applied sciences

MDPI

Article

PDDL Planning with Natural Language-Based Scene Understanding for UAV-UGV Cooperation

Jiyoun Moon * and Beom-Hee Lee

Automation and Systems Research Institute, Department of Electrical and Computer Engineering, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul 08826, Korea

* Correspondence: jiyounmoon@snu.ac.kr; Tel.: +82-10-9911-3572

Received: 13 August 2019; Accepted: 29 August 2019; Published: 10 September 2019

Abstract: Natural-language-based scene understanding can enable heterogeneous robots to cooperate efficiently in large and unconstructed environments. However, studies on symbolic planning rarely consider the semantic knowledge acquisition problem associated with the surrounding environments. Further, recent developments in deep learning methods show outstanding performance for semantic scene understanding using natural language. In this paper, a cooperation framework that connects deep learning techniques and a symbolic planner for heterogeneous robots is proposed. The framework is largely composed of the scene understanding engine, planning agent, and knowledge engine. We employ neural networks for natural-language-based scene understanding to share environmental information among robots. We then generate a sequence of actions for each robot using a planning domain definition language planner. JENA-TDB is used for knowledge acquisition storage. The proposed method is validated using simulation results obtained from one unmanned aerial and three ground vehicles.

Keywords: mission planning; language descriptions; semantic graphs; autonomous robots; artificial intelligence

1. Introduction

Natural language-based scene understanding is a critical issue for symbolic planning for heterogeneous multi-robot cooperation. We can mitigate the knowledge acquisition problem associated with the area of symbolic planning by sharing the environmental information expressed in natural language with diverse robots. Recently, heterogeneous multi-robot systems composed of robots with different abilities have received increasing attention as they are required in a broad range of fields such as surveillance, environment exploration, and field robotics [1]. Various symbolic planning studies have been conducted to generate a sequence of actions for each robot to achieve success in a shared mission. In particular, planning domain definition language (PDDL) is used as a standardized artificial intelligence planning language [2] and provides flexibility when planning actions for robots to achieve mission goals [3]. Miranda et al. [4] embedded a symbolic task planner using PDDL in the robot operating system (ROS) for multi-robot navigation. Zhang et al. [5] presented a multi-robot symbolic planning system with an iterative interdependent algorithm to find the optimal plans that minimize overall cost. Compared to many studies that aimed to maximize overall utility and reduce costs during identification of optimal plans for multi-robots, the environmental information sharing method between robots can mitigate environmental knowledge acquisition problems but continues to be insufficiently studied. We can solve various mission planning problems by allowing robots to find the environmental data of unmodeled objects and sharing them. Robots can gather information about early unmodeled objects, extract meaningful information from them, and share them to solve various mission planning problems, particularly in problems such as finding survivors in wildfire areas or spotting

leaky gas lines. Through data sharing from robots in unknown environmental exploration works, it is possible to secure work efficiency and system flexibility of the entire robot. Corah et al. [6] employed a Gaussian mixture model to map the surrounding environment while maintaining a low volume of memory for communication-efficient planning. However, since this method uses an algorithm designed for a specific sensor, it poses a practical application issue for a heterogeneous multi-robot system composed of different processors, implementation techniques, and sensors. Moreover, since these methods share raw sensor data, the additional process needed to achieve meaningful information from the sensor data imparts inefficiency to the overall process. Therefore, in this paper, we propose a symbolic planning method that shares natural language-based environment information containing semantic meaning, rather than raw sensor data, for heterogeneous multi-robot cooperation.

Semantic scene understanding via objects or natural languages, rather than points, lines, and planes that cannot contain semantic meanings, has been widely researched in robotics and computer vision [7–9]. The conventional mission planning methods hardly consider unmodeled objects; thus, the unmodeled objects are handled by the motion planning on the basis of maps with points, lines, and planes. With the assistance of the object-oriented semantic graph map in various forms, unmodeled objects from the dynamic environment can be considered when robots generate plans to achieve goals. Zhang et al. [10] generated object-level entities using the semantic simultaneous localization and mapping (SLAM) algorithm. Karpathy and Fei-Fei [11] generated dense captions for multiple regions and the overall area in an image using bidirectional recurrent neural networks (RNNs) and a multimodal RNN. Yao et al. [12] found semantic and spatial relationships between objects in images through graph convolutional networks (GCNs) and long short-term memory (LSTM). The results of this semantic information are utilized for various applications such as robot navigation [13], image retrieval [14], and question and answer functions [15]. However, the application of heterogeneous multi-robot cooperation planning is not considered.

On the one hand, deep learning outperforms extraction of semantic information from an unseen environment, but it is difficult to learn high-level processes that require causal reasoning, analogical reasoning, or planning using data [16]. On the other hand, symbolic planning that uses a logic model can guarantee solution optimality, but it can only be applied to a predefined environment. To combine deep learning and classical planning, Asai and Fukunaga [17] encoded images as latent vectors with a variational autoencoder and applied PDDL planning. Mao et al. [18] proposed a neuro-symbolic concept learner that learns visual scenes using a neural network and expresses them in an executable form in symbolic programs. In this study, symbolic planning and deep learning techniques are integrated to propose a cooperation planning architecture with natural language scene understanding for a heterogeneous robot team, as shown in Figure 1. Convolutional neural networks (CNNs), GCNs, and RNNs are used for natural language description and scene graph generation. JENA-TDB is used to share the semantic representation of the environment among the robots. The planning phase of ROSPlan [19] is used for generating plans. The proposed method is verified by a simulation using one unmanned aerial vehicle (UAV) and three unmanned ground vehicles (UGVs).

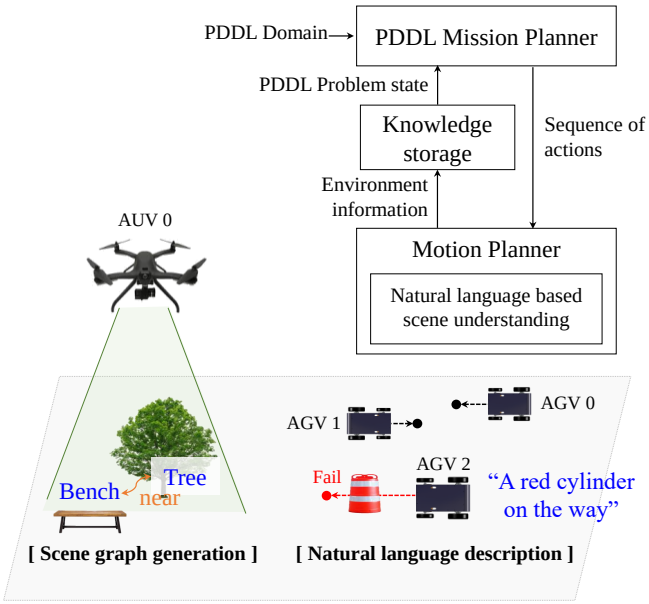


Figure 1. Outline of our approach: PDDL mission planner is utilized to generate the sequence of actions using the PDDL domain and environment information for heterogeneous robots. The surrounding environment is represented as a scene graph. If a robot fails the mission, it generates a natural language description.

2. Related Work

This paper is related to studies of heterogeneous multi-robot cooperation planning and natural language-based semantic scene understanding, the idea being to connect symbolic planning and deep learning.

2.1. Heterogeneous Multi-Robot Cooperation Planning

The multi-robot system has the advantage that it can perform complex tasks that cannot be accomplished by one single powerful robot with many capabilities through cooperation [20]. For example, a large building can be cleaned with one robot, but it is time-consuming and unrealistic. Thus, a multi-robot system that dispatches the overall mission into smaller sub-problems to individual robots is necessary. Rosa et al. [1] proposed a cooperative control scheme for a heterogeneous ground-air robot team. Wurm et al. [21] integrated a temporal planning approach with a PDDL planner for heterogeneous teams of robots. Jang et al. [22] solved the decision-making issues of aerial robots using an integrated decision-making framework. Kingry et al. [23] represented the environment in a scalar field and created a time-optimized mission plan for UGVs using a cascaded heuristic optimization algorithm. However, most studies of heterogeneous multi-robot systems focus on achieving shared goals effectively, with minimum time and cost, through algorithms rather than acquiring knowledge of the environment using multiple robots.

Some researchers have attempted to solve the environmental knowledge acquisition problem through data sharing among the robots. Reis et al. [24] used an adaptive transmission method for efficient distributed information sharing. Jiang and Lu [25] proposed a shared information integration method for cooperative environmental data gathering. Foerster et al. [26] introduced two approaches that could learn how information may be shared: reinforced inter-agent learning and differentiable inter-agent learning. These studies shared raw sensor information that could hardly infer semantic meanings without algorithms. They had to be designed suitably for the individual robots in a heterogeneous multi-robot team. Unlike conventional studies, sharing information embedded with semantic meaning in the form of natural language can enable the heterogeneous robots to easily understand and communicate with each other. Moreover, we can decrease the quality of service

problem, which is often observed in field robotics, by transmitting a compact representation of the environmental information. We introduce a method that acquires environmental knowledge in the form of natural language and applies it to multi-robot cooperation planning.

2.2. Natural Language-Based Scene Understanding

Many studies on robotics have proposed graph-based SLAM using semantic scene understanding and various sensors. Himri et al. [27] performed object recognition using range data and feature-based semantic SLAM with a UAV. Li et al. [28] proposed a dense 3D SLAM system composed of stereo-ORB-SLAM and a CNN for a traffic environment. Mao et al. [29] combined a matured SLAM system named RTAB-Map and a CNN to utilize depth image information. However, they rarely considered the natural language inference problem, which is important in multi-robot communication.

However, semantic scene understanding using natural languages such as image captioning, visual question and answering (VQA), and scene graph generation is widely studied in the field of computer vision. Lu et al. [30] generated image captions using an attention-based neural encoder-decoder framework. Lu et al. [31] utilized a co-attention model in a hierarchical fashion to perform VQA. Dai et al. [32] proposed a deep relational network that can exploit the statistical dependencies of detected objects and their relationships. Since these approaches use images as inputs, graph maps, which are widely used as environment representation by robots, are rarely utilized. This paper proposes an architecture that includes natural language description and scene graphs generated using a graph map in multi-robot planning.

2.3. Connecting Symbolic Planning and Deep Learning

Many studies of robotics involving mission planning with symbolic planners have been conducted. Srivastava et al. [33] demonstrated off-the-shelf task implementation with a PDDL planner. Dornhege et al. [34] applied geometric reasoning to symbolic planning and conducted real-world mobile manipulation experiments. Manso et al. [35] utilized graph models and graph rewriting rules with a symbolic planner for human–robot interaction. However, symbolic planning is hardly applied to new, unforeseen, and dynamic environments, because the environments should be modeled directly by a human or via a compiler. However, deep learning, which is a data-driven approach, has shown outstanding performance in environmental cognition [36–38]. To take advantage of both fields, Zhang and Sornette [39] introduced a deep symbolic network to represent any knowledge as a symbol. Liao and Poggio [40] converted objects into symbols using an object-oriented deep learning algorithm. They focused on generating symbols using deep learning, rather than setting the overall architecture for planning. In this study, we propose a method to bridge the gap between symbolic planning and deep learning techniques, and verify it using heterogeneous multi-robot cooperation planning.

3. Architecture

This section explains the framework devised to connect deep learning techniques and the symbolic planner for cooperation among heterogeneous agents. Unlike conventional planning systems for robots [19], our framework entails natural language-based cognition and a knowledge engine for multiple agents. The general overview of the framework is shown in Figure 2. It is composed of perception, cognition, planning, coordination, execution, and memory storage. Perceptively, sensor information obtained from environments is continuously passed to cognition. During cognition, scene understanding-based natural language is created by generating language description and scene understanding using deep learning techniques. Then, the generated semantic information is passed on to the knowledge engine while raw sensor data are sent to episodic memory storage. Using the episodic memory and knowledge collected from multiple robots, the PDDL planning agent builds a sequence of actions for each agent. Then, the robots complete the required actions through coordination and execution. The details are as illustrated in Figures 3 and 4.

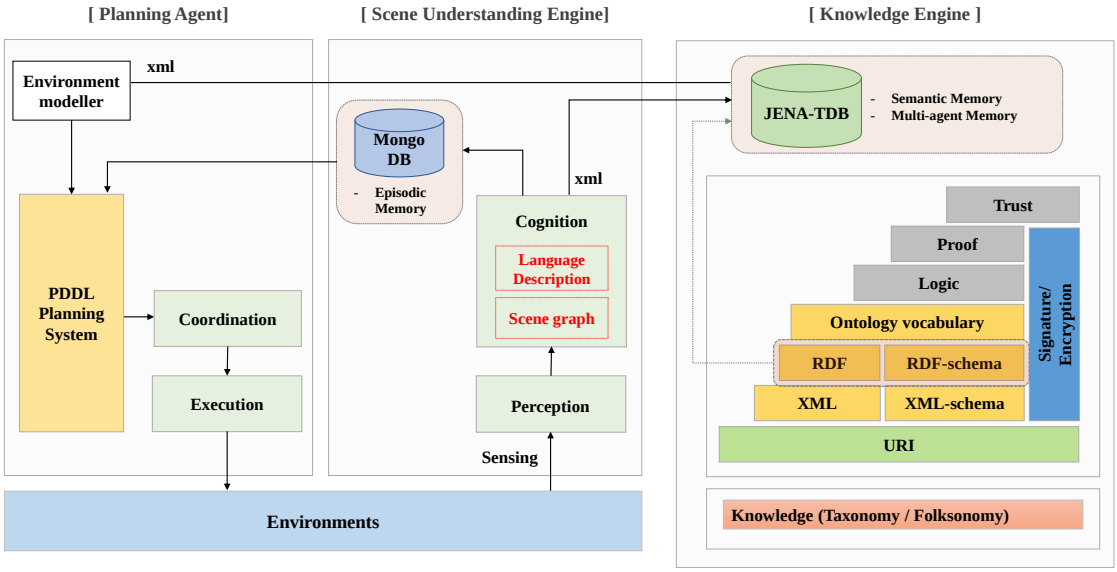


Figure 2. General overview of the proposed architecture.

3.1. Natural Language-Based Cognition

Cognition part in scene understanding engine is largely composed of semantic graph generation, language description, and scene graph generation as shown in Figure 3. To understand the surrounding environment in natural language, we generate a natural language description and scene graph. In this study, we assume that the robots use a graph map (for motion planning) generated using semantic SLAM, which is a widely used environment representation method in robotics [7]. To utilize the graph map $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ that contains features and positions of the detected object as nodes $v_i \in \mathcal{V}$ and their relationships as edges $e_{ij} = (v_i, v_j) \in \mathcal{E}_{ij}$, we closely follow Moon and Lee [41] for generating the language description and graph inference phase of Xu et al. [42] for the scene graph generation. However, since the edge information of the graph map is binary, which can only infer whether a connection exists or not, or a weighted value that indicates relations such as the Euclidean distance between objects, it is difficult to find the semantic meaning. Therefore, we additionally extract features of the union region of two objects for edge information. For each v_i and e_{ij}, features are extracted using VGGNet [43]. f_i^v is the feature vector of v_i, and f_{ij}^e is the feature vector of e_{ij}. p_i is position vector of v_i.

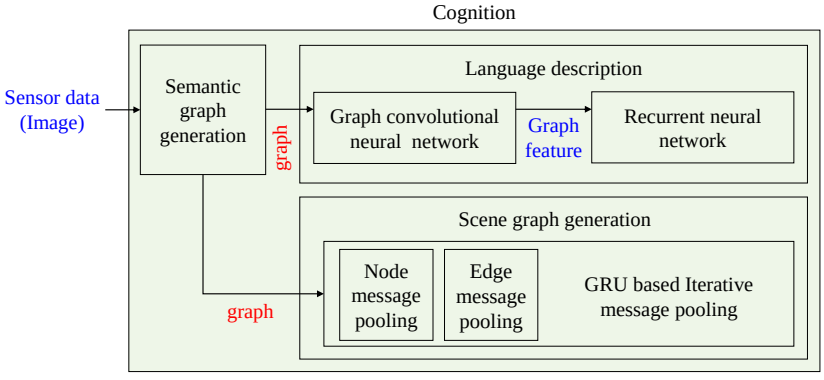


Figure 3. On language description part, a GCN extracts features from the graph map. The extracted graph feature is concatenated with a word and feed into the RNN as input. Then, the RNN generates sentence attention over the graph. On scene graph generation part, Two different message pooling methods are performed. Node message pooling uses the inbound and outbound edge states with a node. Edge message pooling uses the object states with an edge. This process is repeated to precisely predict the natural language words corresponding to the nodes and edges of the graph.

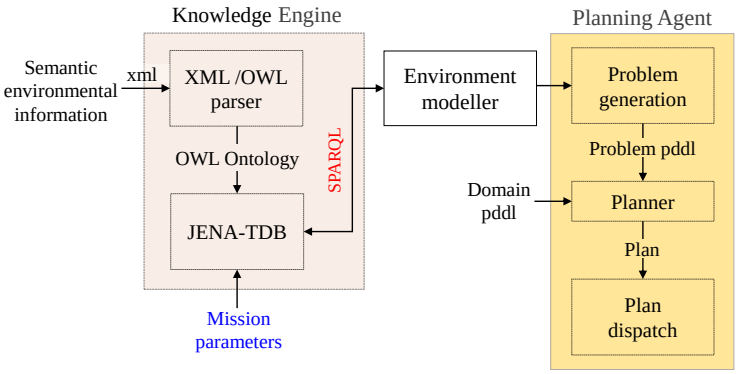


Figure 4. Detailed architecture of the knowledge engine, environment modeler, and PDDL agent.

A GCN with graph convolution layers defined by spectral graph theory and fully connected layers is utilized to extract features from irregular and non-Euclidean graphs. Then, an RNN is used to generate a language description over the graph. The RNN takes the encoded graph features concatenated with a word vector and predicts the probabilistic distribution of the target word vector. Given that we also back-propagate the GCN when training the RNN, we can expect that graph features that fit the generated sentence will be extracted. The generated description can be used to understand the surrounding environment when an unexpected situation occurs.

Scene graph generation involves the process of finding appropriate words corresponding to each node and edge of the graph. We denote variables that need to be predicted as $\mathbf{g} = (v_i^{class}, e_{ij} \mid i = 1 \ldots n, j = 1 \ldots n, i \neq j)$, where $\mathcal{C}$ is a set of object classes and $\mathcal{R}$ is a set of relationship types, $v_i^{class} \in \mathcal{C}, e_{ij} \in \mathcal{R}$. The optimal g^* is found as follows:

$$\mathbf{g}^* = argmax_{\mathbf{g}} Pr(\mathbf{g} \mid f_i^v, f_{ij}^e) \tag{1}$$

$$Pr(\mathbf{g} \mid f_i^v, f_{ij}^e) = \prod_{i \in V} \prod_{j \neq i} Pr(v_i^{class}, e_{ij} \mid f_i^v, f_{ij}^e) \tag{2}$$

The iterative message pooling method based on the gated recurrent unit (GRU) is utilized. Edge features and node features are fed into the edge GRU and node GRU as the initial value, respectively. After the message pooling, the edge message is fed into the edge GRU and the node message is fed into the node GRU. The iteration that follows precisely predicts words for the nodes and edges. The scene graph can be used to gather environmental information in natural language for large and unstructured environments.

3.2. Knowledge Engine

The knowledge engine obtains semantic environmental information in XML and stores it in triple store, which uses a resource description framework (RDF) such as "subject-predicate-object" or "resource-property type-value" unlike the conventional relational database that saves data in "key-value." Triple store uses the SPARQL protocol and RDF query language (SPARQL) to create, read, update, and delete the graph data that contain relations between objects. The triple store facilitates the reasoning process by using the relations and attributes between objects to find new relations. In this study, we utilize JENA-TDB, a type of triple store. It is an open source framework developed by Apache for the manipulation of RDF data. JENA-TDB provides persistence storage for the RDF and web UI with the Apache Fuseki interface using the http protocol.

The XML/OWL parser in the knowledge engine parses the XML file into OWL Ontology. Ontology is a model that explicitly describes conceptual meanings by restricting the relations in the artificial intelligence field. OWL is one of the ontology expression languages. It is designed to create an environment in which machines and agents can understand and utilize resources using reasoning

and formal syntax. OWL defines the class and property of instances, describes relations between the classes and subclasses, and infers new concepts. In this study, we classify the topology and semantic relations among objects as object property relations and the attributes of the object as data property relations when the knowledge engine receives the XML file containing the taxonomy of classes and subclasses of semantic information achieved by cognition. The classified relations are described in OWL in the XML/OWL parser. The generated OWL ontology is saved in JENA-TDB using the Fuseki http protocol. When JENA-TDB receives a request from the environment modeler to hand over the required information to set the initial and goal states for mission planning, SPARQL is used to gather data.

3.3. PDDL Planning Agent

We utilize the planning agent of Cashmore et al. [19] as the PDDL planning agent. ROSPlan provides planning in the robot operating system (ROS). However, because natural language information achieved from surrounding environments is hardly utilized, we modified it appropriately for our approach. Two nodes are added to ROSPlan: one is the language description node and the other is the scene graph generation node. Besides Mongo DB, JENA-TDB is added for semantic memory storage. Plan dispatcher is extended to cover additional environment information from the simulator. In the planning agent, problem PDDL generation, plan generation, action dispatch, and replanning are performed. From the environment modeler and Mongo DB, data related to initial state and mission parameters are gathered and feed into problem generation. Then, the problem PDDL is automatically generated and handed to a planner with domain PDDL. In this paper, the POPF planner is used. Once the plan is generated, the plan dispatch parses the PDDL actions to the ROS messages for the robots to complete the overall plan. During the execution, if an action fails because of changes in the environment, the planning agent reformulates the problematic PDDL by replanning.

4. Experiment

We demonstrate the proposed framework with a patrolling scenario and find the missing child using one UAV and three AGVs. The operational diagram for the proposed method is illustrated in Figure 5. It is composed of the control tower, natural language processor, simulator, and JENA-TDB. The scenario is run in the simulation to verify the proposed architecture. The details are as follows.

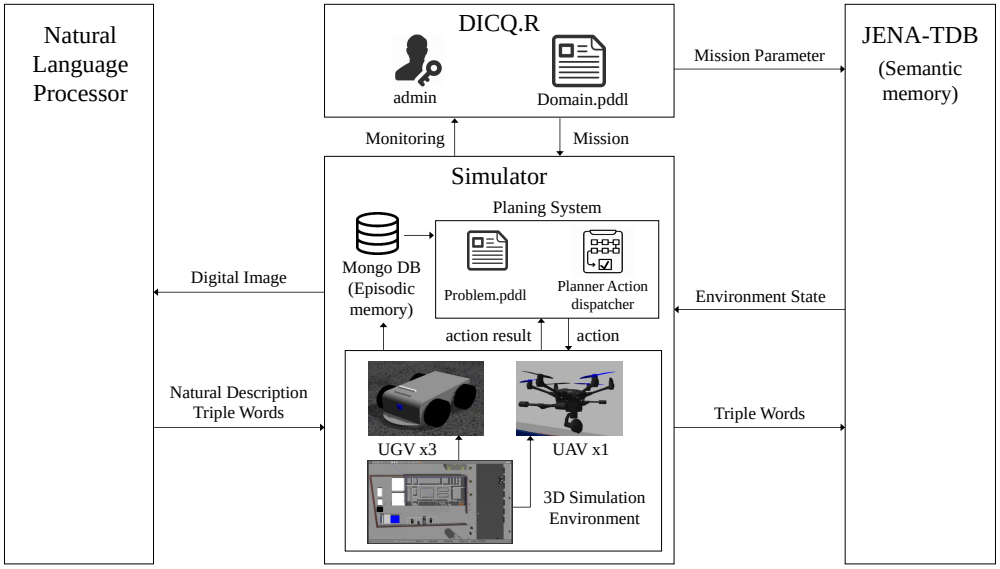


Figure 5. Operational diagram for the proposed method: It consists of a control tower, natural language processor, simulator, and JENA-TDB. Planning and execution are performed by the simulator. Natural language-based scene understanding is processed in the natural language processor. JENA-TDB is used as the semantic information processor.

4.1. Experiment Setting

The simulation environment was designed as an area around REDONE technologies cooperation, as shown in Figure 6. The size of the area was 110 m × 100 m. We utilized three AGVs of REDONE technologies, each named Smart Cookie, and 1 UAV of REDONE technologies, named Beyond. Each Smart Cookie has 2D laser sensors and an RGB-D camera. Beyond is equipped with an RGB-D camera. The laser sensor is used for navigation on the execution part while the RGB-D cameras are used for cognition for the natural language-based scene understanding. Each robot navigated using the generated map and sensor. The platform was set up with Ubuntu 16.04, ROS Kinetic, and Gazebo 7. JENA-TDB is used as the semantic memory and Mongo DB is used as the episodic memory. DICQ.R is the control tower. We used tensorflow library and Python for the natural language processing, whereas JAVA was used for JENA-TDB, and C++ was used for the simulator. Socket communication was utilized to transfer information between processors. To train the neural network for scene understanding, we used the COCO dataset and visual genome dataset for language description and scene graph generation, respectively. Since these datasets use images for natural language processing, we manually generated a graph using bounding boxes and train the networks. The details of the trained network are shown in Appendix A.

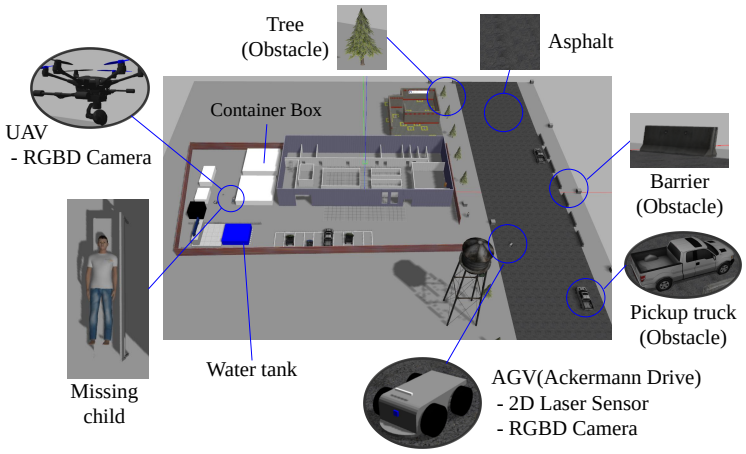


Figure 6. Simulation environment.

4.2. Scenario

The overall scenario outline is illustrated in Figure 7. Two missions were performed. One involved patrolling the area, and the other was concerned with finding a missing child. While the robots were visiting the point of interest (POI) for patrolling, a mission to find a missing child was generated by the DICQ.R. Every robot was required to report the current situation to the DICQ.R as well as if an unusual situation occurred. During the mission, we surmised what may happen if a dynamic obstacle, which a robot could not approach, were to suddenly appear at the POI. In this situation, the robot will generate natural language to report the current situation to the DICQ.R. Also, we expected at least one of the robots to find the missing child. In this case, we generated scene graphs to add POIs for the other robots to check. Analogously, the natural language-based scene understanding can be applied to other planning missions.

	Description	Activity
Name	Patrol and find a missing child	-
Preconditions	• Smart cookie1,2,3, Beyond-pf at the initial point	-
Flow of events	• Every robot perform patrol mission (A1)	1. Each robot visit assigned POI (Point Of Interest)
		2. Repeat activity 1
	• Every robot can perform actions received by DICQ.R	-
	• Every robot navigate using sensors (camera, lidar) using the generated map	-
	• Every robot reports the current situation to DICQ.R if an unnormal situation occurs	-
	• Perform 'find missing child' mission received from DICQ.R (A2)	1. DICQ.R command Smart cookie1,2,3, and Beyond-pf1 to find a missing child
		2. Smart cookie1,2,3, and Beyond-pf1 visit every POI to find the missing child
		3. Beyond-pf1 find a human
		4. Beyond-pf1 create POI at human position
		5. DICQ.R generate new mission for Smart cookie1,2,3 to go to created POI to check the found human is the missing child
		6. Mission completed
Exception Processing	• Smart cookie cannot approach POI due to a dynamic obstacle (A3)	1. Smart cookie generates natural language to describe the current situation to DICQ.R
		2. Replanning
	• Beyond-pf1 found a human expected to be the missing child (A4)	1. Save the scene graph in JENA-TDB
		2. Using semantic information of JENA-TDB, generate new mission
Post-Condition	• Every robot send execution result to DICQ.R	-

Figure 7. Overall scenario outline: Using one UAV and three AGVs, patrolling was conducted and the missing child was found.

4.3. Results

The experiment involving patrolling and finding a missing child was successful. In this study, we used 16 POIs for robot patrolling according to the assigned area. When the child went missing, assume that a human is present at POI 9. Then, the robots were asked to check all the POIs and find a human who is likely to be the missing child. When such a human is detected, a scene graph is generated and sent to the DICQ.R. Using the achieved semantic information, a POI is added and the closest robot is asked to go to POI to check if the detected human is the missing child. Tables 1 and 2 show the generated plans for the robots. In the initial plan, POI 16 is not included. After the human is detected by Beyond, a new POI (16) is generated and is checked by Smart cookie.

We used the XML file structure to send the semantic graphs to DICQ.R. The XML/OWL parser located inside the knowledge engine is provided triplet data that contain scene graph information. The OWL file is generated by the classification processes of object property and data property relations. The object property relation is relevant to the relationship between objects, and the data property relation is relevant to the properties of these objects. According to the command from the DICQ.R, which provides the mission parameters, JENA-TDB fetches semantic information using SPARQL and sends it to the environment modeler. For example, using the received triple data of "human-behind-tree," "behind" is saved as "owl:ObectProperty rdf:about plan:behind/." "human-hasPositionX-100" is saved as "owl:DataProperty rdf:about plan:hasPositionX." Also, objects are parsed as "owl:NamedIndividual," which is used to describe instances.

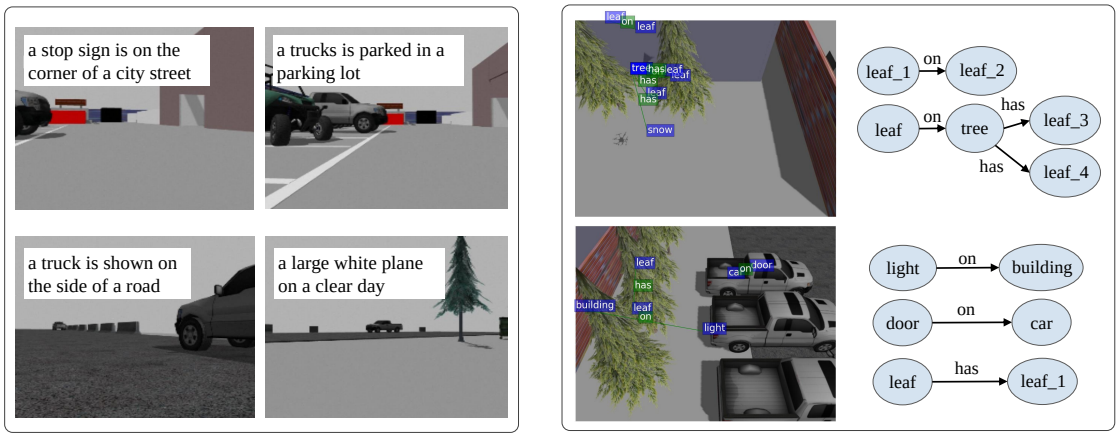
Table 1. Generated plan for Part 1 of the scenario.

0.000:	(goto_point_indoor cookie0 POI0 POI0)	[20.000]
0.000:	(goto_point_outdoor cookie1 POI12 POI12)	[20.000]
0.000:	(goto_point_street cookie2 POI2 POI2)	[20.000]
0.000:	(fly beyond0 POI6 POI6)	[20.000]
20.001:	(goto_point_indoor cookie0 POI0 POI1)	[20.000]
20.001:	(goto_point_outdoor cookie1 POI12 POI13)	[20.000]
20.001:	(goto_point_street cookie2 POI2 POI3)	[20.000]
20.001:	(fly beyond0 POI6 POI7)	[20.000]
40.001:	(goto_point_indoor cookie0 POI1 POI10)	[20.000]
40.001:	(goto_point_outdoor cookie1 POI13 POI14)	[20.000]
40.001:	(goto_point_street cookie2 POI3 POI4)	[20.000]
40.001:	(fly beyond0 POI7 POI8)	[20.000]
60.001:	(goto_point_indoor cookie0 POI10 POI11)	[20.000]
60.001:	(goto_point_outdoor cookie1 POI14 POI15)	[20.000]
60.001:	(goto_point_street cookie2 POI4 POI5)	[20.000]
60.001:	(fly beyond0 POI8 POI9)	[20.000]
80.001:	(detect beyond0 POI9 human)	[20.000]

Table 2. Generated plan for Part 2 of the scenario.

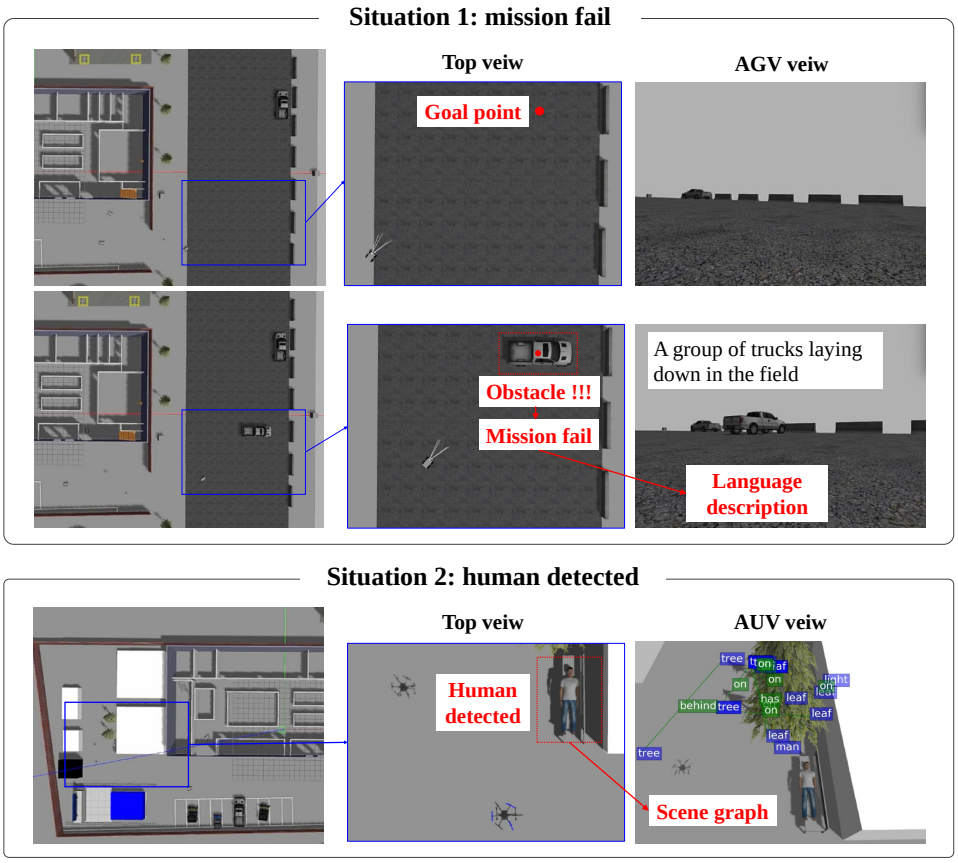
0.000:	(goto_point_outdoor cookie1 POI15 POI16)	[20.000]

The generated language descriptions and scene graphs are shown in Figure 8a,b. The language descriptions and scene graphs were successfully generated in the simulation environment. As illustrated in Figure 8c, we utilized the natural language-based scene understanding across two situations: (1) language description in the "failed mission situation" to inform the control tower about the current situation, and (2) the scene graph in the "human detected situation," to add a POI to verify whether the detected human is the missing child. As a result, we verified that the proposed framework could successfully perform the required planning using heterogeneous multiple robots based on natural language-based scene understanding.



a ***b***

Figure 8. *Cont.*



c

Figure 8. Experiment results: (**a**) Examples of generated language descriptions; (**b**) Examples of the generated scene graphs; (**c**) Results of the simulation.

5. Conclusions

We proposed a new framework for heterogeneous multi-robot cooperation based on natural language-based scene understanding. While other studies only used the raw sensor data for the purposes of perception, we focused on identifying semantic meanings from the surrounding environment to efficiently share information between heterogeneous agents. The framework combines deep learning and symbolic planning. Neural networks were used for the generation of semantic graphs and language descriptions. JENA-TDB was utilized to store semantic triple data. By gathering the data appropriate for mission parameters from JENA-TDB, the PDDL planner generated the sequence of actions for each robot. Using one UAV and three AGVs, the proposed method was successfully verified via simulation involving patrolling and finding a missing child.

Author Contributions: Conceptualization, J.M.; Methodology, J.M.; Resources, B.-H.L.; Software, J.M.; Supervision, B.-H.L.; Writing—original draft, J.M.

Funding: This work was supported in part by the Brain Korea 21 Plus Project and in part by a Bio–Mimetic Robot Research Center funded by Defense Acquisition Program Administration, and by Agency for Defense Development (UD190018ID).

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

Abbreviations

The following abbreviations are used in this manuscript:

PDDL	Planning Domain Definition Language
ROS	Robot Operating System

SLAM	Simultaneous Localization and Mapping
RNN	Recurrent Neural Networks
GCN	Graph Convolutional Network
LSTM	Long Short-Term Memory
CNN	Convolutional Neural Network
GRU	Gated Recurrent Unit
UAV	Unmanned Aerial Vehicle
UGV	Unmanned Ground Vehicle
VQA	Visual Question and Answering
RDF	Resource Description Framework
SPARQL	SPARQL Protocol and RDF Query Language
ROS	Robot Operating System
POI	Point of Interest

Appendix A. Architecture of the Trained Neural Network for Language Description

GCN is used for graph feature extraction. LSTM is utilized to generate sentences describing the graph. The maximum number of nodes is set to 80. We have added empty nodes if the detected objects are less than 80. The details of the trained neural network are shown in Table A1. The parameters are learned by Nadam. We used ReLU as the activation function.

Table A1. Overall neural network architecture for language description.

Layer Type	Filters/ Units	Output Size	Connected to	Number of Parameters
Input(Node)	-	80×4163	-	-
Input(Edge)	-	80×80	-	-
Graph convolution1	1024	80×1024	Input(Node) Input(Edge)	4,262,912
Graph convolution2	64	80×64	Graph convolution1	65,536
Fully Connected1	-	512	Graph convolution2	2,621,952
Input(Words)	-	44	-	-
Embedding	256	44×256	Input(Words)	798,976
LSTM1	256	44×256	Embedding	525,312
LSTM2	1000	1000	Fully Connected1 LSTM1	7,076,000
Fully Connected2	-	3121	LSTM2	3,124,121

References

1. Rosa, L.; Cognetti, M.; Nicastro, A.; Alvarez, P.; Oriolo, G. Multi-task cooperative control in a heterogeneous ground-air robot team. *IFAC-PapersOnLine* **2015**, *48*, 53–58. [CrossRef]
2. Wally, B.; Vyskocil, J.; Novak, P.; Huemer, C.; Sindelar, R.; Kadera, P.; Mazak, A.; Wimmer, M. Flexible Production Systems: Automated Generation of Operations Plans based on ISA-95 and PDDL. *IEEE Robot. Autom. Lett.* **2019**, *4*, 4062–4069. [CrossRef]
3. Chu, F.J.; Xu, R.; Seguin, L.; Vela, P. Toward Affordance Detection and Ranking on Novel Objects for Real-world Robotic Manipulation. *IEEE Robot. Autom. Lett.* **2019**, *4*, 4070–4077. [CrossRef]
4. Miranda, D.S.S.; de Souza, L.E.; Bastos, G.S. A ROSPlan-Based Multi-Robot Navigation System. In Proceedings of the 2018 Workshop on Robotics in Education, Joao Pessoa, Brazil, 6–10 November 2018; pp. 248–253.
5. Zhang, S.; Jiang, Y.; Sharon, G.; Stone, P. Multirobot symbolic planning under temporal uncertainty. In Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems, Sao Paulo, Brazil, 8–12 May 2017; pp. 501–510.

6. Corah, M.; O'Meadhra, C.; Goel, K.; Michael, N. Communication-Efficient Planning and Mapping for Multi-Robot Exploration in Large Environments. *IEEE Robot. Autom. Lett.* **2019**, *4*, 1715–1721. [CrossRef]
7. Bowman, S.L.; Atanasov, N.; Daniilidis, K.; Pappas, G.J. Probabilistic data association for semantic slam. In Proceedings of the International Conference on Robotics and Automation, Singapore, 29 May–3 June 2017; pp. 1722–1729.
8. Young, T.; Hazarika, D.; Poria, S.; Cambria, E. Recent trends in deep learning based natural language processing. *IEEE Comput. Intell. Mag.* **2018**, *13*, 55–75. [CrossRef]
9. Aneja, J.; Deshpande, A.; Schwing, A.G. Convolutional image captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 5561–5570.
10. Zhang, L.; Wei, L.; Shen, P.; Wei, W.; Zhu, G.; Song, J. Semantic SLAM Based on Object Detection and Improved Octomap. *IEEE Access* **2018**, *6*, 75545–75559. [CrossRef]
11. Karpathy, A.; Fei-Fei, L. Deep visual-semantic alignments for generating image descriptions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3128–3137.
12. Yao, T.; Pan, Y.; Li, Y.; Mei, T. Exploring visual relationship for image captioning. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 684–699.
13. Walter, M.R.; Hemachandra, S.; Homberg, B.; Tellex, S.; Teller, S. A framework for learning semantic maps from grounded natural language descriptions. *Int. J. Robot. Res.* **2014**, *33*, 1167–1190. [CrossRef]
14. Johnson, J.; Krishna, R.; Stark, M.; Li, L.J.; Shamma, D.; Bernstein, M.; Li, F.-F. Image retrieval using scene graphs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3668–3678.
15. Ma, L.; Lu, Z.; Li, H. Learning to answer questions from image using convolutional neural network. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016.
16. Garnelo, M.; Arulkumaran, K.; Shanahan, M. Towards deep symbolic reinforcement learning. *arXiv* **2016**, arXiv:1609.05518.
17. Asai, M.; Fukunaga, A. Classical planning in deep latent space: Bridging the subsymbolic-symbolic boundary. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
18. Mao, J.; Gan, C.; Kohli, P.; Tenenbaum, J.B.; Wu, J. The Neuro-Symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision. *arXiv* **2019**, arXiv:1904.12584.
19. 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 Twenty-Fifth International Conference on Automated Planning and Scheduling, Jerusalem, Israel, 7–11 June 2015.
20. Gautam, A.; Mohan, S. A review of research in multi-robot systems. In Proceedings of the IEEE 7th International Conference on Industrial and Information Systems, Chennai, India, 6–9 August 2012; pp. 1–5.
21. Wurm, K.M.; Dornhege, C.; Nebel, B.; Burgard, W.; Stachniss, C. Coordinating heterogeneous teams of robots using temporal symbolic planning. *Auton. Robots* **2013**, *34*, 277–294. [CrossRef]
22. Jang, I.; Shin, H.S.; Tsourdos, A.; Jeong, J.; Kim, S.; Suk, J. An integrated decision-making framework of a heterogeneous aerial robotic swarm for cooperative tasks with minimum requirements. *Auton. Robots* **2019**, *233*, 2101–2118. [CrossRef]
23. Kingry, N.; Liu, Y.C.; Martinez, M.; Simon, B.; Bang, Y.; Dai, R. Mission planning for a multi-robot team with a solar-powered charging station. In Proceedings of the International Conference on Intelligent Robots and Systems, Vancouver, BC, Canada, 24–28 September 2017; pp. 5233–5238.
24. Reis, J.C.; Lima, P.U.; Garcia, J. Efficient distributed communications for multi-robot systems. In *Robot Soccer World Cup*; Springer: Berlin, Germay, 2013; pp. 280–291.
25. Jiang, J.; Lu, Z. Learning attentional communication for multi-agent cooperation. In *Advances in Neural Information Processing Systems: Annual Conference on Neural Information Processing Systems, Montreal, QC, Canada, 2–8 December 2018*; Neural Information Processing Systems: San Diego, CA, USA, 2018; pp. 7254–7264.
26. Foerster, J.; Assael, I.A.; de Freitas, N.; Whiteson, S. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in Neural Information Processing Systems: Annual Conference on Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016*; Neural Information Processing Systems: San Diego, CA, USA, 2016; pp. 2137–2145.

27. Himri, K.; Ridao, P.; Gracias, N.; Palomer, A.; Palomeras, N.; Pi, R. Semantic SLAM for an AUV using object recognition from point clouds. *IFAC-PapersOnLine* **2018**, *51*, 360–365. [CrossRef]
28. Li, L.; Liu, Z.; Özgïner, Ü.; Lian, J.; Zhou, Y.; Zhao, Y. Dense 3D Semantic SLAM of traffic environment based on stereo vision. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium, Changshu, China, 26–30 June 2018; pp. 965–970.
29. Mao, M.; Zhang, H.; Li, S.; Zhang, B. SEMANTIC-RTAB-MAP (SRM): A semantic SLAM system with CNNs on depth images. *Math. Found. Comput.* **2019**, *2*, 29–41. [CrossRef]
30. Lu, J.; Xiong, C.; Parikh, D.; Socher, R. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 375–383.
31. Lu, J.; Yang, J.; Batra, D.; Parikh, D. Hierarchical question-image co-attention for visual question answering. In *Advances in Neural Information Processing Systems: Annual Conference on Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016*; Neural Information Processing Systems: San Diego, CA, USA, 2016; pp. 289–297.
32. Dai, B.; Zhang, Y.; Lin, D. Detecting visual relationships with deep relational networks. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3076–3086.
33. Srivastava, S.; Fang, E.; Riano, L.; Chitnis, R.; Russell, S.; Abbeel, P. Combined task and motion planning through an extensible planner-independent interface layer. In Proceedings of the IEEE International Conference on Robotics and Automation, Hong Kong, China, 31 May–7 June 2014; pp. 639–646.
34. Dornhege, C.; Hertle, A.; Nebel, B. Lazy evaluation and subsumption caching for search-based integrated task and motion planning. In Proceedings of the International Conference on Robotics and Automation Workshop on AI-Based Robotics, Karlsruhe, Germany, 6–10 May 2013.
35. Manso, L.J.; Bustos, P.; Alami, R.; Milliez, G.; Núnez, P. Planning human-robot interaction tasks using graph models. In Proceedings of the International Workshop on Recognition and Action for Scene Understanding, Valletta, Malta, 5 September 2015; pp.15–27.
36. Zhou, Y.; Tuzel, O. Voxelnet: End-to-end learning for point cloud based 3d object detection. In Proceedings of the IEEE international conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4490–4499.
37. Ghesu, F.C.; Georgescu, B.; Zheng, Y.; Grbic, S.; Maier, A.; Hornegger, J.; Comaniciu, D. Multi-scale deep reinforcement learning for real-time 3D-landmark detection in CT scans. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 176–189. [CrossRef] [PubMed]
38. Wu, Z.; Song, S.; Khosla, A.; Yu, F.; Zhang, L.; Tang, X.; Xiao, J. 3D shapenets: A deep representation for volumetric shapes. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 8–10 June 2015; 1912–1920.
39. Zhang, Q.; Sornette, D. Learning like humans with Deep Symbolic Networks. *arXiv* **2017**, arXiv:1707.03377.
40. Liao, Q.; Poggio, T. *Object-Oriented Deep Learning*; Center for Brains, Minds and Machines: Cambridge, MA, USA, 2017.
41. Moon, J.; Lee, B. Scene understanding using natural language description based on 3D semantic graph map. *Intell. Serv. Robot.* **2018**, *11*, 347–354. [CrossRef]
42. Xu, D.; Zhu, Y.; Choy, C.B.; Li, F.-F. Scene graph generation by iterative message passing. In Proceedings of the IEEE international conference on Computer Vision and Pattern Recognition, Venice, Italy, 22–29 October 2017; pp. 5410–5419.
43. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.