Towards Explanations of Plan Execution for Human-Robot Teaming

Jiyoun Moon[1] jiyounmoon@snu.ac.kr
Daniele Magazzeni[2] daniele.magazzeni@kcl.ac.uk
Michael Cashmore[3] michael.cashmore@strath.ac.uk
Dorian Buksz[2] dorian.buksz@kcl.ac.uk
Beom-Hee Lee[1] bhlee@snu.ac.kr
Yong-Seon Moon[4] moon@sunchon.ac.kr
Sang-Hyun Roh[5] rsh@urc.kr

[1]Automation and Systems Research Institute, Department of Electrical and Computer Engineering, Seoul National University,
[2]King's College London, London WC2R 2LS
[3]University of Strathclyde, Glasgow G1 1XH
[4]Department of Electronics Engineering, Sunchon National University
[5]REDONE TECHNOLOGIES CO., LTD

Abstract

Human-robot teaming is inevitable in various applications ranging from manufacturing to field robotics because of the advantages of adaptability and high flexibility. To become an effective team, knowledge regarding plan execution needs to be shared by verbalization. In this respect, semantic scene understanding in natural language is one of the most fundamental components for information sharing between humans and heterogeneous robots, as robots can perceive the surrounding environment in a form that both humans and other robots can understand. In this paper, we introduce semantic scene understanding methods for verbalization of plan execution. We generate sentences and scene graphs, which is a natural language grounded graph over the detected objects and their relationships, with the graph map generated using a robot mapping algorithm. Experiments were performed to verify the effectiveness of the proposed methods.

1 Introduction

A traditional robotic system can perform simple and repetitive tasks in well-structured environments. However, the application of robotic systems to various fields such as medicine, manufacturing, and exploration has led to an increasing demand of highly flexible robots that can work efficiently in an uncertain environment, which has resulted in a considerable amount of attention being paid to such robots [WZG19]. Combining the capabilities of humans such as adaptability, creativity, and intelligence and the abilities of robots such as rigidity, endurance, and speed can dramatically increase work efficiency [TKL+14]. Cooperation between humans and heterogeneous robots can play an important role in adapting robots to an unstructured and dynamic environment [COGM19]. Many algorithms have been developed to resolve issues such as sensing, perception to planning, control, and safety in human-robot teaming [MdSB18, ZJSS17]. Among the various elements that need to be considered for a human-robot system, the most important function is verbalization of plan execution, which includes scene

Copyright © 2019 by the paper's authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

understanding based on natural language as illustrated in Figure 1. This can enable humans and robots to share information in a form they can both understand, which is the most basic ability required for cooperation.

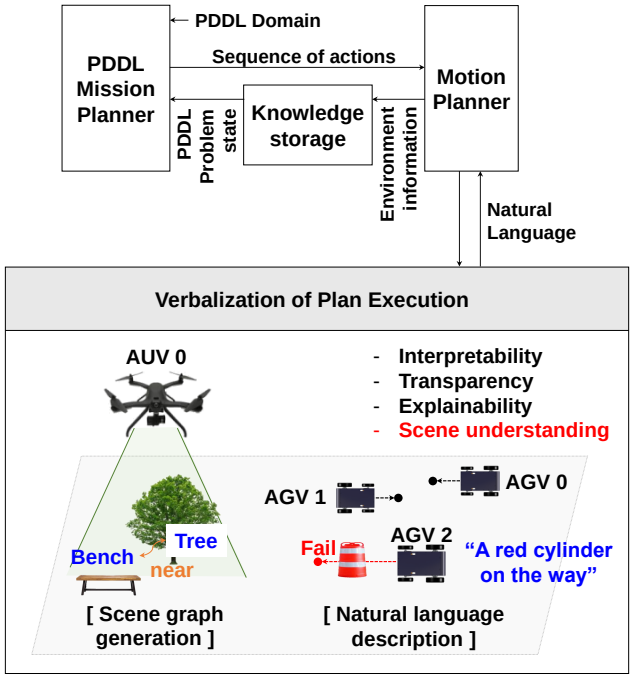

Figure 1: *Verbalization of plan execution, which is composed of interpretability, transparency, explainability and scene understanding plays an important role in human-robot teaming. In this paper, we focus on generating scene graphs and language descriptions for scene understanding.*

Semantic scene understanding is the process of perceiving environmental information in natural language or a form that can infer semantic meanings. In robotics, semantic mapping algorithms, which generate graphs that denote features and positions of detected objects as nodes, have been widely studied recently [ZWS+18, BADP17]. The graphs generated by these algorithms are unlike the maps generated by conventional methods, which consist of points, corners, lines, and planes. However, the generated semantic graph is rarely applied to data sharing methods for humans and robots, and these graphs need to be expressed in natural language. Natural-language-based scene understanding is studied in various forms such as image captioning [XBK+15, KFF15], visual question answering [GGH+17], and scene graph generation [WSW+17] in the field of computer vision. However, these methods are rarely applied to the semantic graph maps that are used by robots to represent the environment. Moreover, they do not address the problem of mission planning where humans and heterogeneous robots cooperate to achieve a common goal. In this paper, we generate scene graphs and language descriptions to focus on scene understanding, which is one fundamental element of verbalization of plan execution. A graph-based convolutional neural network [DBV16] is employed to generate sentences attention over graphs. An iterative message passing [XZCFF17] technique based on the gated recurrent unit (GRU) is used to generate scene graphs. We verified the proposed algorithms through experiments.

2 Approach

This section describes two methods of scene understanding using semantic graphs for plan execution verbalization. First, we generate natural language grounded scene graphs composed of objects as nodes, and their relationships as edges. Then, language descriptions that describe the overall scene are generated. The details are as follows.

2.1 Semantic graph generation

Semantic scene understanding based on graph maps is widely studied in robotics. However, these graph maps are rarely used for robotic applications such as mission planning, natural language processes, or plan execution ver-

balization. We address the issue of natural language-based surrounding scene understanding for the verbalization of plan execution using semantic graph maps. In this study, we assume that the graph map of the surrounding environment is generated in advance using semantic simultaneous localization and mapping (SLAM). To construct a similar graph with semantic SLAM, the features and position of objects are set as nodes. Features of objects are used for data association in the SLAM front-end. The position information of objects is utilized for graph optimization in the SLAM back-end. The generated semantic graph map G is illustrated in Fig 2. In this paper, multiple objects in the image are detected using a region proposal network [RHGS15] and encoded into feature vectors using the neural network, VGGNet [SZ14].The vector concatenated with the image information related to the $i-th$ object and the bounding box of the object is set to the feature vector f_i^v of node $v_i \in V$. We set the feature vector representing the union region of two objects as f_{ij}^e of edge $e_{ij} = (v_i, v_j) \in E_{ij}$ that connects v_i and v_j.

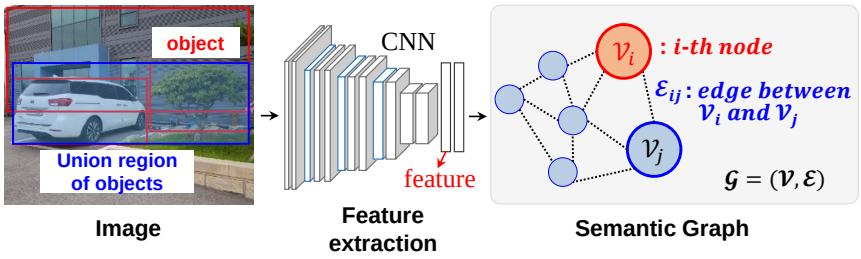

Figure 2: *Scene graph generation: Detected objects are encoded as graph node features. The union regions of two objects are encoded as graph edge features. A convolutional neural network is utilized for feature encoding.*

2.1.1 Graph inference

We infer the optimal word for each node and edge of the generated semantic graph. The graph inference process for the semantic graph f_i^v and f_{ij}^e is as follows.

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

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

where, C and R are a set of object classes and relationship types, $v_i^{class} \in C, v_i^{bbox} \in \mathbb{R}^4, e_{ij} \in R$. An iterative message passing model [XZCFF17] is utilized for graph inference. Node message pooling focuses on finding words for nodes through both the inbound and outbound edge states. Edge message pooling focuses on finding words for edges through both the object and subject states. Through repeated message pooling, we generate scene graphs comprising the most optimal words for each node and edge.

2.1.2 Language description

We generate language description for a semantic graph. The conventional methods utilizing convolutional neural network are rarely applied to graph data that is irregular and unstructured. We generate sentences using a graph convolutional neural network defined by spectral theory as follows.

$$H^{(l+1)} = \sigma(\hat{D}^{-\frac{1}{2}} \hat{A} \hat{D}^{-\frac{1}{2}} H^{(l)} W_g^{(l)}) \tag{3}$$

where, $\hat{A} = A + I$ is an adjacency matrix A with self-connection I. $\hat{D}_{ii} = \sum_j \hat{A}_{ij}$ and $W_g^{(l)}$ are a degree matrix and a trainable variable, respectively. $H^{(l)} \in \mathbb{R}^{M \times D}$ is output of the l-th layer, where $H^{(0)} = X$. In this paper, a graph is encoded as a 1024-dimensional vector with a fully connected layer. Then, a concatenated vector composed of a graph feature and a word is fed into a recurrent neural network to predict the probabilistic distribution of words.

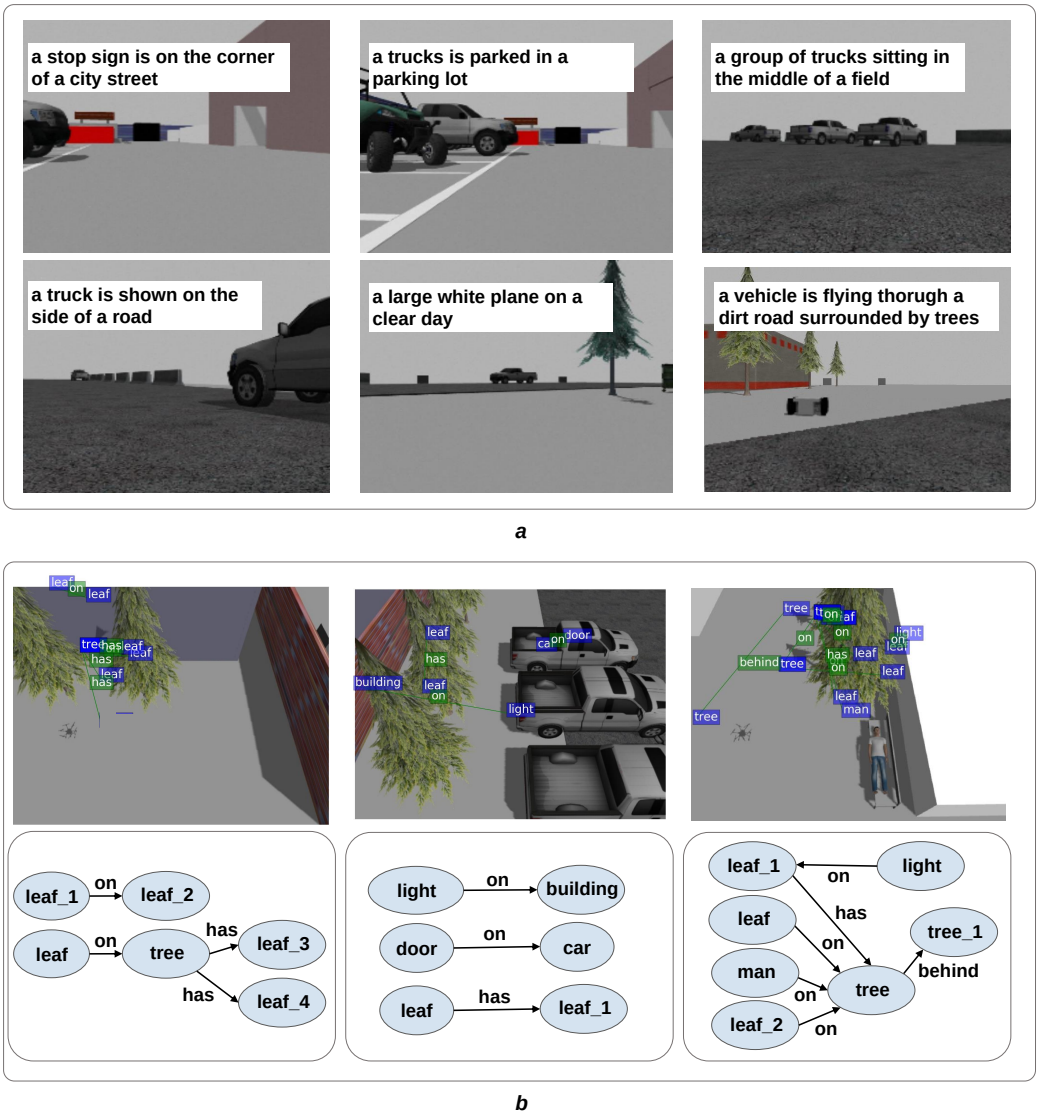

Figure 3: *Simulation Results: (a) Language description (b) Scene graph generation*

3 Experiment

We generated scene graphs and language descriptions using images of the surrounding environment obtained with mobile robots performing mission planning of surveillance in the simulation environment. Ubuntu 16.04, ROS Kinetic, and Gazebo 7 were used to set up the simulator. Two datasets were utilized for the neural network training. The network for language description was trained with a COCO dataset [LMB+14], whereas the network for scene graph generation was trained with a visual genome dataset [KZG+17]. The COCO dataset consists of images, object boundary boxes, and captions. The visual genome dataset is composed of annotations of object relationships and object labels. As these datasets only have images, we constructed graphs before training the networks; VGGNet [SZ14] was used for graph construction. We set the maximum number of nodes at 20 in order to cope with various sizes of graphs; when fewer than 20 nodes were present, empty nodes with zeros were added.

Even though the networks were trained with datasets from the real world, the proposed methods successfully generated language description and scene graphs for the simulation world as illustrated in Fig. 3. These results can be utilized for the verbalization of plan execution. Language description can contribute toward recovering from mission failure, as the failure of a robot is inevitable. For example, assume that a robot has to go to a

certain position and wait for a human to load a package. As it does so, a car blocks the path to the robot and the human cannot approach it; consequently, the robot will fail its mission. In this case, the robot can inform humans about the failure by describing the current situation and move to a new position to complete the mission. Scene graph generation can contribute toward gathering information in unseen and dynamic environments in a compact and communicable form. For example, assume that a robot is located in a place where a human cannot approach it. The generated scene graph can be used for humans to identify the place where the robot is located.

4 Artificial Intelligence Planning

AI Planning is a branch of AI that aims to provide automation by generating a structure of actions that one or multiple agents use to transition from an initial state to a desired goal state in a given environment. This is achieved by creating a model of the environment. The model aims to accurately represent the capabilities of the agent and the objects present in the environment, their attributes, as well as the relationship between them. In particular, the model includes an initial state, possible actions that affect the state, as well as the desired goal condition.

A planner is used to find one or more plans. A plan is a partially-ordered set of actions which, once executed are predicted by the model to achieve the goal condition. Typically planners perform search through the state-space in order to find one or more action sequences that provide a transition from the initial state into a state in which the goal condition holds. These forward-search planners (e.g. [CCFL10]) are equipped with various heuristics in order to find solutions faster than having to explore every state in the state space, thus enabling their use for planning and replanning online.

5 Planning and Plan Execution

Task planning for robots means planning with incomplete and unreliable data. Observations can be made from sensors in order to update the model used for planning and execution through state estimation. An up-to-date model for planning reduces the risk of plan failure, and can identify earlier when a plan under execution is no longer valid. However, even so it is likely that plans fail during execution, and in such cases it is critical that the robotic agent is able to explain to the operator exactly why.

The work presented in this paper can be usefully integrated with task planning in two main ways. First the generated scene graph can be used to update the model with new objects and relationships. Relations in the scene graph can be used to update the (spatial) predicates that describe the current state in the planner's model. Second, verbalization of the scene graph enhance descriptions of the state that can be used to describe why the plan has failed. If a location has become unreachable because of an obstruction, a verbalization of the scene graph, such as the examples in Fig. 3, can be given to a operator as an explanation of plan failure. This allows the operator to understand how the environment is different from what was expected, and what to do next. In this section we discuss future work in this direction.

A team of robots can be controlled through task planning using the ROSPlan [CFL+15] framework for task planning in ROS. The scene graph will be integrated with ROSPlan to perform continuous updates to the current state through an integration with the ROSPlan sensor interface. This can automatically connect the scene graph generation of relations such as *light on building* into the predicates of the planning model. This integration has two main advantages: first, the spatial relations in the planner's model are kept up-to-date, which is a necessary function if the robot operates within a dynamic environment. Second, new objects that are detected can be immediately described in terms of their position and relation to other objects. This is a necessary step for the planner to understand how they can be used in a plan, or what effect they might have on the state.

Plan execution on board the robots will be extended to include verbalization describing the plan under execution. This will be done by integrating the verbalization component with the plan execution components of ROSPlan in the following two ways: first to provide verbalization of updates to the current state, and second to provide verbalization of obstructions that prevent the robot from achieving its goal. In human-robot teaming scenarios, it is important that the human operator is given sufficient situational awareness to judge the state of the plan. By verbalizing the updates to the planner's model, an operator does not have to be an expert in the language of the domain model to understand what the robot is sensing. In addition, by verbalizing the reason for plan failure, the operator can quickly understand which unexpected event or object has resulting in the failure of the plan.

6 Conclusion

Verbalization of plan execution is the most fundamental component of human-robot collaboration in that it can share information in an interpretable form to achieve a shared goal. In this paper, two methods of semantic scene understanding are proposed for the verbalization of plan execution. A graph convolutional neural network and iterative message pooling are utilized to generate both language description and a scene graph, respectively. The proposed method was successfully verified with the simulator in our study.

Acknowledgement

This work was supported in part by Korea Evaluation Institute of Industrial Technology (KEIT) funded by the Ministry of Trade, Industry & Energy (MOTIE) (No. 1415162366 and No. 141562820) and in part by a Bio–Mimetic Robot Research Center funded by Defense Acquisition Program Administration, and by Agency for Defense Development (UD190018ID).

References

[BADP17] Sean L Bowman, Nikolay Atanasov, Kostas Daniilidis, and George J Pappas. Probabilistic data association for semantic slam. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 1722–1729. IEEE, 2017.

[CCFL10] Amanda Coles, Andrew Coles, Maria Fox, and Derek Long. Forward-chaining partial-order planning. In *ICAPS*, pages 42–49, 2010.

[CFL+15] M. Cashmore, M. Fox, D. Long, D. Magazzeni, B. Ridder, A. Carrera, N. Palomeras, N. Hurtós, and M. Carreras. Rosplan: Planning in the robot operating system. In *Proceedings International Conference on Automated Planning and Scheduling, ICAPS*, 2015.

[COGM19] Micah Corah, Cormac O'Meadhra, Kshitij Goel, and Nathan Michael. Communication-efficient planning and mapping for multi-robot exploration in large environments. *IEEE Robotics and Automation Letters*, 4(2):1715–1721, 2019.

[DBV16] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems*, pages 3844–3852, 2016.

[GGH+17] Zhe Gan, Chuang Gan, Xiaodong He, Yunchen Pu, Kenneth Tran, Jianfeng Gao, Lawrence Carin, and Li Deng. Semantic compositional networks for visual captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5630–5639, 2017.

[KFF15] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137, 2015.

[KZG+17] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017.

[LMB+14] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[MdSB18] Dannilo Samuel Silva Miranda, Luiz Edival de Souza, and Guilherme Sousa Bastos. A rosplan-based multi-robot navigation system. In *2018 Latin American Robotic Symposium, 2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)*, pages 248–253. IEEE, 2018.

[RHGS15] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[SZ14] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[TKL+14] Stefanie Tellex, Ross A Knepper, Adrian Li, Daniela Rus, and Nicholas Roy. Asking for help using inverse semantics. In *Robotics: Science and systems*, volume 2, 2014.

[WSW+17] Qi Wu, Chunhua Shen, Peng Wang, Anthony Dick, and Anton van den Hengel. Image captioning and visual question answering based on attributes and external knowledge. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1367–1381, 2017.

[WZG19] Nana Wang, Yi Zeng, and Jie Geng. A brief review on safety strategies of physical human-robot interaction. In *ITM Web of Conferences*, volume 25, pages 1–3. EDP Sciences, 2019.

[XBK+15] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057, 2015.

[XZCFF17] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5419, 2017.

[ZJSS17] Shiqi Zhang, Yuqian Jiang, Guni Sharon, and Peter Stone. Multirobot symbolic planning under temporal uncertainty. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, pages 501–510. International Foundation for Autonomous Agents and Multiagent Systems, 2017.

[ZWS+18] Liang Zhang, Leqi Wei, Peiyi Shen, Wei Wei, Guangming Zhu, and Juan Song. Semantic slam based on object detection and improved octomap. *IEEE Access*, 6:75545–75559, 2018.