Semantic World Model for Off-Road Environments for Autonomous Navigation and Decision Making

Anant Bhamri[1], and Swaminathan Gopalswamy[1]

Abstract— **High terrain variability and lack of structure in off-road environments makes autonomous navigation a challenge. Even when employing a variety of sensors, perception algorithms at best provide geometric, segmented and classified information on the environment. In particular they lack the semantics that describe the implication of the information. For example, multiple factors such as terrain, agent capability, and weather conditions play a coupled and significant role in navigation. In this paper we propose to capture the semantics by representing knowledge on complex off-road environments through an ontology that facilitates description of classes of objects and the implicit relationships between them. Domain-specific ontologies to describe agent and weather semantics are also presented, along with the logical rules that define the coupling between these ontologies. This enables the use of automated semantic-reasoners to answer queries and support autonomous navigation and decision-making. We illustrate these concepts on a realistic off-road environment to support navigational planning for off-road autonomous vehicles.**

I. INTRODUCTION

Autonomous driving with and without human interactions is increasingly being explored in a variety of operations, in applications ranging from transportation [1] to disaster recovery support [2]. A critical enabler to successful decision-making in complex environments is a good understanding of the environment, which entails understanding the meaning (*semantics*) of that information in light of goals [3]. Representations to support such understanding (the *world model*) typically goes beyond storage of raw data to include abstractions and relationships between data elements [4], [5], [6]. Prior work in world modeling has broadly evolved in two distinct categories: (i) geometric modeling (such as point cloud, 2D grid, 3D voxel based representations of the world), with some segmentation and classification of parts of the world model in terms of physically recognizable entities (such as a building, or a tree, or grass, etc.) [7], [8], [9], [10], and (ii) object-focused modeling, where objects are defined in terms of their geometry and location, and the environment is a composition of the objects [11], [12], [13]. In both the approaches, the environment is described with sufficient levels of geometric and "textural" fidelity, so that vehicle navigation (planning and control decision making) is facilitated. However, the above approaches pay little attention to the implicit information that can be extracted from the non-geometric relationship between the objects in the world and the flow of knowledge between them, which enables inference of information not directly available from map data or perception. Any consideration for the semantic implication is hard-coded into the subsequent navigational algorithms.

To address this issue, use of knowledge frameworks have been explored. In [14], the robot knowledge is represented in layers: a diagnostic knowledge or observation layer, a common sense (or domain) knowledge layer, and lastly instance knowledge, to support task planning. The use of ontologies to represents the physical space using spatial information and semantic knowledge of the objects in the space for navigation and improving task planning capabilities by reasoning about the semantic information is explored in [15], [16], [17], [18], [19] and [20]. [21] explored the use of spatial semantic hierarchy for qualitative representations of space for robot exploration, using separate ontologies to capture different domain aspects (sensory, controls, topological) and their interactions. In [22] the authors explore the development of a knowledge framework that describes the information that a service robot can use in regards to perceptual features, and to answer queries about the relative positioning of objects. Though most of the above knowledge-framework based approaches do attempt to leverage the semantic information in the environment, they do not explicitly address the terrain of the environment, instead they focus on indoor and domestic settings. In our work we propose an ontological representation for complex unstructured off-terrain environments capturing the terrain information.

The use of multiple ontologies for inference and decision-making, as in [21] and [23] (for traffic systems) allows for modular development, and promotes reuse across development teams. In our proposed work, along with the ontology for off-road terrain and environment, we also describe ontologies for the agent and the weather as they relate to off-road navigation. By using the open and standard language OWL [24] we are able to leverage independently developed automated reasoners (e.g. [25], [26], [27]) to support decision making.

Our primary contribution is the development of an ontology-based world model that can capture information about complex, unstructured environments consisting of off-road terrains and the various agents that need to operate in it, and allowing the use of automated reasoners to support decision making. We demonstrate the suitability of the framework in a real-world off-road environment.

II. OFF-ROAD SEMANTIC WORLD MODEL

A. Ontologies

The primary development in this paper is the world model ontology, along with smaller supporting ontologies for the

[1]Anant Bhamri, and Swaminathan Gopalswamy are with Texas A&M University, College Station, TX, USA {anantbhamri, sgopalswamy}@tamu.edu

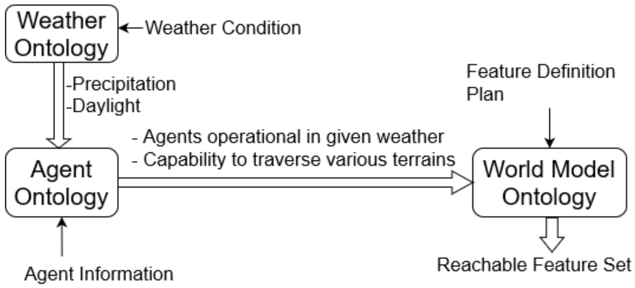


Fig. 1: *System Architecture showing how the different ontologies interact with each other. The information flows from the weather ontology to the agent ontology, which is used to add information to the world model to finally perform the reachabilty analysis.*

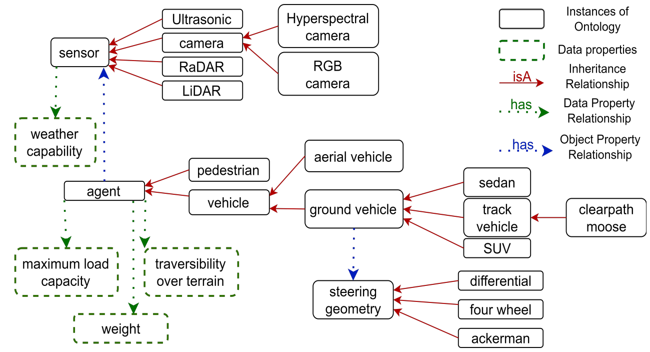


Fig. 2: *Hierarchical representation various ontological concepts used to define the agent ontology.*

weather and the agent. Fig. 1 shows the proposed framework and knowledge flow from the weather ontology to the agent ontology to the world model ontology. The Semantic Web Rule Language (SWRL) [28] is used to define the rules acting as constraints on the ontologies.

1) Weather and Agent Ontologies: We develop a minimal ontology for the weather, focusing on concepts and functional relationships that directly impact navigation capabilities of the various agents. Specifically, whether it is rainy, and how rain impacts terrain traversibility and sensor functioning.

Weather Ontology Concepts, inspired by [23], define the state of the weather and utilize key aspects that are relevant to the application, like rain, sunny, snowy, etc.

Agent Ontology Concepts categorize the agent based on the mode of locomotion (pedestrian, ground vehicle, or an aerial vehicle), the powertrain and steering architecture, the sensors they could carry (cameras, LiDARs or RaDARs), etc. Some example of properties and functional relations associated with it are:

- *weatherCapability* - Data property specifying if a *sensor* can be used in a given weather condition like rain.
- *traversibilityGravelRain* - Data property that defines if the agent can traverse the terrain type gravel in rain. Similar properties are defined for each instance of the agent for the various possible combination of the terrain and the weather.

Run Time Rules: The run time rules are used to specify the usable agents in various weather conditions.

Examples of run time rules in the agent ontology:

- Only waterproof sensors can operate in rainy conditions.
- For an agent to be deemed capable to perform a mission either one of it's sensors must work in the weather.

Figure 2 shows some of the properties and the instances used in the *agent* ontology.

2) World Model Ontology:

Ontological Concepts: The two main *Entities* defined are:

- *feature* - Subset of the world; represents human recognizable elements on the ground, such as buildings, grassy fields or rivers. It inherits from *terrain*, and *geometry* class.
- *terrain* - Defines terrains, such as *concreteTerrain*, *grassTerrain*, *mudTerrain*, etc. which are observed in off-road unstructured environments [29]. Subclass of *properties*. It has data properties to provide a numerical value for traversibility.

Figure 3 shows a subset of the inheritance relationships, and object and data properties of the test examples.

Functional relations: They are defined using *Object and Data Properties*. Examples:

- *isNeighborOfFeature* - Adjacency relationship between the *feature*s. This is a symmetric relation.
- *plan* - Asserts which *feature* is the *source*, *destination*, and which needs to be avoided, *avoid*.
- *canReachFeature* - Ability to reach one *feature* from another, for a given agent type. This is not usually explicitly defined in the structural definition, but rather inferred from the traversibility properties and other information like the neighbors of a feature. This is a transitive relation.

Run Time Rules: Some key rules are categorized by their purposes as below:

- *Ontology Extensions for World Model*:
  - Traversibility of a feature, inheritable from *terrain*.
  - If the maximum supported weight of a *feature* is greater than the maximum weight of the agent then the agent can use the *feature*, else it is asserted to be *avoided*.
- *Reachability Rules*:
  - If two neighboring *feature*s are not marked to be avoided and are both *traversible*, then the two *features* are reachable from each other.
- *Planning Assistance Rules*: A key objective of the mission is to plan possible pathways from the set of source *features* to the set of destination *features*.
  - A *feature* that can be reached from any *feature*, that is part of the *source* set, is assigned to the *reachabilitySet*.
  - Planning can be significantly enhanced by knowing the set of *deadEnds*, i.e. the *features* that can be completely eliminated without it impacting the ability to perform a mission. Rules are defined to assert the conditions under which a *feature* can be a *deadEnd*. For example, if a *feature* that is not part of the *source*

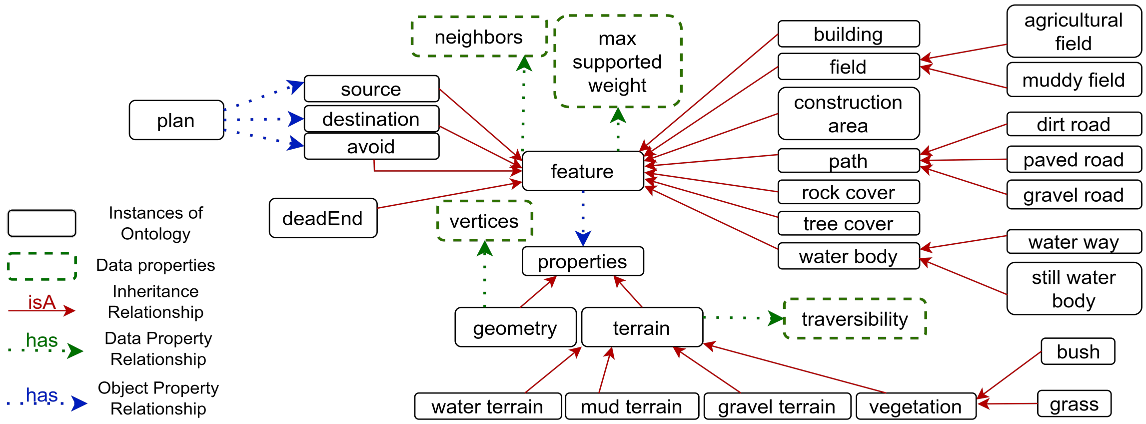


Fig. 3: *Hierarchical representation of the containment specifications and how the classes inherit from one another. This in conjunction to the object and data properties forms the ontological representation for the world model.*

has two neighbors, and one of the neighbors is *not-traversible*, then other one is a *deadEnd*.

- Similarly, if a *feature* has n distinct neighbors and $n-1$ of the neighbors are *deadEnd*s, then the *feature* is also a *deadEnd*.
- Similar rules are added to accommodate all the possible combinations when the neighbors are either *non-traversible* or to be *avoided* or *deadEnds*.

B. Ontology Instantiation

We initially add individuals to the above mentioned ontologies separately. The weather ontology is initialised using the weather conditions of the location of interest at the time the navigational activity needs to be performed. The agent ontology is instantiated with all the available agents. We populate the type of sensors that each agent has, agent size and weight, and its traversibility value over various terrains in various weather conditions. Next, we instantiate the world model ontology using the geometric, geographic and semantic information of the environment, which is obtained from segmenting aerial images of the environment [30]. This is defined by the geographic location and shape, which is represented using polygons. Lastly, a plan is defined with the starting and end location that needs to be reached. In addition, we can also define features that need to be avoided, depending on the task at hand.

III. Decision support using Semantic Models

The ontologies are reasoned over using the Pellet reasoning engine [25]. The Algorithm 1 shows how the reasoning uses multiple ontologies. First, reasoning is performed using on the agent ontology to find the list of agents that can operate in the given weather condition. Next, for each agent in the set of agents, we define the traversibility values for the various terrain types in the environment. This is necessary as these values change based on the combination of the agent, the terrain and the weather. Lastly, the reasoner is run to find all the features reachable from the source, and if the destination is reachable from the source, the agent is specified as capable to perform the mission.

Algorithm 1 Reachability Analysis

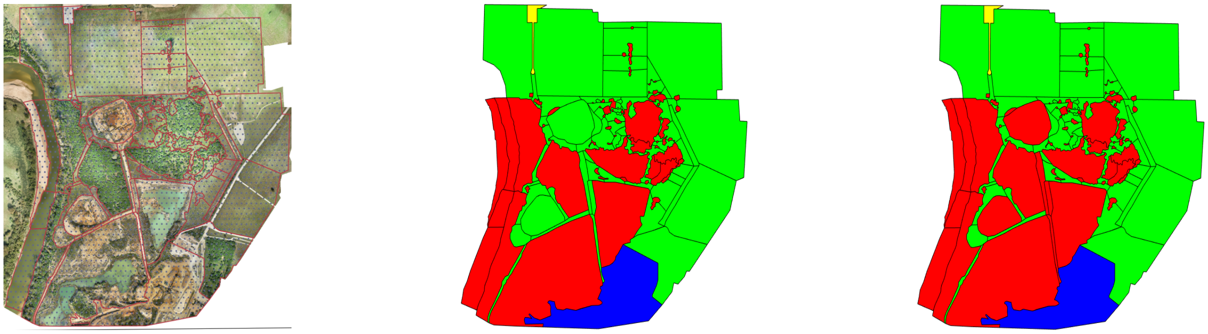
Define the weather type in the agent ontology using weather ontology
Run Reasoner on agent ontology (Find the set of agents which have sensors usable in the defined weather)
for all agents in set **do**:
    Define traversability values for all terrains
    Run Reasoner (Check Reachability)
    if Destination in Reachability Set **then**
        Mark agent as capable of mission
        Generate OG based input / graph for low level plan
    end if
end for

IV. Experiments

We apply the semantic world model framework developed above to an off-road testing facility at the Texas A&M University, Rellis Campus, College Station, Texas (Fig.4a). The area is 1.5km x 1.5km, previously a sand-quarry, and provides a lot of complexity, with muddy terrain, dense and sparse trees and shrubs, off-road trails, etc. The bottom right had some on-going construction activity, and in the north end is the roadway connecting the region to the on-road areas.

A fleet of three vehicles (agents) is instantiated to perform the testing: a JEEP (ackerman steering vehicle), a Polaris ATV (off-road vehicle with ackerman steering) and a Clearpath Moose (off-road, multi-axled vehicle with differential steering). The Moose and the JEEP have all three sensors: cameras, LiDARs and RaDARs mounted and the Polaris only has a LiDAR mounted on it. For generating the world model ontology, we used an aerial image of the environment, and segmented it into the different regions of interest, shown in Fig. 4a. The world model ontology is instantiated with this information, and a mission is defined to go from the construction site to the road on the north in rainy weather.

The automated reasoning was performed on a PC with the following specifications: (i) System: 4 Core, 1.8GHz (i7-8550U) with 16GB RAM(on Windows 11 OS), (ii) Software: OWLAPI version 5.1.17 [31], and the OPENLLET (PELLET) [32] reasoner v. 2.6.5 using JAVA for the programming.



(a) The aerial image of the off-road testing overlaid with the various human-recognisable features in the environment. Each feature has a semantic class associated to it. The mission is to go from the construction site in the bottom right corner to the road in the north, in rainy conditions. The decision to make is - which vehicle can do that. Can the reasoner determine that the Moose can traverse the muddy trails between the water bodies but the JEEP cannot do that, and the Polaris does not have the sensors to operate on rain?

(b) Reachability analysis results for the Clearpath Moose. Along with removing non-traversible regions, the reachability analysis also removes dead-Ends, such as the feature to the west of the river. (A total of 67 features are removed). Only 66 features remain in the reachability set, reducing the search space for the local planners. As the destination is inside this set, we conclude that the Moose can reach the destination from the construction side.

(c) Compared to the Figure 4b, additional features are marked to be avoided for the JEEP. These features have muddy terrain, and the JEEP is not capable of driving in muddy terrain during the rain. It is this logical inference that is enabled by the use of a declarative ontological representation of the world model in conjunction with automated reasoners, instead of being hard-coded into planning algorithms. There are a total of 61 features out of 133 total features that are in the reachability set for the JEEP.

Avoid Reachable Source Destination

Fig. 4: *Reachability analysis performed using the proposed framework to first infer which vehicle will be able to autonomously navigate during rainy conditions. In the example, the only possible path for JEEP to reach the destination would be to first go up north and then east, where as the Moose also has an option to go north-east using a muddy trail to reach the destination, which can be seen in their respective reachability sets.*

A. Reasoning for Decision Support

The reasoners were used to check for consistency and generate inference, on the following questions:

- Find the list of agents capable of operating in a given weather condition, specifically "rainy" condition.
- Given a source region, find the *reachabilitySet* from source. This set contains all the features that can be reached from the source. If this set contains the destination then we infer that we can reach the destination. We further remove the *features* inferred to be *avoided* from this to get the smaller subset of the traversibility graph or a cost map with a lot more occupied cells, which is used to plan the optimal path from source to destination.

B. Results

We observed that due to Polaris just having a LiDAR, was deemed incapable of performing the mission after the reasoning on the agent ontology. This happened as it's only sensors performance degrades in rainy conditions. Next, we ran the reasoner on the world model ontology for the JEEP and the Moose to find the reachability sets for them. As, the Moose is able to traverse muddy terrain even in rainy conditions, the reachability set for it is larger than the reachability set for the JEEP. As the JEEP is unable to traverse muddy terrain in rains, the muddy trails and the excavated mine areas were inferred as additional avoid features. In Figure 4c we can see that for the JEEP we have additional features which are marked as ones to be avoided as compared to the one for the Moose, Figure 4b.

The reasoning takes around 30 seconds for each agent. The environment has a total of 133 features. A traversability graph with the features as nodes and neighbors as edges has 133 nodes and 694 edges. The reachable set of the Moose and JEEP are reduced to 66 and 61; reduced to graph with (66 nodes, 286 edges) and (61 nodes, 224 edges) respectively, reducing the search space for the local planner considerably.

V. Conclusion

We have proposed a world modeling framework using multiple ontologies for automated-reasoning-based planning for autonomous agents in off-road environments. Ontologies for the environment, the agent, and weather are defined, that describe the world in terms of human understandable features, off-road terrain, and their properties. Functional relationships between the various entities are defined so that automated reasoners could be used to make meaningful insights about the environment and answer queries to support decision making for off-road navigation. We illustrate the value of the approach by applying the framework on a real off-road environment and programmatically using an automated reasoner (PELLET) to answer traversibility questions that are simultaneously dependent on the environment (terrain), the agent, and the weather.

Future work would address uncertainty in the definitions of the axioms and properties of the ontology, and the expected growth in computational complexity with scale. Additionally, we intend to work with collaborators to arrive at an open-standard for off-road ontology.

ACKNOWLEDGMENT

The authors gratefully acknowledge valuable discussions with Dr. Dylan Shell of Texas A&M University and Dr. Robert St.Amant from DEVCOM Army Research Laboratory. This work was supported by Army Research Laboratories under contract W911NF1920243.

REFERENCES

[1] J. Levinson, J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt, M. Sokolsky, G. Stanek, D. Stavens, A. Teichman, M. Werling, and S. Thrun, "Towards fully autonomous driving: Systems and algorithms," in *2011 IEEE Intelligent Vehicles Symposium (IV)*, 2011, pp. 163–168.

[2] L. Matthies, Y. Xiong, R. Hogg, D. Zhu, A. Rankin, B. Kennedy, M. Hebert, R. Maclachlan, C. Won, T. Frost, G. Sukhatme, M. McHenry, and S. Goldberg, "A portable, autonomous, urban reconnaissance robot," *Robotics and Autonomous Systems*, vol. 40, no. 2, pp. 163–172, 2002, intelligent Autonomous Systems - IAS -6. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889002002415

[3] M. R. Endsley, "Toward a theory of situation awareness in dynamic systems," *Human Factors*, pp. 32–64, 1995.

[4] E. Miguelanez, P. Patron, K. E. Brown, Y. R. Petillot, and D. M. Lane, "Semantic knowledge-based framework to improve the situation awareness of autonomous underwater vehicles," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 5, pp. 759–773, 2010.

[5] N. Dahn, S. Fuchs, and H.-M. Gross, "Situation awareness for autonomous agents," in *2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2018, pp. 666–671.

[6] N. Tsiogkas and H. Bruyninckx, "Towards a model for situation awareness for autonomous robots," in *RobOntics 2021, Location: Free University of Bozen-Bolzano*, 2021.

[7] S. Thrun and M. Montemerlo, "The graph slam algorithm with applications to large-scale mapping of urban structures," *The International Journal of Robotics Research*, vol. 25, no. 5-6, pp. 403–429, 2006. [Online]. Available: https://doi.org/10.1177/0278364906065387

[8] S. Thrun, "Robotic mapping: A survey," in *Exploring Artificial Intelligence in the New Millenium*, G. Lakemeyer and B. Nebel, Eds. Morgan Kaufmann, 2002, to appear.

[9] J. Ryde and H. Hu, "3d mapping with multi-resolution occupied voxel lists," *Autonomous Robots*, vol. 28, no. 2, pp. 169–185, 2010.

[10] A. Shaban, X. Meng, J. Lee, B. Boots, and D. Fox, "Semantic terrain classification for off-road autonomous driving," in *Proceedings of the 5th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, A. Faust, D. Hsu, and G. Neumann, Eds., vol. 164. PMLR, 08–11 Nov 2022, pp. 619–629. [Online]. Available: https://proceedings.mlr.press/v164/shaban22a.html

[11] D. Wolter and L. J. Latecki, "Shape matching for robot mapping," in *Pacific Rim international conference on artificial intelligence*. Springer, 2004, pp. 693–702.

[12] D. Wolf, A. Howard, and G. Sukhatme, "Towards geometric 3d mapping of outdoor environments using mobile robots," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005, pp. 1507–1512.

[13] S. Ikeda and J. Miura, "3d indoor environment modeling by a mobile robot with omnidirectional stereo and laser range finder," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006, pp. 3435–3440.

[14] M. Hanheide, M. Göbelbecker, G. S. Horn, A. Pronobis, K. Sjöö, A. Aydemir, P. Jensfelt, C. Gretton, R. Dearden, M. Janicek, *et al.*, "Robot task planning and explanation in open and uncertain worlds," *Artificial Intelligence*, vol. 247, pp. 119–150, 2017.

[15] M. Tenorth and M. Beetz, "Knowrob — knowledge processing for autonomous personal robots," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 4261–4266.

[16] M. Tenorth, "Knowledge processing for autonomous robots," Ph.D. dissertation, Technische Universität München, 2011.

[17] M. Tenorth and M. Beetz, "Knowrob—knowledge processing for autonomous personal robots," in *2009 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2009, pp. 4261–4266.

[18] M. Tenorth, L. Kunze, D. Jain, and M. Beetz, "Knowrob-map - knowledge-linked semantic object maps," in *2010 10th IEEE-RAS International Conference on Humanoid Robots*, 2010, pp. 430–435.

[19] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J.-A. Fernandez-Madrigal, and J. González, "Multi-hierarchical semantic maps for mobile robotics," in *2005 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2005, pp. 2278–2283.

[20] C. Galindo, J.-A. Fernández-Madrigal, J. González, and A. Saffiotti, "Robot task planning using semantic maps," *Robotics and autonomous systems*, vol. 56, no. 11, pp. 955–966, 2008.

[21] B. Kuipers, "The spatial semantic hierarchy," *Artificial Intelligence*, vol. 119, no. 1, pp. 191–233, 2000. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0004370200000175

[22] G. H. Lim, I. H. Suh, and H. Suh, "Ontology-based unified robot knowledge for service robots in indoor environments," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 41, no. 3, pp. 492–509, 2011.

[23] W. Chen and L. Kloul, "An ontology-based approach to generate the advanced driver assistance use cases of highway traffic," in *10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management.*, 2018.

[24] D. L. McGuinness, F. Van Harmelen, *et al.*, "Owl web ontology language overview," *W3C recommendation*, vol. 10, no. 10, p. 2004, 2004.

[25] E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, and Y. Katz, "Pellet: A practical owl-dl reasoner," *Journal of Web Semantics*, vol. 5, no. 2, pp. 51–53, 2007.

[26] R. Shearer, B. Motik, and I. Horrocks, "Hermit: A highly-efficient owl reasoner." in *Owled*, vol. 432, 2008, p. 91.

[27] D. Tsarkov and I. Horrocks, "Fact++ description logic reasoner: System description," in *International joint conference on automated reasoning*. Springer, 2006, pp. 292–297.

[28] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, M. Dean, *et al.*, "Swrl: A semantic web rule language combining owl and ruleml," *W3C Member submission*, vol. 21, no. 79, pp. 1–31, 2004.

[29] W. Smith, D. Grabowsky, and D. Mikulski, "Atlas, an all-terrain labelset for autonomous systems," *Ground Vehicle Systems Engineering and Technology Symposium (GVSETS)*, 2022.

[30] A. Medellin, A. Bhamri, R. Langari, and S. Gopalswamy, "Real-time semantic segmentation using hyperspectral images for mapping unstructured and unknown environments," 2023.

[31] M. Horridge and S. Bechhofer, "The owl api: A java api for owl ontologies," *Semantic web*, vol. 2, no. 1, pp. 11–21, 2011.

[32] G. Singh, S. Bhatia, and R. Mutharaju, "Owl2bench: a benchmark for owl 2 reasoners," in *The Semantic Web–ISWC 2020: 19th International Semantic Web Conference, Athens, Greece, November 2–6, 2020, Proceedings, Part II 19*. Springer, 2020, pp. 81–96.