology Development, Barcelona, Spain, 4–7 October 2012; pp. 377–380.
24. Paull, L.; Severac, G.; Raffo, G.V.; Angel, J.M.; Boley, H.; Durst, P.J.; Gray, W.; Habib, M.; Nguyen, B.; Ragavan, S.V.; et al. Towards an Ontology for Autonomous Robots. In Proceedings of the Conference on Intelligent Robots and Systems, Algarve, Portugal, 7–12 October 2012; pp. 1359–1364.
25. Li, R.; Wei, L.; Gu, D.; Hu, H.; McDonald-Maier, K. Multi-layered map based navigation and interaction for an intelligent wheelchair. In Proceedings of the Robotics and Biomimetics, Shenzhen, China, 12–14 December 2013; pp. 115–120.
26. Carbonera, J.; Fiorini, S.; Prestes, E.; Jorge, V.; Abel, M.; Madhavan, R.; Locoro, A.; Gonçalves, P.; Haidegger, T.; Schlenoff, C. Defining positioning in a core ontology for robotics. In Proceedings of the 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, Tokyo, Japan, 3–7 November 2013; pp. 1867–1872.
27. Wu, H.; Tian, G.h.; Li, Y.; Zhou, F.y.; Duan, P. Spatial semantic hybrid map building and application of mobile service robot. *Robot. Auton. Syst.* **2014**, *62*, 923–941. [\[CrossRef\]](#)
28. Riazuelo, L.; Tenorth, M.; Di Marco, D.; Salas, M.; Gálvez-López, D.; Mösenlechner, L.; Kunze, L.; Beetz, M.; Tardós, J.D.; Montano, L.; et al. RoboEarth Semantic Mapping: A Cloud Enabled Knowledge-Based Approach. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 432–443. [\[CrossRef\]](#)
29. Burroughes, G.; Gao, Y. Ontology-Based Self-Reconfiguring Guidance, Navigation, and Control for Planetary Rovers. *J. Aerosp. Inf. Sys.* **2016**, *13*, 1–13.
30. Ramos, F.; Vázquez, A.S.; Fernández, R.; Olivares-Alarcos, A. Ontology based design, control and programming of modular robots. *Integr.-Comput.-Aided Eng.* **2018**, *25*, 173–192. [\[CrossRef\]](#)
31. Deeken, H.; Wiemann, T.; Hertzberg, J. Grounding semantic maps in spatial databases. *Robot. Auton. Syst.* **2018**, *105*, 146–165. [\[CrossRef\]](#)
32. Sun, X.; Zhang, Y.; Chen, J. High-Level Smart Decision Making of a Robot Based on Ontology in a Search and Rescue Scenario. *Future Internet* **2019**, *11*, 230. [\[CrossRef\]](#)
33. Crespo, J.; Castillo, J.C.; Mozos, O.; Barber, R. Semantic Information for Robot Navigation: A Survey. *Appl. Sci.* **2020**, *10*, 497. [\[CrossRef\]](#)
34. Joo, S.H.; Manzoor, S.; Rocha, Y.G.; Bae, S.H.; Lee, K.H.; Kuc, T.Y.; Kim, M. Autonomous navigation framework for intelligent robots based on a semantic environment modeling. *Appl. Sci.* **2020**, *10*, 3219. [\[CrossRef\]](#)
35. Karimi, S.; Iordanova, I.; St-Onge, D. An ontology-based approach to data exchanges for robot navigation on construction sites. *arXiv* **2021**, arXiv:2104.10239.
36. Shchekotov, S.; Smirnov, N.; Pashkin, M. The ontology driven SLAM based indoor localization technique. *J. Phys. Conf. Ser.* **2021**, *1801*, 012007. [\[CrossRef\]](#)
37. Cardinale, Y.; Cornejo-Lupa, M.; Ticona-Herrera, R.; Barrios-Aranibar, D. A Methodological Approach to Compare Ontologies: Proposal and Application for SLAM Ontologies. In Proceedings of the 22nd International Conference on Information Integration and Web-Based Applications & Services, Chiang Mai, Thailand, 30 November–2 December 2020; pp. 223–233.
38. Tanaka, H. Editdistance 0.3.1 PyPi. 2016. Available online: <https://pypi.org/project/editdistance/0.3.1/> (accessed on 17 November 2021).
39. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
40. Jian, N.; Hu, W.; Cheng, G.; Qu, Y. Falcon-AO: Aligning Ontologies with Falcon. In Proceedings of the Workshop on Integrating Ontologies, Banff, AB, Canada, 2 October 2005; pp. 87–93.
41. Duque-Ramos, A.; Fernandez-Breis, J.; Stevens, R.; Aussenac-Gilles, N. OQuARE: A SQuARE-based approach for evaluating the quality of ontologies. *J. Res. Pract. Inf. Technol.* **2011**, *43*, 159–176.
42. Krech, D. RDFlib: A Python Library for Working with RDF. 2006. Available online: <https://github.com/RDFLib/rdfliib> (accessed on 17 November 2021).
43. Grisetti, G.; Stachniss, C.; Burgard, W. Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling. In Proceedings of the 2005 IEEE International Conference on Robotics and Automation, Barcelona, Spain, 18–22 April 2005; IEEE: Piscataway, NJ, USA, 2005; pp. 2432–2437.
44. Hornung, A.; Wurm, K.M.; Bennewitz, M.; Stachniss, C.; Burgard, W. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Auton. Robot.* **2013**, *34*, 189–206. [\[CrossRef\]](#)