Textual Signs Reading for Indoor Semantic Map Construction

Basem Ibraheem
Computers and Systems Dept.
Electronics Research Institute
Giza, Egypt

Abeer Hamdy
Electronics Research Institute,
and
British University in Egypt
Egypt

Nevien Darwish
Faculty of engineering
Cairo University
Giza, Egypt

ABSTRACT

Semantic maps have come up to facilitate high-level human robot interaction like "deliver a laptop for maintenance". Textual signs posted on the walls and room doors in indoor environments are environment features that could be easily identified by service robots. These signs provide some semantic information like the function of the room and its occupants. In addition, they consider landmarks for the service robots. This paper is a continuous work of developing a framework for creating a semantic map for indoor environment using mobile robot [17]. The paper addresses the problem of automatically detecting and recognizing textual signs during robotic mapping. Then, annotating these recognized signs to a previously generated robotic grid map. ATRV-mini robot has been used in our experiments.

Keywords

Mobile Robot, Map Learning, NNet, ATRV-mini, Semantic Map, OCR.

1. INTRODUCTION

A robot programmed for navigation inside a building area is categorized as an autonomous indoor robot. Autonomous indoor robot commonly has sensing facilities to avoid collision with obstacles and may also include a map for the area it is moving inside, such a map typically defines the forbidden barriers and allow the robot to understand the permitted space for performing tasks. When the robot is required to reach a target by a motion plan, it should be able to identify its current location, and its relation to the target location using its stored map. Existence of landmarks and distinct environment features will provide a great help for localization and motion planning. In addition, the human robot interface will be much user friendly if it can control the robot with environment entities that will be common between human and robots, for example, if the robot is required to reach the administration office, it will just receive the command "Go to the administration office".

Researchers have suggested many types of maps in the context of automatic environment map learning using robots. The common types are: 1) Metric Area Based maps, 2)Topological maps, 3) Hybrid maps and 4)Semantic maps.
Metric Area Based Map [4], [15], [2], [20], [22], [36],[38] is a grid based area map uses grid cells to represent the regions of the environment. Each cell (region) is distinguished by a property that represents its occupation which could be: unknown, empty, occupied. While, Topological Map [3], [16] represents the environment as a graph, with the nodes representing recognizable locations or landmarks, and the edges representing clear paths from one node to another, usually doors or corridors.
Metric maps are easy to construct but suffer from the required high computational cost. On the contrary, topological maps are difficult to build but small space is required to build a huge one. Researchers suggested Hybrid Maps [23], [1], [5] to overcome the limitation of each map type. Hybrid map can be obtained by correlating recognized landmarks on certain locations over an occupancy metric map. This type of maps leads to another challenge which is the matching techniques and algorithms. The landmarks chosen for recognition should be simple enough in order to permit an easy identification from different view angles and distances. Landmarks could be regions of the environment that can be recognized later [28] or objects of the environment with perception algorithms designed specifically for each object type. Textual signs that exist in indoor environment ,like room numbers and name of occupants, consider landmarks that can be recognized with common means of character recognition. They belong to the second type [25], [34].
Recently, Semantic maps [32], [17], [14], [26], [35] have come up to enable high-level and more intelligent robot development and improve human-robot interaction by inferring information from semantics. The term "semantic map" is used in [14], [26] to refer to a spatial representation of the robot environment which also include the type and location of objects. The presence of an object at a given location may provide semantic information: e.g., a "printer" is a space where a printer machine is located. Textual signs are objects that can represent two important aspects of the environment which are the function of each place and its relation to the high level tasks as well as the people functioning inside the facility. Thus, it should be possible to request a function from the robot such as deliver a laptop for maintenance or put some envelops in the outgoing mail box.

This paper continues the work started in [17] for creating a semantic map. It implements the modules that are responsible of detecting and recognizing the textual. Then, attaching the recognized signs to the occupancy probability map previously generated in [17] to produce a hybrid one. Although some off-the-shelf components are available for this objective but their performance is low with non-document images like those captured by the robot [8] so we have developed our optical character recognition using neural network and integrated it with an open source computer vision library to achieve our objective.

The rest of the paper is organized as follows: section 2 presents some related work to text detection. Section 3 presents the previously proposed framework for building the semantic map [17]. Section 4 introduces the development platform. Sections 5, 6 and 7 discuss the textual sign detector, reader and

correlation modules. Experiments and Results are presented in section 8. Section 9 concludes the paper.

2. RELATED WORK

There has been a great number of research deals with text detection and optical character recognition [6, 8, 25, 21, 19, 34]. Liang [21] and Jung [19] provide comprehensive survey of text detection methods.
Generally, Text detection methods could be categorized into two groups: 1)Texture-based methods and 2)Region-based methods. Texture-based methods scan the image at different scales then classify the neighbourhoods of pixels based on some text properties like variance of intensity, high density of edges, etc. In general, the methods of this category are criticized for:1)Their large computational complexity results from scanning the images at several scales, 2)Their precision is better with small text, 3)Their inability to detect slanted text precisely. The other category of text detection algorithms is based on regions. In these methods pixels exhibit certain properties such as approximately constant colour are grouped together. This approach can detect text at any scale and is not limited to horizontal texts. The approach used in this paper belongs to the second category.

3. PROPOSED SEMANTIC MAP CONSTRUCTION FRAMEWORK

The proposed framework for semantic map construction is composed of five main modules, as shown in figure 1, which are:

1. Steering and Exploration Engine

It is a Fuzzy Controller responsible for controlling the robot motion to safely explore an indoor environment, while doing the data acquisition tasks required to construct the semantic map.

2. Raw and Spatial Data Acquisition Layer

This layer consists of two main components: "Online occupancy map generator" and "Textual sign detector". *Online Occupancy Map Generator* creates an online metric map by using a probability model of sonar beams with respect to robot location information. While *Textual Sign detector is responsible* for seeking for signs around the explored areas. Whenever a sign is detected, it signals the steering and exploration engine to stop the robot until the sign image is correctly zoomed and acquired.

3. Information Detection and Extraction

This module consists of two components: High resolution offline occupancy map generator and Textual sign reader. *High-Resolution offline Occupancy Map generator* eliminates sonar noise from the online one and generates a more accurate map with higher feature details. *Textual Sign Reader Module* is activated whenever a sign is detected to recognize its textual content.

4. Semantic Knowledge Mapping

This module includes three components: Sign labeling and correlation component, indoor ontology models and reasoning engine and knowledge domain updating component.
Sign labeling and correlation is responsible for overlaying the recognized signs on the occupancy map, with respect to location and orientation information of the robot and camera head to generate a hybrid map.
Knowledge domain updating component applies object recognition and correlation techniques on images perceived by the robot for improving the knowledge existing in the constructed semantic map. While, the *indoor ontology models and reasoning engine* defines the relations between entities of the indoor environment domains such as a corridor connects between halls or forks into rooms, a door rotates to allow or prevent entering a room , an elevator connects between rooms in different levels and so on. The different functions of rooms in a facility will also be identified and linked to each other as well as equipments existing by default due to a defined function such as a Shredder machine in the mail room.

5. High-Level Task Planner Interface

It's a *Text Based Query System* that includes a human computer interface tailored to receive instructions in simple English phrases biased by realistic needs such as "send the admin laptop to maintenance" or "Fetch the manager's mail".

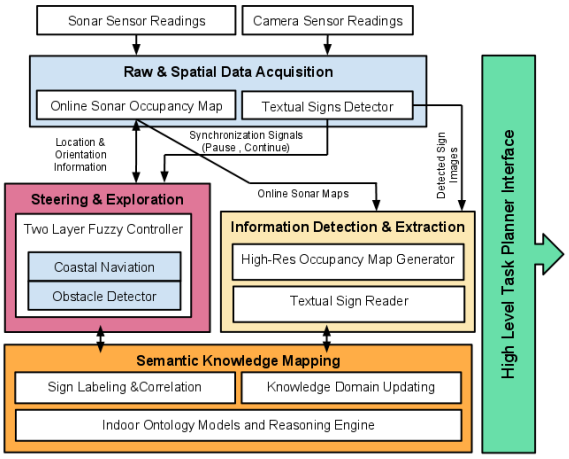


Fig. 1: High-Level Semantic Task Planner Framework

This paper discusses the implementation of three components which are: 1) Textual signs detector, 2) Textual sign reader and 3) Sign Labelling and correlation.

4. DEVELOPMENT PLATFORM

4.1. The Robot

Our research robot is "ATRV-Mini", which is a product of IRobot [18]. It is a skid steering robot, with 16 Polaroid™ sonar sensors array and a PTZ CCD camera as shown in fig 2. PTZ stands for "Pan, Tilt, Zoom" cameras, where the vision sensors are equipped with a zoom controllable lens and the whole camera is mounted on a rotating basement.

Fig. 2: ATRV-Mini robot with Sony EVID30 – PTZ Camera

4.2. The Simulation Environment

Simulation environment facilitates developing and testing the software before porting it to the real robot. "player-stage" open source simulator [30] is one of the simulators that had been investigated and found to be the best for our purpose.

"Player" is a cross-platform robot device interface and server for robot control. Player server is installed on the robot to provide the client (robot control program) access to the robot devices (sensors and actuators) over TCP network. The client can be written using any programming language that supports TCP sockets, and can run on any computer with a network communication to the robot. Player supports concurrent connections to the robot devices which allow for concurrent clients. While "Stage" is a multi-agent visual simulation environment that can be used to simulate mobile robots moving in two dimensional bitmapped environments. It is capable of simulating many sensors such as sonar, laser and cameras, also simulating actuators such as robot motors and gripper arms. The simulated robot devices customized by "stage" are replaced by the device drivers of the ATRV-mini devices to make the simulated robot in compatible with the real one. Stage presents a standard Player interface so few or no changes are required to move between simulation and hardware. Stage receives the same control code and executes it on the visual interface as if it is a robot with player server and device drivers installed.

To achieve our purpose "player" and "stage" have been interfaced with an open source C++ data flow oriented development environment called "FlowDesigner"[12] with some reusable artificial intelligence and image processing libraries. FlowDesigner features a RAD GUI with a visual debugger and can be used to build complex applications by combining small, reusable building blocks. It is used to facilitate building the required fuzzy and neural network engines and the computer vision image processing modules of the framework. FlowDesigner is written in C++ and features a plugin mechanism that allows plugins/toolboxes to be easily added. A toolbox was developed and added to the flow designer to facilitate communication with the player server or stage. Additional free open source software under GNU has been used to assist in the construction of the framework which are: 1)Image manipulating program "GIMP" [13] to manipulate maps and photos captured by robot camera. 2)Computer Aided Design for drawing maps "QCAD" [31].

5. TEXTUAL SIGN DETECTOR

The objective of this component is to search for the sign without having any prior knowledge about its location or background color. So, a hit test is performed with every homogenous color segment , the robot can see, to check out if it really contains meaningful text. Once a sign is detected, it is segmented for letters extraction. The following two subsections discuss the *real time color segmentation process* for sign detection and *the sign segmentation process* for letter extraction.

5.1. Real Time Color Segmentation

It is assumed that the sign color is homogenous and different from the walls, which is most likely to occur in real life environments. The common sign colors are red, green and yellow. If one of the color segments is found with a suitable area (number of pixels) the segment is passed as a detected sign to the sign segmentation process. An open source color segmentation software CMVision (Colour Machine Vision) [5] that performs a low level color reduction (as shown in fig 3) has been used to extract the sign color from the environment color.

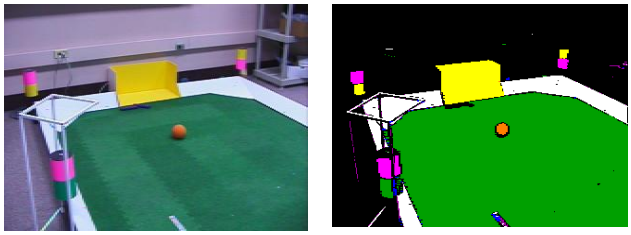


Fig. 3: The CMVision algorithm reduces the number of colors used while approximating each color to its most suitable color in the reduction set. The image on the left has a smoother shading (high number of levels), while the one on the right has average shade for each group of colors.

5.2. Sign Segmentation

If the image captured by the robot camera contains a homogeneous color segment of a suitable area; a letter extraction process is performed for later feed into the textual sign reader module. In order to maximize the throughput of the letter extraction and recognition process, a good view of the color segment (sign) is required. The more the segment is at the centre of the image (due to camera pan) and having a larger area (due to camera zoom) , the higher the probability of catching any letters written over it. A Generic PID Algorithm [39] adjusts the camera pan, tilt and zoom properties to centre the color segment and increase its area in the view. Letter extraction process proceeds as follows:

1) Averaging of 24 bit RGB into 8 bit gray scale.

$$GrayscaleByte = \frac{RedByte + GreenByte + BlueByte}{3}$$

2) Adaptive thresholding to convert the 8 bit gray scale image into 1 bit black and white image, where the threshold value is given by the following equation:

$$Threshold = \frac{\sum_i Intensity(i)}{PixelCount}$$

Where,

i: *is the pixels from* 1 *to image resolution* (320×240)

3) Segmentation by applying top-bottom, left-right pixel grouping algorithm [10].

Fig 4 shows an example on the sign segmentation for letter extraction process.

6. TEXTUAL SIGN READER

This module uses an OCR and a previously stored dictionary, contains all the possible sign texts, to read the detected signs. The following two subsections discuss both of the implementation of the OCR using NNet and the text recognition processes.

6.1. Letter Recognition using NNet

The major two types of signs in indoor environment are location signs and direction signs. ***Location Sign*** contains text or figure about the place where it is hanged up. While, ***Direction Sign*** contains information about a heading, available for motion, and a text description of the location that shall be reached, upon performing this motion,. Headings and directions are always represented by arrows. So a NNet has been trained to identify English letters and arrows printed on any surface with reasonable contrast from various viewpoints.
The training set consists of standalone arrows, characters and numbers in "Arial" font, skewed, distorted and with different view angles as shown in fig 5a. The training set was prepared by printing letters on white sheets, and capturing them with the robot camera. Each letter is converted to a binary data matrix (-1,1) of size 13 x 13 pixels (fig 5b) which are fed to the NNET. The size of the Letter pixels has been chosen by trading off the speed of neural processing time and the expected viewing angles that can be covered. Previous experiments by L'etourneau,et. [25] showed that 13 x 13 pixels provide recognizable characters on $\pm 30^o$ viewing angle. Additional three features are fed to the NNET which are: 1) the letter's Height divided by its width, 2) X-axis of the letter's center of gravity, 3) Y-axis of the letter's center of gravity (fig 5b). These features are commonly chosen among different character recognition researchers [15], [34]. Consequently, the input layer of the NNET consists of 172 neurons, and 1 hidden layer of 7 neurons and 40 output neurons for 10 numeric digits, 26 characters and 4 arrows. The network uses hyperbolic tangent transfer function, as in the case with most OCR networks [25],[34].

Training was done over 3000 epochs by providing the training patterns and the correct corresponding output to the NNet using Delta-Bar-Delta [27]. A heuristic algorithm for modifying the learning rate μ_{ij} for each weight w_{ij} (i=neuron index, j=layer index(1~3)) as training progresses is as follows:

For each weight w_{ij} do:

- The gradient at the current timestamp is compared to the average of the gradients of the previous steps. Where, the gradient of error with respect to time t is given by:

$$\bar{g}_{ij}(t) = (1-\beta)\bar{g}_{ij}(t) + \beta\bar{g}_{ij}(t-1)$$

Where,

$$0 < \beta < 1$$

- if the gradient is in the same direction
the learning rate μ_{ij} is increased, Else
the learning rate μ_{ij} is decreased.

The learning rate μ_{ij} at time t+1 is given by the following equation:

$$\mu_{ij}(t+1) = \begin{cases} \mu_{ji}(t) + \kappa \rightarrow \overline{g_{ij}}(t-1)g_{ij}(t) > 0 \\ (1-\gamma)\mu_{ji}(t) \rightarrow \overline{g_{ij}}(t-1)g_{ij}(t) < 0 \\ \mu_{ji}(t) \rightarrow otherwise \end{cases}$$

Where, β, k and γ are: Initial learning rate 0.00001, Increase 1.04, Decrease 0.6 respectively.

6.2. Text Recognition

Textual sign may contain multiple words, so after letter segmentation took place, the letters of each word should be grouped together. The criterion for grouping characters into one word is to find a larger vertical spacing than that found between letters. Due to errors in the recognized letters, words may not be recognized correctly. For a textual message to be represented by realistic word; a dictionary, includes possible sign texts, is used to measure the correctness of recognized words. Two factors define the probability of word existence in the dictionary: 1) Total existence probability of recognized letters in a dictionary word, 2)Total location matching score (letter location in the word), which is given by the following equation [25]:

$$P(X|w) = \prod_{k=1}^{N} P_{w,k}(w[k])$$

Where, X: Dictionary best hit word, w: Observed letter, N: No. of letters in the word.

The word, from the dictionary, with the highest possibility of occurrence is picked up.

After a text is recognized a sign entry is recorded in the landmarks file. Sign entry includes (Segment color, Recognized text, Camera PTZ information, Robot location and orientation).

7. SIGN LABELING AND CORRELATION

This module allocates the detected signs on the previously generated occupancy probability map [17]. It is assumed that the detected sign exists on the nearest wall sensed by the robot sonar beams in the same direction of the camera head. As, each sign entry in the landmark file includes the robot position when the sign has been detected, a simple geometry is applied to intersect the camera head axis line and the nearest wall object as shown in fig 6. The digital differential analyzer algorithm [11] is used to plot a virtual line over the occupancy map and it stops when occupancy information is maximum.

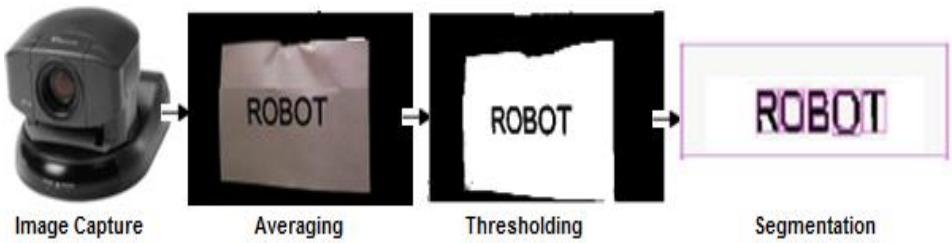


Fig. 4: Image segmentation for the photo captured by the robot.

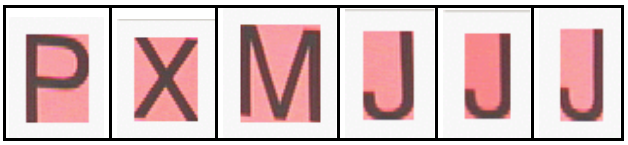


Fig. 5a: Separate English Letters and J Letter from different view angles

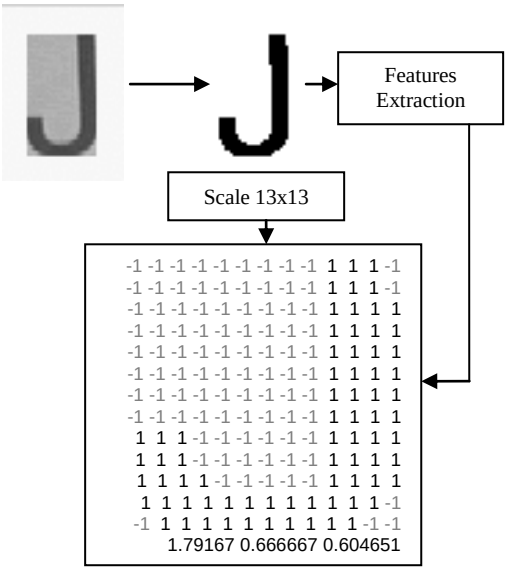


Fig. 5b: Features of letter J to be fed to the NNet

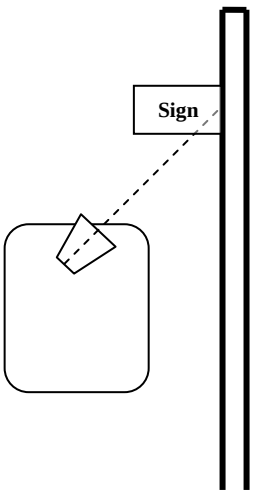


Fig. 6: Locating an identified sign on the nearest wall.

7. EXPERIMENTS AND RESULTS

The camera of the ATRV-Mini robot "Sony EVID30" is directly used in the implementation. The following subsections discuss the testing of the previously discussed components.

8.1. NNet training for letter recognition

Three papers with a set of English capital letters, numbers and arrows (36 letter each) printed at different order. They are video captured by the robot camera at 30 different angles, then segmented making a total of 3240 letters for supervised training. One of the encountered problems during conducting the experiments is the effect of lightening conditions on the segmentation (for letter extraction) process. To overcome this problem, another adaptive threshold is used to convert the 8 bit grayscale image into 1 bit black and white image which is given by the following equation:

*Threshold = (max intensity – min intensity)*c*

Where, c, is a constant factor.

8.2. NNet Evaluation

The character is considered recognized when the output neuron associated with the character is activated by at least 0.8**.** A performance test is carried out on the resulting neural network and is shown by Table 1. Letters are well recognized from different viewing angles.

Table 1. NNet Performance Test

Letter	True	Flase/Null
1	82.78	17.22
2	79.86	20.14
3	85.60	14.40
4	74.22	25.78
5	71.30	28.70
6	51.36	48.64
7	85.60	14.40
8	48.54	51.46 (B)
9	74.22	25.78
A	84.14	15.86
B	82.78	17.22
C	85.60	14.40
D	85.60	14.40
E	84.14	15.86
F	85.60	14.40
G	82.69	17.31
H	85.60	14.40
I	82.78	17.22
J	67.54	32.46(I)
K	85.60	14.40
L	85.60	14.40
M	81.06	18.94
N	85.60	14.40
O	84.49	15.51
P	76.53	23.47
Q	85.60	14.40
R	85.60	14.40
S	69.85	30.15
T	85.60	14.40
U	76.78	23.22
V	82.69	17.31
W	85.60	14.40

X	74.30	25.70
Y	79.69	20.31
Z	85.60	14.40
➔	90	10 (H)
←	91.6	8.4 (E)
↑	92.4	7.6 (T)
↓	84.3	7.7,8 (J,L)

8.3. Text Recognition test

A series of evaluations has been performed on the signs including arrows, numbers and multi line text. Figure 7 shows a test case from reality of the described textual sign recognized during exploration. Figure 8 shows a complete test case of a directional sign.

8.4. Sign Labelling and correlation

Figure 9b shows a generated grid map annotated with the recognized signs (4 signs) during robot exploration of the environment shown in figure 9a.

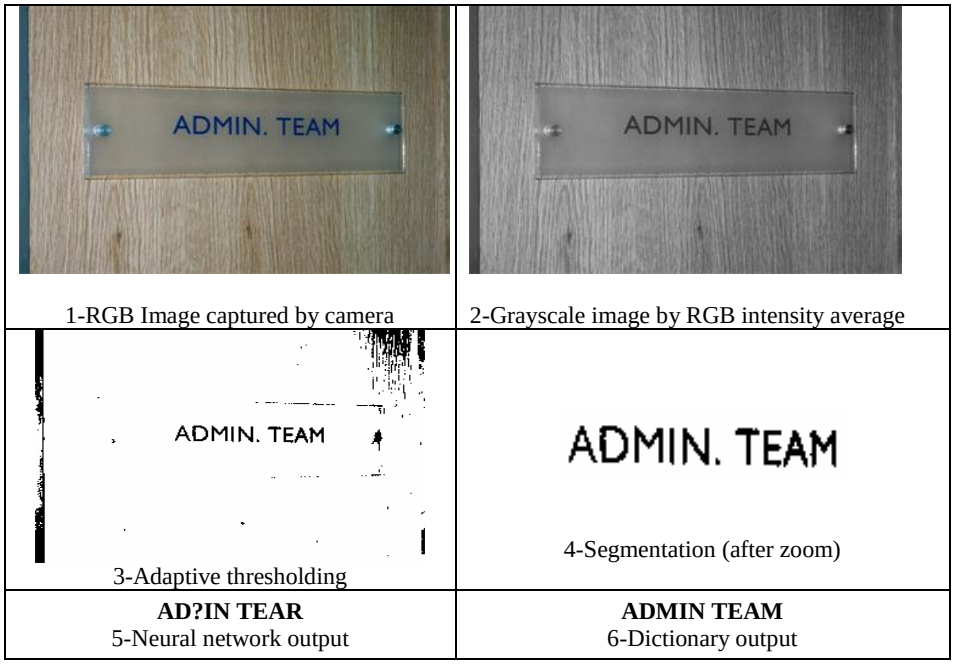


Fig. 7: TestCase1 for Textual Sign Reader Module

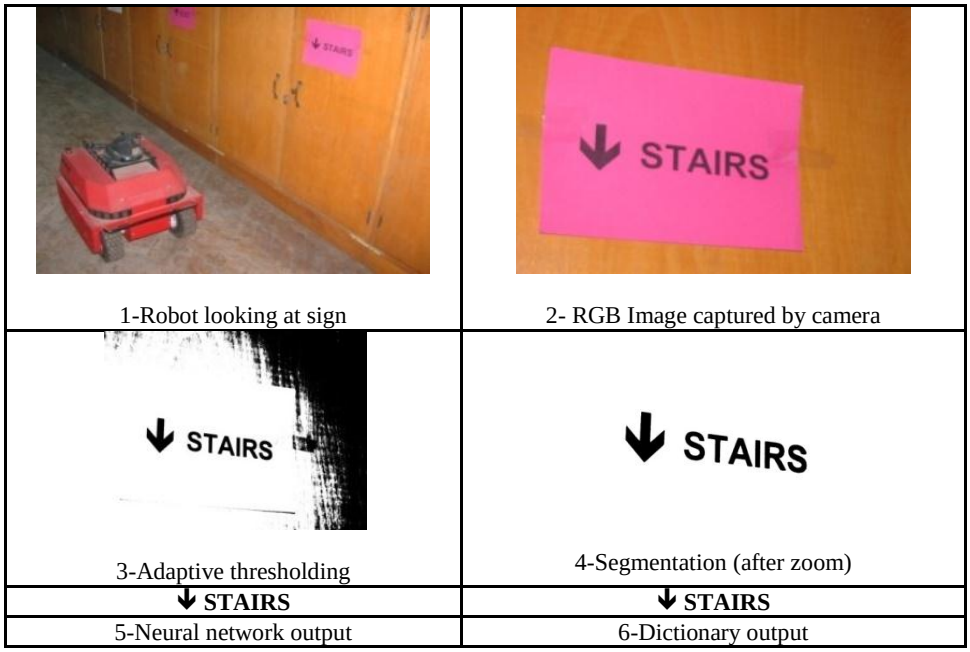
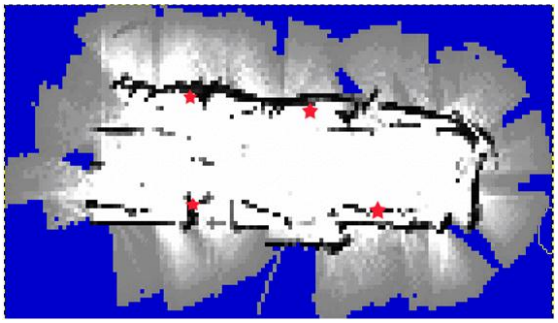


Fig. 8: TestCase2 for textual sign reader module

(a)



(b)

Fig. 9: (a) A noisy environment has 4 signs. (b) The generated hybrid map. Stars visualize sign locations from the landmarks file

8. CONCLUSION AND FUTURE WORK

Our goal is to build a semantic map for indoor environment using a mobile robot. The proposed framework to achieve this goal was presented in a previous work [17]; in addition to the implementation of the modules responsible for robot exploration and map construction. This paper discusses the implementation of the textual sign detector and reader modules. A NNet was trained for recognizing the detected signs from different view points. The recognized signs are allocated on the previously generated occupancy probability map [17] to produce a hybrid one that incorporates all extracted knowledge. Currently we are investigating how to move forward with the generated hybrid map towards a semantic model.

9. ACKNOWLEDGMENT

We acknowledge Prof. Ashraf Abedl Wahab professor of machine learning at Electronics Research Institute for his contibution to the work presented in this paper.

10. REFERENCES

[1] B. Lorenz, H. Ohlbach, and E. Stoffel, " A Hybrid Spatial Model for Representing Indoor Environments", Journal of Web and Wireless Geographical Information Systems, SpringerLink, Volume 4295, 2006.

[2] Beeson,P., MacMahon,M., Modayil,,J. Murarka,A., Kuipers, B., Stankiewicz, B. 2007, Integrating Multiple Representations of Spatial Knowledge for Mapping, Navigation, and Communication, Interaction Challenges for Intelligent Assistants, In Proceedings of AAAI Spring Symposium Series.

[3] Beevers, K. R., and Huang, W. H. 2005, Loop closing in topological maps, In Proceedings of (ICRA 2005) IEEE Int. Conference on Robotics and Automation.

[4] Birk, A. and Carpin, S. 2006. Merging occupancy grid maps from multiple robots. In Proceedings of the IEEE Special Issue on Multi-Robot Systems.

[5] Bruce, J., Balch, T., and Veloso, M. 2000, "Fast and cheap color vision on commodity hardware", In Proceedings of Workshop on Interactive Robotics and Entertainment.

[6] B. Epshtein, E. Ofek and Y. Wexler, "Detecting text in natural scenes with stroke width transform", Computer Vision and Pattern Recognition, 2010.

[7] Carlos, J., Broch, P., Teresa, M. and Monferrer, E. 2005 "Cognitive Maps for Mobile Robot Navigation: A Hybrid representation Using Reference Systems", In Proceedings of the 19th International Workshop on Qualitative Reasoning.

[8] Case, C., Suresh, B., Coates, A. and Ng, A.Y. 2011. Autonomous sign reading for semantic mapping. In Proceedings of IEEE International conference on Robotics and Automation.

[9] C. Galindo, J.A. Fernández-Madrigal, J. González, A. Saffiotti, "Robot Task Planning using Semantic Maps", Journal of Robotics and Autonomous Systems 56,2008.

[10] Dai1, F., Sugisaka2, M. and Zhang3, B., 2011, A Survey of Image Segmentation by the Classical Method and Resonance Algorithms. InTech.

[11] F. V. Mayorov, "Electronic Digital Integrating Computers Digital Differential Analyzers", London: Iliffe Books Ltd, 1964.

[12] FlowDesigner project http://www.sf.net/flowdesigner.

[13] GIMP: http://www.gimp.org/

[14] Galindo, C., Saffiotti, A., Coradeschi, S., Buschka, P., Fern´andez-Madrigal, J., Gonz´alez, J., 2005, Multi-hierarchical semantic maps for mobile robotics, In Proceedings of the IEEE/RSJ Int. Conf on Intelligent Robots and Systems.

[15] Grisetti, G., Stachniss, C., Burgard, W., 2005, Improving Grid Based SLAM with Rao Blackwellized Particle Filters by Adaptive Proposals and Selective Resembling, In Proceedings of Int. Conference on Robotics and Automation.

[16] Huang, W. H. and Beevers, K. R. , 2005, Topological mapping with sensing-limited robots, In M. Erdmann et al., editors, Algorithmic Foundations of Robotics VI, Zeist, Springer.

[17] Ibrahim, B., Hamdy, A., 2011, Building Semantic Maps for High-Level Task Planning of Mobile Robots, In Proceedings of 1st International Conference on New Paradigms in Electronics and Information Technologies.

[18] IRobot, http://www.irobot.com.

[19] Jung, K., Kim, K., Jain, A. K., 2004, Text information extraction in images and video: a survey, Pattern Recognition.

[20] Jung-Suk, L., Yep, N. S., Kyun, C. W., 2011, Robust RBPF-SLAM for Indoor Mobile Robots Using Sonar Sensors in Non-Static Environments, Advanced Robotics.

[21] J. Liang, D. Doermann, H. Li, "Camera-based analysis of text and documents: a survey", International Journal on Document Analysis and Recognition", 2005.

[22] Kleeman, L., 2003, Advanced sonar and odometry error modeling for simultaneous localization and map building, In Proceedings of Intelligent Robots and Systems.

[23] Kuipers, B., Modayil, J., Beeson, P., MacMahon, M. and Savelli, F., 2004, Local Metrical and Global Topological Maps in the Hybrid Spatial Semantic Hierarchy, In Proceedings of Int. Conference on Robotics and Automation.

[24] LaValle, S.M. 2006 Planning Algorithms, Cambridge University Press http://planning.cs.uiuc.edu/.

[25] L´etourneau, D., Michaud, F., Valin, J., Proulx, C., 2003, Textual Message Read by a Mobile Robot, In Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems.

[26] Meger, D., Forssen, P., Lai, K., Helmer, S., McCann, S., Southey, T., Baumann, M., Little, J., Lowe, D., Dow, B., 2007, Curious George: An attentive semantic robot, In Proceedings of IROS Workshop: From sensors to human spatial concepts.

[27] Minai, A., Williams, R. 1990. Back-propagation heuristics: a study of the extended delta-bar-delta algorithm. In Proceedings of IJCNN International Joint Conference.

[28] Morgenstern, C. and Heisele,B. 2003 "Component Based Recognition of Objects in an Office Environment", http://cbcl.mit.edu/publications/ai-publications/2003/AIM-2003-024.pdf.

[29] O'Sullivan, S., Collins, J.J., Mansfield, M., Haskett, D. and Eaton, M., 2004, Linear Feature Prediction for Confidence Estimation of Sonar Readings in Map Building, In Proceedings of 9th International Symposium on Artificial Life and Robotics.

[30] Player Stage project, http://www.sf.net/playerstage.

[31] Qcad: www.qcad.org

[32] Ranganathan, A., Dellaert ,F. 2007 Semantic modeling of places using objects. In Proceedings of Robotics Science and Systems Conf.

[33] Rogers III, J. G., Trevor, A J. B., Nieto-Granda, C. and Christensen, H.I., 2011, Simultaneous Localization and Mapping with Learned Object Recognition and Semantic Data Association, In Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems.

[34] Steck, M. and Müller, R. ,2003, Road Sign Recognition, Term Paper, Computer Perception with Artificial Intelligence, University of Applied Sciences, Biel, Switzerland.

[35] S. Vasudevan, S. Gachter, V. Nguyen, R. Siegwart, "Cognitive maps for mobile robots and object based approach" , Robotics and Autonomous Systems 55, 2007.

[36] Thrun, S., Liu, Y., Koller, D. ,2004, Simultaneous localization and mapping with sparse extended information, International Journal of Robotics Research.

[37] Trevor, A. J. B, Rogers III, J. G. , Nieto-Granda, C. and Christensen, H.I., 2010, Applying domain knowledge to SLAM using virtual measurements, In Proceedings of International Conference on Robotics and Automation.

[38] X. Li, X. Huang and M. Wang, "Robot Map Building From Sonar Sensors And Dsmt", Information & Security. An International Journal, Vol. 20, 2006.

[39] Y. Li, K.H. Ang, and G.C.Y. Chong, "PID control system analysis and design - Problems, remedies, and future directions", IEEE Control Systems Magazine, 26 (1), 2006.