On Semantic Support System for USAR operations with mobile robots

P. Musialik
Institute of Mathematical Machines, Warsaw, PJMusialik@gmail.com

This paper describes the design and the basic implementation of a Semantic Support System (SSS) for Urban Search and Rescue (USAR) operations. The system is intended for use by rescue teams equipped with mobile robots. The goal of the work is to provide the rescue team with a global operational picture that ensures proper information availability. The core of the system is a semantic model of the environment based on Qualitative Spatio-Temporal Representation and Reasoning (QSTRR)[1] framework. Thus, the ontology of the semantic model is described. The ontology is based on Humanitarian Data Model enhanced by the robot`s information. SSS is designed to work with multiple data sources: Geographic Information System (GIS), 3D point clouds, camera images etc. The article describes,as an example, the results of building the semantic model from 3D point clouds. An example of qualitative reasoning based on the semantic model is shown.

Key words: USAR, semantic mapping, ontology, qualitative reasoning

Introduction and Related Work

In this paper a Semantic Support System design is described. The system is dedicated for use in Urban Search and Rescue (USAR) operations with mobile robots(ground and aerial). The main goal is to fully use the information that the robot provides during mission execution. This is achieved by creating a semantic model of the environment. Such model has many potential uses in USAR activities: mission planning – additional information about the environment allows for better coordination and decision making; mission execution – help with analyzing mission data allows the rescue personnel to react to quickly changing situation of critical area, mission documentation – automatic processing of robot data easies the burden of documenting mission progress. Another important advantage is possibility of creating real life-based virtual training scenarios and environments based on created model. It is worth noting that mentioned uses were discussed with the rescue personnel in course of ICARUS project[1].

In the literature a few examples of support system for similar activities may be found. In [2] a support system SARplan for Canadian search and rescue general operations was described. In [3] a decision support system for Maritime SAR was shown. SARFOS – a support system for Mountain SAR was shown in [4]. Finally a dedicated USAR support system – FRIEDAA, developed in course of I-LOV project is described in [5].

An important part of the system is creating a semantic model of the environment. Paraphrasing [6], a semantic model is representation of the environment that besides spatial information provides temporal and functional descriptions of the objects in the scene, along with their parameters. One of the representations of a semantic model is semantic map. In [6] Nüchter and Hertzberg describe a methodology for building a semantic map from 3D laser range data. Another approach to this problem is shown in [7]. In [8] a probabilistic model of the environment, dedicated to USAR activities, is proposed. Sensor fusion is proposed as a mean for increasing accuracy. Finally in [9] a functional map for USAR mission is proposed.

In section II the ontology of the semantic environment model is described. In section III parts of the SSS are shown. Experiments are shown in section IV. The paper finalizes with conclusions and description of future work.

Environment Model and Ontology

The core of SSS is a dedicated ontology which allows for transforming raw environment data into a semantic model. USAR environment is specific in it's nature and requires specialized dictionary to be fully described. Additionally the tools that are intended to be used by rescue personnel should be easily understandable, without need of special knowledge. The requirements may be summarized:

- accurate representation of the environment,

[1] The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n°285417. (Ang.: Integrated Components for Assisted Rescue and Unmanned Search operations).

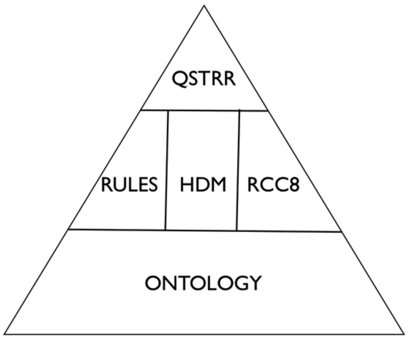

Figure 1. The scheme of the ontology

- compatibility with existing USAR data models,
- understandability for people not trained in software development and artificial intelligence techniques.

The ontology is based on the QSTRR framework, a well defined model for representing and reasoning about spatial and temporal data. Keeping the created semantic model compatible with the framework provides means for easy realization of qualitative reasoning and extension of the knowledge base. QSTRR also provides a good basis for objects physical properties(location, mass etc.)

The general scheme of the ontology is shown in figure. 1. Ontology itself is composed of three groups: concepts, relations and rules. Each of those groups is described in next subsections.

Concepts
A concept can be defined as a description of certain object class. In the scene there can be many single objects of the same concept. Each concept has a list of parameters of certain type: number, word, boolean(true/false) or a group of the previous. Additionally some parameters are concepts themselves ex. material. The concepts list is based on extended Humanitarian Data Model[10]. HDM is a Open-StreetMaps compatible USAR data model, that was first used during Haiti 2010 intervention. The group of concepts can be divided, considering HDM compatibility, into three groups: pure HDM concepts, extended HDM concepts, non-HDM concepts. Considering the type of described objects the groups that can be found are: real spatial concepts (building, vehicle), abstract spatial concepts (area), non spatial concepts(parameter concepts etc.).

Relations
For modeling relations Region Connection Calculus 8 (RCC8) [11] was chosen. Apart from being a well established relation description model, it is used in QSTRR. This allows for maintaining compatibility. Relations defined in RCC8 are: disconnected (DC), externally connected (EC), equal (EQ), partially overlapping (PO), tangential proper part (TPP), tangential proper part inverse (TPPi), non-tangential proper part (NTPP), non-tangential proper part inverse (NTPPi).

Rules
The group of rules is responsible for keeping the internal integrity of the ontology. The rule can be defined as a simple function that for certain set of input provides unambiguous output. The scheme of a rule is as follows:

```
{
        Inputs: Concepts, constant values
        Body: Checking relations between concepts and comparing values of the concepts parameters
        Output: Concept or value of concept parameters
}
```

The input of a rule is a set of concepts. The concepts may be of real and abstract type. Additionally a rule may use a constant value describing the environment. The body of the rule is a set of queries that compare relations and parameters of input concepts. The examples of queries are:

Is concept C1 in relation R1 with C2?
Is parameter X of concept C1 of value Y? etc.

Based on the results of the queries an output is generated. The output may be a newly generated concept or changing/setting a parameter/set of parameters of certain input concept. In short a rule checks if there exist an extended phrase that is semantically similar to a shorted representation in the ICARUS ontology.

The main difference between the rules set and reasoning using the ontology is that rules are used for internal ontology changing only. They are responsible for keeping the concepts' description accurate in current internal state and to remove inconsistencies. For example rules may change the damage status of a building from light to medium but they won't give an answer where the next rescue team should be send.

An example of a rule:

```
Check Victim Probability{
    Input: Building Concept B1, constant value: disaster time
    Queries:
    Q1: If(B1.damageLevel>medium) -> Yes/No
    Q2: if(B1.is_checked) -> Yes/No
    Q3: if(B1.Type=resident or B1.Type=public_use) Yes/No
    Q4: If( disaster time = daytime) Yes/No
    Output: B1.victim_probability  <- high (4xY) medium(3xY) low(2xY) no_victims(Q2=N)
}
```

Semantic Support System

SSS is responsible for carrying out two main tasks: building a semantic representation of the environment based on the data available for robotic platforms and reasoning based on the built semantic model. The first one is done by scene

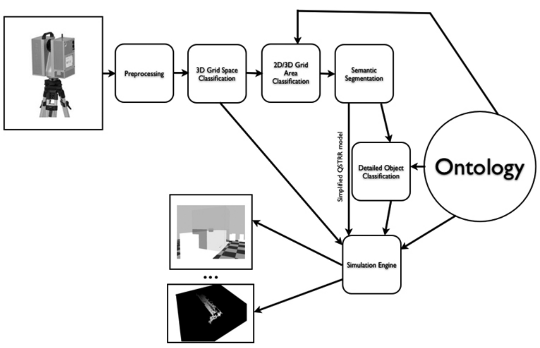

Figure 2. A scheme of SSS for 3D point clouds

segmentation and object recognition tools. For the second a specialized simulation engine is being developed.

One of the usual mobile robot's sensors is a laser range scanner. Most useful versions of such devices give as an output 3d point clouds. In this paper the main focus will be put on this data source. The analysis will be made based on normal vectors of points.

The scheme of the semantic environment building based on 3D point clouds is shown in figure 2. The main phases are: raw data preprocessing, space classification, area classification, segmentation based on area semantic label. After those operations a simple QSTRR compatible model is available for use in the simulation engine. Such representation has only basic information about the object shape (bounding box) and concept class (the classes recognized are ground, building, vegetation, undefined). To increase the fidelity of the model an additional step is foreseen: detailed object recognition. Each phase is described in the following subsections.

Preprocessing

The preprocessing step is responsible for preparing the data for further analysis. Two operations are performed: filtration and computing normal vectors. Filtration has two steps: before normal vectors computation and after. The first filtration is made by computing number of point's neighbors in a certain radius. If the number is lower than the threshold the point is removed from the cloud. After this point normal vectors are calculated. The method used is PCA-SVD, enhanced by NVIDIA CUDA parallel computation [12]. The cloud is divided into a regular grid to decrease the nearest neighbor search time. After calculation of normal vectors, the second filtration phase takes place. The points for which normal vector could not be calculated are removed from the cloud.

Space Classification

First operation done on the cloud after initial processing is creating space concepts for whole scene. This information is dedicated for the UAV type robots. The scene may have 4 different classes: free, dangerous, attention needed, unknown. For the detection purpose the scene is divided into cells of 10x10x10m. The cell's space is considered free if there are no point in it's volume. If the cell has more points then a certain threshold it is considered dangerous and should be avoided. Attention needed status is given to the cells that meat the following requirements: they have connection to free cells, they have less then 2000 points and the points take up a small volume of the cell. Lastly all the free cells that have been classified as free but lay further then the half of max range of the laser scanner are considered unknown as at such distance small object can have no measurement points. Figure. 3 shows some results for the 3D point cloud of 150x150m area.

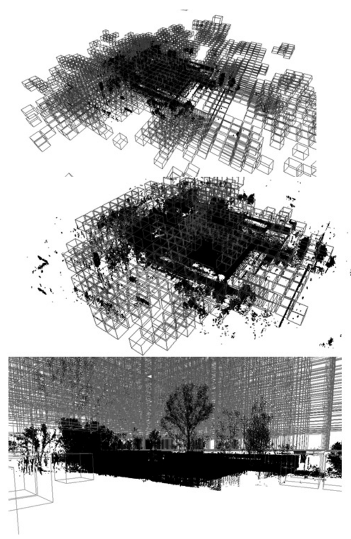

Figure 3. Area space classification: red-dangerous, orange-attention needed, green-free

Area Classification

After space classification the 3D point cloud is divided into 4 major groups: "buildings", "vegetation", "ground" and "other". The classification is rough, aiming for helping in the segmentation process and filtering out bigger objects for the scene. The classification has the following phases:

1. Finding the ground seed: a X-group histogram of vertically directed normal vectors is calculated. On the most populous group a RANSAC plain search is made. After finding the plane the 3D point cloud has all points that lay in certain A threshold from the plain and have coincident vector direction filtered out. The points are the seeds for the ground.
2. The second step is dividing the 3D point clouds int a regular grid. Two approaches have been tested: 2D grid and regular 3D grid. The 2D version divides the cloud into BxB size cells in the horizontal plane only. In 3D version a regular grid decomposition into BxBxB cells is performed. The next step is different depending on the type of the grid used. Only the cells of a population of at least 100 points are checked.

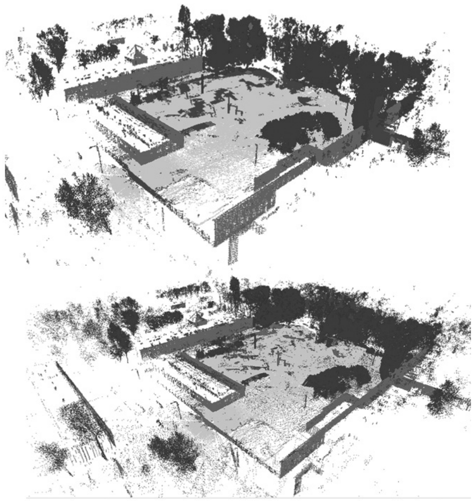

Figure 4. Area classification: up: 2D grid result; down: 3D grid result

3. The area classification is done based on normal vectors directions histogram. For each cell the number of consistent normal vectors are counted and divided into the groups: dominating (>50% of all vectors), major (C-50% of all points), other (<C% of all points). For the 2D grid the used C value was 20% for 3D 30%. The following classification is then made:
    - 2D grid: If a cell has dominating or major groups the cell is checked for possible planes. If a plane is found the cell is classified as building. Plane search is done by RANSAC search and checking the similarity between plane normal vector direction and the direction of the most populous group. If the search fails the number of vector direction groups is checked. If there are more groups then D the cell is considered as "vegetation". If all tests are negative it is considered other
    - 3D grid: The cell is considered building if it has dominating or major groups. If the first check fails it is considered as "vegetation", if the number of groups is larger than D. If all tests fail it is considered "other".
4. Independently of the type of the grid used the nest step is extending the ground class. All cells that were classified as buildings are tested for being a part of the ground. If they lay next to a ground path and normal vector direction is different from the ground path vector directions by E the cell is considered as "ground". Such test gives satisfying result because the normal vector computation algorithm does not smooth the connections between different planes.
5. The last phase of the classification is correction. The classes of cells are corrected based on the classes of neighboring cells. "Orphan" cells(that have no neighbor of the same class) are changed to the major class in the vicinity. Additionally the "other" class, as the weakest, can be changed if there exist a majority of other class types. Only building, vegetation and "other" classes are corrected.

Figure 4 shows the results of classification for 2D and 3D grid. The 2D grid has better results at assigning "other" class to small objects. The 3D approach is better for detection of local differences between larger objects. In the results shown further in the paper the 3D approach was used.

Segmentation

Last obligatory step for creating a QSTRR compatible basic entity is segmentation of the scene. The segmentation is done using the seed growth method based on two conditions: distance and class. The point is considered a part of the object, when the distance to any current object point is less than F and the point is of the same class. To increase to computation speed the 3D point cloud is divided into 3D regular cells to lower the nearest neighbor search. For each object a bounding box is found. Figure 5. shows the results of segmentation for F=0.2m. The objects extracted in this step are simple QSTRR concepts and can be used in the semantic simulation.

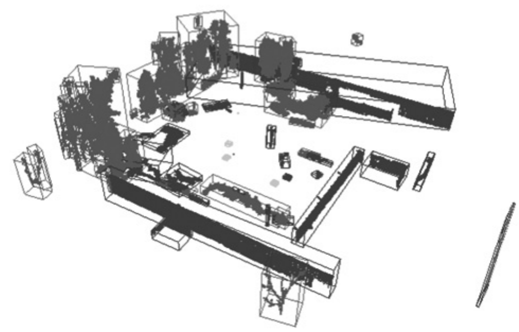

Figure 5. Semantic model after segmentation

Detail Object Classification

To increase the fidelity of the semantic model and broaden the number of detected entities a detailed object classification step is foreseen. The methods that are being tested for this task are Implicit Shape Mode[13] and Complex Shape Histogram [14]. Currently the module is under development.

Semantic Simulator

The simulator is the basic tool for reasoning in SSS. It allows for testing different model state hypothesis by simulation potential outcome. The simulator is based on NVIDIA PhysX physics engine. This choice was maid because of high fidelity of the engine and ability to easily model the RCC8 relations by using shape events. PhysX generates events if two actors' shapes are beginning contact, are in contact or have ended contact. Additionally an actor with purely virtual shape can be defined. Such functionalities are compatible with the ontology of SSS. The Experiments shows an example of using the simulator.

Experiments

A group of experiments has been carried out using the results of the described system. Three results are shown: path planning based on semantic information, traversability analysis based on robot concept parameters and space analysis for UAV operator. The environment built in the experiments was built with the following parameter values: A=0.3m, B=2m, C=30%, D=10, E=15 degrees, F=0.2m. The raw data was gathered using ZF5010 geodetic laser scanner.

Path-planning is done based on the semantic labels objects. If the object is of ground class it is passable. If the class is building or vegetation it is not passable. As the pessimistic scenario is used, the “other” class is also not passable. Based on this information a traversability map is made. On this map a path is found between two points. The method used is GPGPU enhanced diffusion method [15]. Figure 6 shows the results of the experiment.

Traversability analysis based on robot parameters is shown in figure 7. Three theoretical max slopes for the robot were simulated: 12 degrees, 18 degrees and 45 degrees. For the first value the hill is not traversable and for the last it is fully traversable. For middle value the area which can be used is shown.

The last experiment was analysis of space around a potential UGV. In figure 8 an example for the simulation engine is shown. The result is based on space classification done during semantic model creation. On the left an example of “attention needed space is shown”. There is a cable between two buildings. As the cable is small in comparison with the space, there is a possibility of UAV flying through this space, yet a special attention is needed not to hit the cable. On the right an example of space information is shown. The simulated UAV (green box) is on the verge of four space ares: 2 free, 1 AN (cable), and 1 dangerous (building). The operator is shown this information and could adapt the steering accordingly.

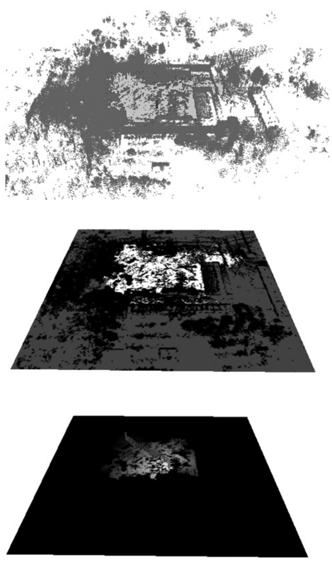

Figure 6. Path Planning: Up: traversability by label; Middle: traversability 2D map; Down: found path

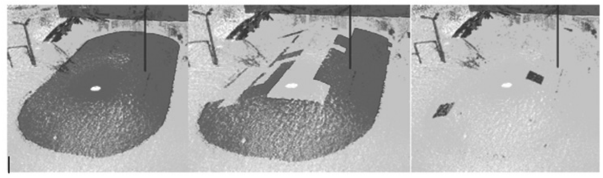

Figure 7. Traversability for different robot’s max slope parameter: Left: 12 degrees; Middle: 18 degrees; Right: 45 degrees

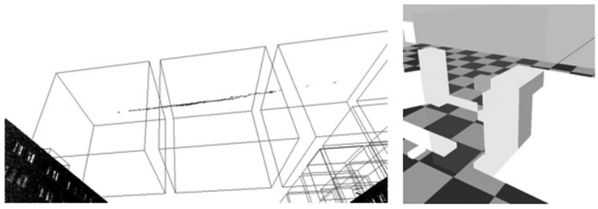

Figure 8. Space information for UGV: Left: attention needed class area with a cable; Right: a visualization of simulated UGV on boarder of 3 classes of ares: green – free, red-dangerous, orange-attention needed

Conclusion and future work

In this paper a design of semantic support system for USAR activities has been shown. The system analyses input data from robotic platform and creates a semantic environment model from them. The model can then be used to make decision about mission execution. The system is currently in the development state and not all functions are fully available, however some examples of its use are presented. Future work will be concentrated on adding additional source of input data to the system: camera images, geographical information system data (OpenStreetMaps, Google Earth etc.). Apart from that, new methods will be added to broaden the number of detected entities and to allow for automatic estimation of higher number of parameters.

Acknowledgment

The research leading to these results has received funding from the European Community’s Seventh Framework Programme (FP7/2007-2013) under grant agreement No° 285417.

References

[1] J. Będkowski, Intelligent Mobile Assistant for Spatial Design Support, Journal of Automation in Construction (2013), pp. 177-186 DOI information: 10.1016/ j.autcon.2012.09.009

[2] I. Abi-Zeid, J. R. Frost, SARPlan: A decision support system for Canadian Search and Rescue Operations, European Journal of Operational Research, Volume 162, Issue 3, 1 May 2005, Pages 630-653, ISSN 0377-2217, http://dx.doi.org/10.1016/ j.ejor.2003.10.029

[3] Y. Weihong; X. Jinkai, Intelligent Decision Support System of Maritime Search and Rescue Based on JADE, *E-Product E-Service and E-Entertainment (ICEEE), 2010 International Conference on* , vol., no., pp.1,4, 7-9 Nov. 2010

[4] B. Ott, K. Sauer, A Wägli, SARFOS - Search And Rescue Forward Operation Support System, Proceedings of ENC-GNSS 2005

[5] Q. Hamp, O. Gorgis, P. Labenda, M. Neumann, T. Predki, L. Heckes, A. Kleiner, L. M. Reindi, Study of efficiency of USAR operations with assistive technologies, Advanced Robotics, vol. 17(5), 2013

[6] A. Nüchter, J. Hertzberg ,Towards semantic maps for mobile robots, Robotics and Autonomous Systems Volume 56 Issue 11, November, 2008 , 915-926

[7] J. Będkowski, A. Masłowski, Application of semantic maps for mobile robot simulation, Elektronika 9/2011

[8] I. Nourbakhsh, K. Sycara, M. Koes, M. Yong, M. Lewis, S. Burion ,Human-Robot Teaming for Search and Rescue, IEEE Pervasive Computing: Mobile and Ubiquitous Systems, January, 2005, 72-78

[9] S. Keshavdas, H. Zender, G.J. Kruijff, M. Liu, F. Colas, FKI-LT – Functional Mapping: Spatial Inferencing to aid Human-Robot Rescue Efforts in Unstructured Disaster Environments, Proceedings of the 2012 AAAI Spring Symposium on Designing Intelligent Robots, AAAI Press, 2012

[10] http://wiki.openstreetmap.org/wiki/Humanitarian_OSM_Tags#HDM_presets

[11] Randell, D. A., Cui, Z. and Cohn, A. G.: A spatial logic based on regions and connection, Proc. 3rd Int. Conf. on Knowledge Representation and Reasoning, Morgan Kaufmann, San Mateo, pp. 165–176, 1992.

[12] Janusz Marian Bedkowski, Andrzej Maslowski, Geert De Cubber, Real time 3D localization and mapping for USAR robotic application, Industrial Robot: An International Journal, Vol. 39 Issue: 5, 2012

[13] J. Będkowski, K. Majek, A. Masłowski, P. Kaczmarek: Recognition of 3D objects for walking robot equipped with Multisense-SL sensor head" 16th International Conference on Climbing and Walking Robots, 14-17 July 2013, University of Technology, Sydney, Australia.

[14] A.Velizhev, R.Shapovalov, K.Schindler, Implicit shape models for object detection in 3D point clouds, International Society of Photogrammetry and Remote Sensing Congress, 2012

[15] B. Siemiatkowska, Coordinating the motion of mobile robots using cellular neural network, Journal of Automation, Mobile Robotics and Intelligent Systems 2 (2) (2008) 65–69.