TITLE: **Spatial Language for Human-Robot Dialogs**

AUTHORS: **Marjorie Skubic[1] (Corresponding Author)**

Dennis Perzanowski[2]

Sam Blisard[1]

Alan Schultz[2]

William Adams[2]

Magda Bugajska[2]

Derek Brock[2]

[1] Computer Engineering and Computer Science Department

201 Engineering Building West

University of Missouri-Columbia, Columbia, MO 65211

skubicm@missouri.edu / snbfg8@mizzou.edu

Phone: 573-882-7766

Fax: 573-882-8318

[2] Navy Center for Applied Research in Artificial Intelligence

Naval Research Laboratory, Washington, DC 20375-5337

<dennisp | schultz | adams | magda>@aic.nrl.navy.mil / brock@itd.nrl.navy.mil

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2003		**-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Spatial Language for Human-Robot Dialogs	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Navy Center for Applied Research in Artificial Intelligence,Naval Research Laboratory,Code 5510,Washington,DC,20375**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT

In conversation, people often use spatial relationships to describe their environment, e.g., "There is a desk in front of me and a doorway behind it", and to issue directives, e.g., "Go around the desk and through the doorway." In our research, we have been investigating the use of spatial relationships to establish a natural communication mechanism between people and robots, in particular, for novice users. In this paper, the work on robot spatial relationships is combined with a multi-modal robot interface. We show how linguistic spatial descriptions and other spatial information can be extracted from an evidence grid map and how this information can be used in a natural, human-robot dialog. Examples using spatial language are included for both robot-to-human feedback and also human-to-robot commands. We also discuss some linguistic consequences in the semantic representations of spatial and locative information based on this work.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**		**39**	

Spatial Language for Human-Robot Dialogs

Marjorie Skubic[1], Dennis Perzanowski[2], Sam Blisard[1], Alan Schultz[2], and William Adams[2]

[1] Computer Engineering and Computer Science Department
University of Missouri-Columbia, Columbia, MO 65211
skubicm@missouri.edu / snbfg8@mizzou.edu

[2] Navy Center for Applied Research in Artificial Intelligence
Naval Research Laboratory, Washington, DC 20375-5337
<dennisp | schultz | adams>@aic.nrl.navy.mil

Abstract

In conversation, people often use spatial relationships to describe their environment, e.g., "There is a desk in front of me and a doorway behind it", and to issue directives, e.g., "Go around the desk and through the doorway." In our research, we have been investigating the use of spatial relationships to establish a natural communication mechanism between people and robots, in particular, for novice users. In this paper, the work on robot spatial relationships is combined with a multi-modal robot interface. We show how linguistic spatial descriptions and other spatial information can be extracted from an evidence grid map and how this information can be used in a natural, human-robot dialog. Examples using spatial language are included for both robot-to-human feedback and also human-to-robot commands. We also discuss some linguistic consequences in the semantic representations of spatial and locative information based on this work.

Index terms – human-robot interaction, multimodal interface, spatial relations, histogram of forces, evidence grid map, semantic representations, locatives

I. Introduction

In conversation, people often use spatial relationships to describe their environment, e.g., "There is a desk in front of me and a doorway behind it", and to issue directives, e.g., "Go around the desk and through the doorway". Cognitive models suggest that people use these types of relative spatial concepts to perform day-to-day navigation tasks and other spatial reasoning [1,2], which may explain the importance of spatial language and how it developed. In our research, we have been investigating the use of spatial relationships to establish a natural communication mechanism between people and robots, in particular, striving for an intuitive interface that will be easy for novice users to understand.

In previous work, Skubic *et al.* developed two modes of human-robot communication that utilized spatial relationships. First, using sonar sensors on a mobile robot, a model of the environment was built, and a spatial description of that environment was generated, providing linguistic communication from the robot to the human [3]. Second, a hand-drawn map was sketched on a Personal Digital Assistant (PDA), as a means of communicating a navigation task to a robot [4]. The sketch, which represented an approximate map, was analyzed using spatial reasoning, and the navigation task was extracted as a sequence of spatial navigation states. In [5], the results of these two modes were compared for similar, but not exact environments, and found to agree.

In this paper, robot spatial reasoning is combined with a multi-modal robot interface developed at the Naval Research Laboratory (NRL) [6,7]. Spatial information is extracted from an evidence grid map, in which information from multiple sensors is accumulated over time [8]. Probabilities of occupancy are computed for grid cells and used to generate a short-term map. This map is then filtered, processed, and segmented into environment objects. Using linguistic spatial terms, a high-level spatial description is generated which describes the overall environment, and a detailed description is also generated for each object. In addition, a class of persistent objects has been created, in which objects are given locations in the map and are assigned labels provided by a user.

The robot spatial reasoning and the NRL Natural Language Processing system are combined to provide the capability of natural human-robot dialogs using spatial language. For example, a user may ask the robot, named Coyote hereafter, "Coyote, how many objects do you see?" Coyote responds, "I am sensing 5 objects." The user continues, "Where are they?" The robot responds, "There are objects behind me and on my left." We consider both detailed and coarse linguistic spatial descriptions, and we also support queries based on spatial language, such as "Where is the nearest object on your right?" In addition, spatial language can be used in robot commands, such as "Coyote, go to the nearest object on your right". Finally, we consider unoccupied space that is referenced using spatial terms, to support commands such as "Coyote, go to the right of the object in front of you".

The paper is organized as follows. Section II provides the background and context of the human-robot interaction and provides an overview of the multi-modal interface used. In Section III, we discuss algorithms used to process the grid map and generate multi-level linguistic spatial descriptions. Section IV discusses how the spatial language is used in an interactive dialog, and Section V provides a discussion of the semantics of spatial language from a linguistics perspective. In Section XY we present some preliminary findings from a pilot study involving a small number of human subjects using our multimodal interface to interact with a mobile robot. We discuss the use of spatial language and the various modalities used in order to get a mobile robot to find an object hidden in a room. We conclude in Section VI.

II. Related Work

From the beginnings of multi-modal interfacing with Bolt's "Put That There" system [10], multi-modal systems have evolved tremendously. Interfaces not only incorporate traditional so-called WIMP technology (windows, icons, menus, pointers) but they now incorporate natural language and gesturing [11,12,6] and dialog management [13,14], to name but a few of the numerous ways humans can now interact with machines.

Previous gestural interfaces, such as [12], have relied on stylized gestures; i.e., certain arm and hand configurations are mapped to gestural information. Other so-called gestural interfaces, such as [11], have concentrated only on one mode of interaction, such as interacting with the displays on a PDA. Because of our limited vision capabilities[1], our natural gestures are limited to two (See Section III)**.** Since we also incorporate a speech component, we are interested in including linguistic information in the way of a dialog. We believe that natural language combined with the other various interactive modes facilitates human-robot interaction and communication. Other interactive systems, such as [13,14], exist and process information about the dialog. However, we are interested in combining dialog management with our natural language understanding and gesture capabilities to facilitate human-robot communication and interaction. In this sense, our multi-modal interface is unique. We turn now to some of the kinds of interactions supported by our multimodal interface.

III. System Overview

One of our main research goals concerns interfacing various modalities in an effort to promote natural interactions. We are also interested in providing the user with a rich selection of interactive modalities so that the user can concentrate on the task at hand, rather than on how to interact. To accomplish this, we have designed and implemented the multimodal interface in Fig.1 to interact with one or more robots.

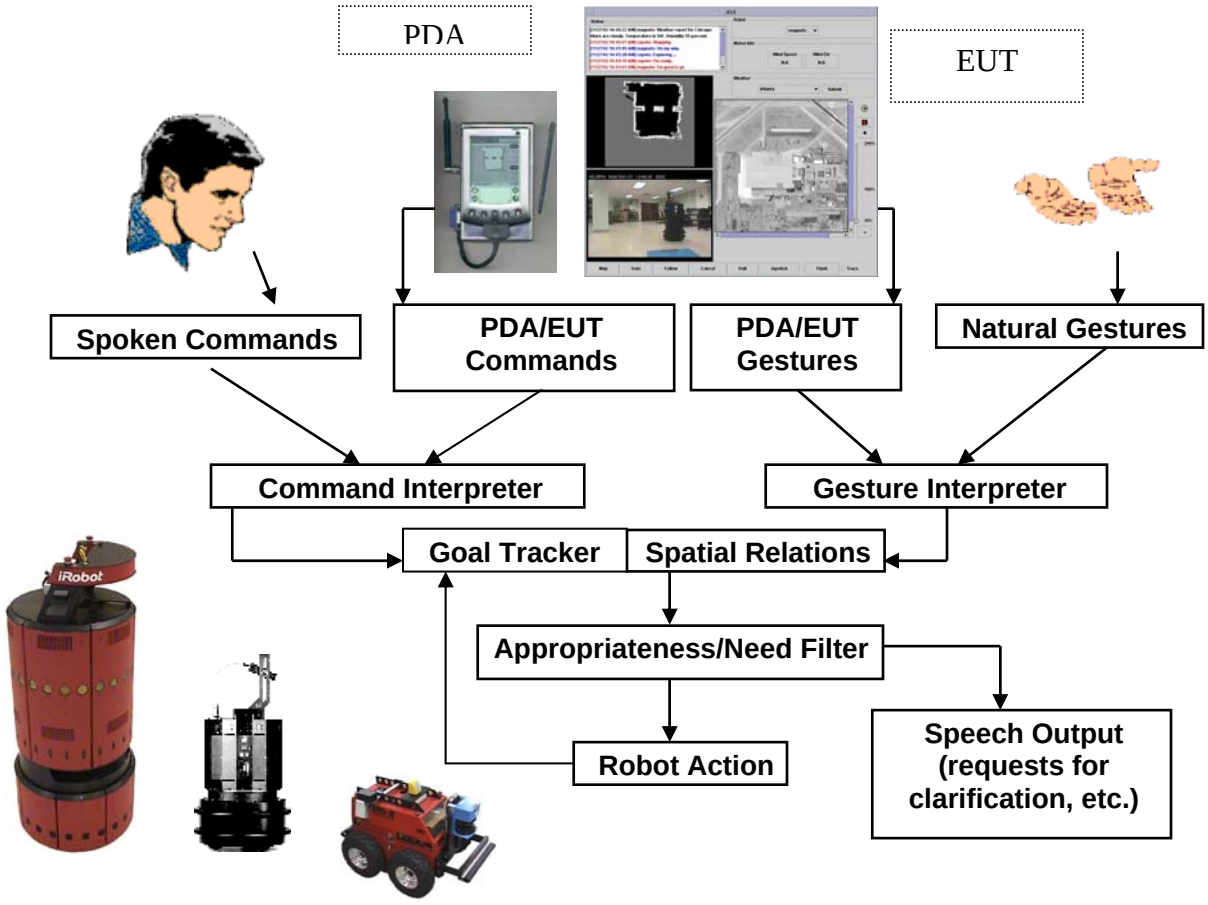


Fig. 1. Schematic overview of the multimodal interface,

using natural, Personal Digital Assistant and End User Terminal commands and gestures

Fig. 1 shows an expanded version of our PDA and touch screen interface which we call the End User Terminal (EUT). A map representation is available on both the PDA screen and on the EUT screen (middle left of EUT screen), along with a written history of the dialog (upper left), menu buttons (upper right and along lower edge), and a satellite view (middle right) for outdoor scenarios. We have been using Nomad 200, ATRV-Jr, and B21r robots.

By providing the human user with a "natural" interface [9], humans can issue spoken commands and gesture (Figs. A1 and A2), as well as direct a robot to interact with objects in its environment (Fig. Z).

Fig. A.1. A human user points to a location with the utterance “Coyote, go over there.”

Fig. A.2. A human user indicates a distance manually with the utterance “Coyote, back up this far.”

Fig. Z. The human user utters “Coyote, go to the right of the pillar” with a superfluous gesture.

Less natural modes of communication, such as pointing and clicking on a Personal Digital Assistant (PDA) screen (Fig. B) and on a touch screen (Fig. C), are also available for modes of interacting with the mobile robot, to allow the user a richer number of choices, and to permit interactions in situations where natural language and gesture may not be appropriate, such as when distance or environmental conditions prevent use of the more natural modes of communication.

Fig. B. A human user indicates a location using a Personal Digital Assistant.

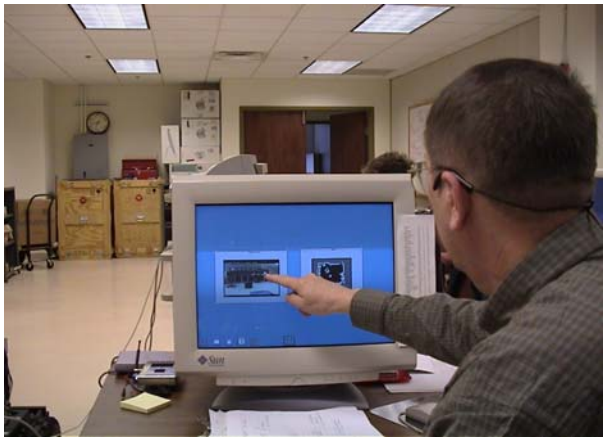


Fig. C. The human user, wearing a wireless microphone, interacts with a robot via a touch screen and speech.

Given the various components of the interface in Fig. 1, we will discuss how the various inputs are processed and show how natural language and gestures of various types are combined to produce either a Robot Action or appropriate Speech Output.

(A-F) show the mapping of a spoken utterance to the corresponding robot command.

(A)

"COYOTE GO OVER THERE" (B)

```
((ADDRESS (NAME N4 (:CLASS SYSTEM) COYOTE)                    (C)
          (IMPER #:V7756 (:CLASS GESTURE-GO)
          (:AGENT (PRON N6 (:CLASS SYSTEM) YOU))
          (:GOAL (NAME N5 (:CLASS THERE) THERE)))))

(COMMAND (FORALL X7 (SETOF N4 SYSTEM)                         (D)
                (EXISTS! X8 (SETOF N5 THERE)
                    (GESTURE-GO :AGENT X7 :GOAL X8))))
```

(2 45 4590231892.67254106) (E)

"4 45 obstacle" (F)

The various spoken commands or clicking on the PDA or EUT screens produce commands that are sent to the Command Interpreter. As part of the Command Interpreter in Fig. 1, the ViaVoice Speech Recognition System analyzes the acoustic signal (A), producing a text string (B) as output. This string is then analyzed by Nautilus [15], our in-house natural language understanding system, to obtain a semantic representation (C), which is then mapped to a representation (D) similar to a logical form used in propositional logic.[2] Gestures from the different sources--PDA screen, EUT terminal, and a structured light rangefinder mounted on the robot--provide input to the representation (D.

The Goal Tracker stores linguistic and state information in a structure which we call a *context predicate* [16] where it can be retrieved at a later time if necessary. Thus, for example, if an action is stopped, the user can command the robot to continue the stopped action at a later time, given that the

information is stored in the Goal Tracker. We return to a more detailed discussion of the context predicate and other semantic issues in the next section (Section IV).

Most of our commands have been direction- or location-oriented, such as "Coyote, go over there," "Coyote, go to the door over there," and "Coyote, go to the left of the pillar." Consequently, we have been addressing issues regarding spatial relations and reasoning. However, humans and robots represent the world differently. Robots use rotational matrices, for example, while humans use qualitative descriptions to describe location information, such as "on your right" or "at 3 o'clock" (the latter being a common military or aviator's way to describe the position of an object on an imaginary clock surrounding the speaker/hearer).

The Spatial Relations component (Fig. 1) provides necessary object and location information to enable to user and the robot to communicate about those elements in a very natural way. Because of the kinds of human-robot interactions which we have been considering, spatial relationships are critical. This component extracts spatial relations from sensory information and translates them into linguistic constructs that people naturally use to accomplish navigation tasks [3,17,18]. We will consider these and other related issues of the Spatial Relations component in greater detail in Sections V and VI.

The Appropriateness/Need Filter (Figure 1) determines if an accompanying gesture is necessary and whether or not an appropriate command or query has been issued. This is verified by the first element of the list in (E). Here "2" indicates that a natural vectoring gesture has been perceived. With the linguistic and gestural information, a robot message (F) is sent to the robotics modules for interpretation and mapping into navigation commands which will not concern us here. In (F), "4" is arbitrarily mapped to certain functions which cause the robot to move. The second element "45" in (F) indicates that the user vectored 45 degrees from an imaginary line connecting the user and the robot. Finally, the robot will move toward some obstacle in the direction of the human's gesture, close enough to avoid a collision with the obstacle.[3] This is translated in the robot component to an appropriate Robot Action.

In the event that the command is not linguistically well-formed, or an appropriate gesture is not perceived, an appropriate message is returned to the user for subsequent error handling in the Speech Output component of Fig. 1. These messages are usually synthesized responses or utterances informing the user that some error has been detected, such as if the user in Figures A.1 or A.2 does not provide gesture information. In these instances, the robot responds, “Where?" or "How far?" accordingly. Providing information during error handling is an important aspect of the interface, allowing the user to respond intelligently, quickly, and easily to situations.

IV. The Semantics of Spatial Language and Linguistic Consequences

The Command Interpreter produces semantic interpretations of the commands in a structure which we call a *context predicate*[4]. Consider the following dialog (1) which is mapped into a *context predicate* (2).

DIALOG:

Human: “Coyote, go over there.” (1)

<an accompanying gesture indicating a location is generated by the human>

Human: “Coyote, how many objects do you see?”

Coyote: “I am sensing five objects.”

CONTEXT PREDICATE:

((ADDRESS ((:CLASS SYSTEM) COYOTE) (2)

(IMPER

((:VERB-CLASS GESTURE-GO) GO)

(:AGENT (:CLASS SYSTEM) YOU)

(:GOAL (:CLASS THERE) THERE)

(:GOAL-STATE INCOMPLETE)))

(ADDRESS ((:CLASS SYSTEM) COYOTE)

(REQUEST (HOW MANY)

((:VERB-CLASS P-SEE) SEE)

(:AGENT (:CLASS SYSTEM) COYOTE)

(:THEME (:OBJECT PLURAL))

(:GOAL-STATE COMPLETE))))

The context predicate (2) is a stack of lists containing semantic information obtained from the Command Interpreter and state information obtained by determining whether or not a Robot Action is completed or not. The action of going to a location, for example, requires a destination, expressed by the goal *there*. Requesting how many objects are sensed causes the Command Interpreter to note that a *request* is being made, quantification issues, *how many,* need to be settled about the *:theme*, namely *object*s. Gesture information, obtained from the Appropriateness/Need Filter, is currently not stored in the context predicate but is obtained elsewhere in the robotics modules. Future research may determine that this is an oversight; however, adding this information to the context predicate is a trivial matter.

In the sample dialog (1), while the human initially orders the robot to go to a certain location, the human immediately follows the command with a request for information about the number of objects Coyote is seeing. The first command, therefore, is incomplete while Coyote attempts to sense the number of objects. However, Coyote can respond with an appropriate answer, and then return to the uncompleted goal; namely, going to a particular location. Information about incomplete/completed goals is obtained from the *goal-state* arguments in the *context predicate*. As goals are obtained, Robot Actions are produced, and the *goal-state* is updated.

Along with tracking goal states, important spatial information must be obtained and updated, since many utterances in a command and control domain involve spatial references. Knowledge of objects and their locations in the immediate environment requires a rich semantic representation.

Given the sensor information and the linguistic descriptions produced by the Spatial Relations component (Sections V and VI), we found that the semantic representations we had been using lacked adequate locative and spatial representations to reason about spatial information. Initially, for example, it

was sufficient for us to know that commands involving locative information, such as (7), could be represented as (8), a somewhat simplified representation for expositional purposes here.

"Coyote, go over there." (7)

(imper: (8)
((p-go: go)
(:agent (system coyote))
(:loc (location there))))

The term *imper* in (8) is an abbreviation the *imper*ative command of (7). (8) further analyzes the command *go* into a class of semantic predicates *p-go*. *p-go* requires certain semantic roles or arguments to be filled, such as an :*agent* role that is the grammatical subject of the sentence. The :*agent* must be semantically classified as a *system* which is how we *Coyote* defined. Finally, *p-go* requires location information, a *:loc* role, the word *there* that is semantically subcategorized as a *location.*

Given this semantic framework, the commands of (9a,b) generate the same semantic representation (10).

"Coyote, go to the elevator." (9a)

"Coyote, go into the elevator." (9b)

(imper: (10)
((p-go: go)
(:agent (system coyote))
(:loc (location elevator))))

However, (10) misses crucial semantic information, namely, that the ultimate locative goal or location is just in front of the elevator (9a) versus inside it (9b). We, therefore, had to expand our representations.

It is not immediately apparent whether (11a,b) or (12a,b) are adequate representations for the utterances in (9a,b).

(imper: (11a)
((p-go-to: go)
(:agent (system coyote))

```
            (:loc (location elevator))))
(imper:                                                    (11b)
      ((p-go-into: go)
            (:agent (system coyote))
            (:loc (location elevator))))

(imper:                                                    (12a)
      ((p-go: go)
            (:agent (system coyote))
            (:to-loc (location elevator))))
(imper:                                                    (12b)
      ((p-go: go
            (:agent (system coyote))
            (:into-loc (location elevator))))
```

(11a,b) compound the number of predicates *go* maps to; namely *p-go-to* and *p-go-into*. (12 a,b) realize only one semantic predicate *p-go* but compound the number of roles of the predicate; namely, *:to-loc* and *:into-loc*.

Both representations capture the locative information for crucially differentiating (9a) and (9b). However, rather than claiming there are several semantic predicates corresponding to the English verb *go,* as realized by the different classes *p-go-to* and *p-go-into* (11a,b), (12a,b) capture the generalization that the English verb *go* maps to a single semantic predicate having multiple roles. Therefore, we choose (12a,b) as adequate semantic representations. Our empirical justification for opting for these representations is simplicity. It seems more intuitively appealing to claim that *go* is a singular semantic verbal concept, taking various locative roles. This conclusion is in keeping with a model-theoretic approach explaining the semantics of locative expressions [24].

Following this line of reasoning, we were able to simplify the representations for sentences like (13) and generalize about other locations, such as elevators and exhibit halls.

"Coyote, the elevator is in front of the exhibit hall." (13)
"Coyote, the elevator is behind the exhibit hall."
"Coyote, the elevator is opposite the exhibit hall."
"Coyote, the elevator is across from the exhibit hall."
"Coyote, the elevator is across the exhibit hall."

Rather than compounding a list of semantic predicates to interpret (13), we map the predicate *be*, syntactically the verb *is* in (13), to a single semantic predicate that we arbitrarily name *be-at-location* having several locative roles (14).

```
(be-at-location: be                                          (14)
        (:theme (location))
            (:in-front-of-loc (location))
            (:behind-loc (location))
            (:relatively-opposite-loc (location))
            (:directly-opposite-loc (location)))
```

In this semantic framework, we maintain the intuitive notion that being in a location is a single semantic concept or predicate, and the actual location is stipulated specifically by a role. In English, this is usually realized as a locative preposition. Therefore, locative and spatial information is mapped to semantic roles of predicates of the domain rather than to different predicates of the domain.

This conclusion may prove to be of interest to the linguistics community. While our results are language-specific, namely to English, research in the semantics of locative and spatial expressions in other languages may show that our claim can be extended to other languages, and to interfaces employing those languages.

V. Generating Spatial Language from Occupancy Grid Maps

A. Preprocessing

The map structure used in this work is an evidence grid map [8]. The indoor environment shown in this paper is represented with a 128 x 128 x 1 cell grid, providing a two-dimensional map of the NRL lab. One cell covers approximately 11cm x 11cm on the horizontal plane of the sonars. Information from the robot sensors is accumulated over time to calculate probabilities of occupancy for each grid cell. One byte is used to store occupancy probabilities; values range from +127 (high probability of occupancy) to -127 (high probability of no occupancy), with 0 representing an unknown occupancy. For the work reported here, these maps are the sensor-fused short-term maps generated by the robot's regular localization and navigation system [19]. Examples of evidence grid maps are shown in Fig. 2(a) and 3(a). For our

purposes, a cell with an occupancy value ≥ +1 is considered to be occupied and is shown in black. All other cells are shown in white.

The evidence grid map is pre-processed with a sequence of operations, similar to those used for image processing, to segment the map into individual objects. First, a filter is applied through a convolution operation. A 3x3 matrix, shown below in (3), is used as the convolution kernel, *K*, to provide a linear filter of the map.

$$K = \begin{vmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{vmatrix} \qquad (3)$$

This has the effect of blurring the map, filtering single cells and filling in some disconnected regions, as shown in Fig. 2(b).

An explicit fill operation is also used to further fill in vacant regions. For each unoccupied cell, if 5 or more of its neighbors are occupied, then the cell status is changed to occupied. Eight neighbors are considered, as shown below in (4) for cell $a_{i,j}$:

$a_{i-1,j-1}$	$a_{i,j-1}$	$a_{i+1,j-1}$
$a_{i-1,j}$	$a_{i,j}$	$a_{i+1,j}$
$a_{i-1,j+1}$	$a_{i,j+1}$	$a_{i+1,j+1}$

(4)

Two passes of the fill operation are executed. Results are shown in Fig. 2(c).

Finally, spurs are removed. A spur is considered to be an occupied cell with only one occupied neighbor in the four primitive directions (diagonal neighbors are not counted). All spurs, including those with a one-cell length, are removed. At this point, the final cell occupancy has been computed for object segmentation. Objects should be separated by at least a one-cell width.

Next, objects are labeled and loaded into a data structure for spatial reasoning. A recursive function is used to label adjacent cells. Occupied cells are initially given numeric labels for uniqueness, e.g., object #1, object #2. Once the cells are labeled, a recursive contour algorithm is used to identify the boundary of the objects. The contour is important in that it provides a representation of the environment

obstacles that is used for spatial reasoning. Examples of the final segmented objects, with their identified contours, are shown in Fig. 2(d) and 3(b).

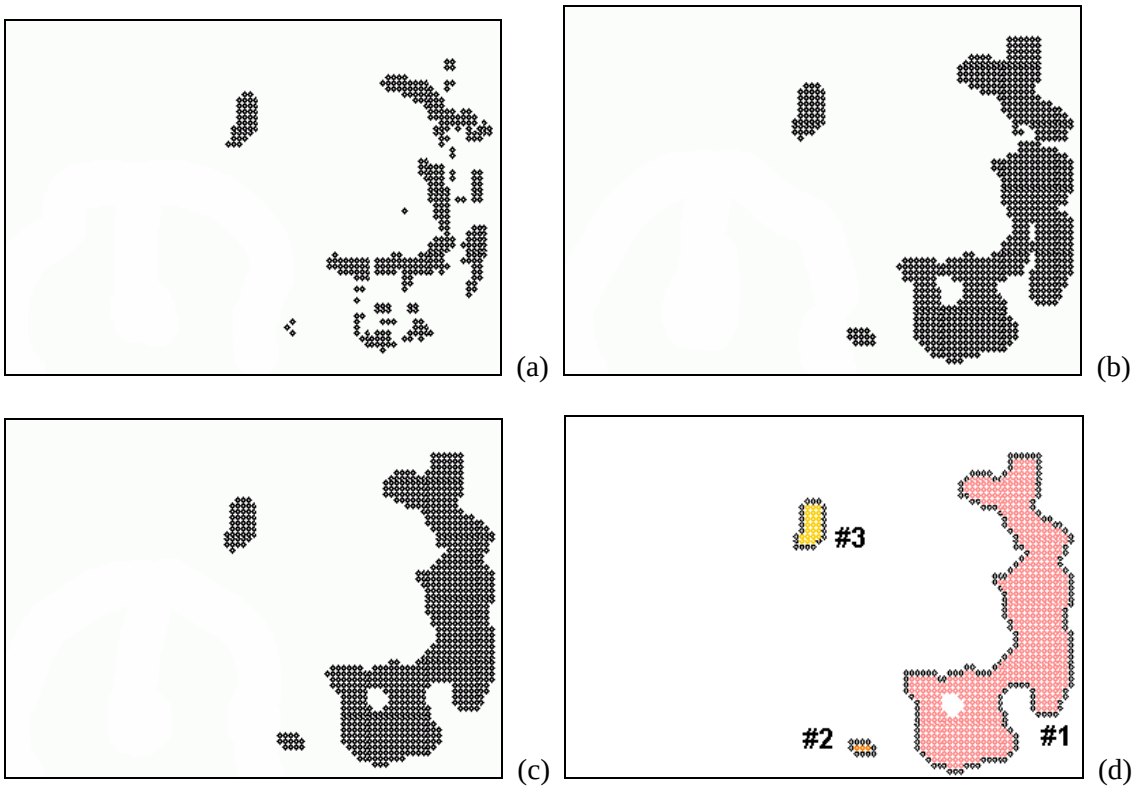


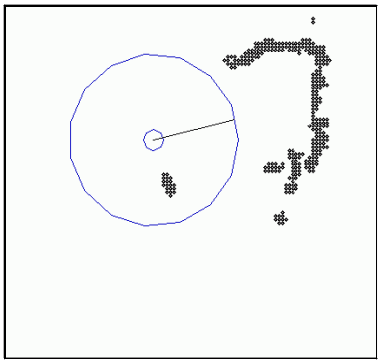
Fig. 2. (a) The southeast part of the evidence grid map. Occupied cells are shown in black. (b) The result of the filter operation. (c) The result of the fill operation. (d) The segmented, labeled map. Physically, object #1 corresponds to a section of desks and chairs, object#2 is a file cabinet, and object #3 is a pillar.

B. Modeling spatial relationships

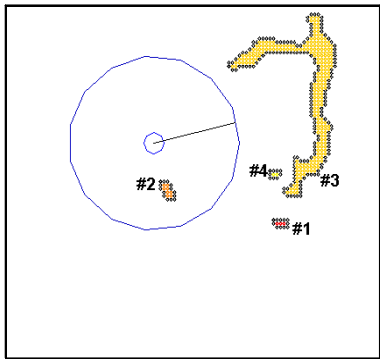
Spatial modeling is accomplished using the histogram of forces [20], as described in previous work [3,4,5,17,18,21]. For each object, two histograms are computed (the histograms of constant forces and gravitational forces), which represent the relative spatial position between that object and the robot. Computationally, each histogram is the resultant of elementary forces in support of the proposition object #i is in direction θ of the robot. For fast computation, a boundary representation is used to compute the histograms. The robot contour is approximated with a rectangular bounding box. The object boundaries are taken from the contours of the segmented objects in the grid map.

The two histograms give different views of the environment; the histogram of constant forces provides a global view and the histogram of gravitational forces provides a local view. Features from the histograms are extracted and input into a system of fuzzy rules to generate a three-part linguistic spatial description: (1) a primary direction (e.g., *the object is in front*), (2) a secondary direction which acts as a linguistic hedge (e.g., *but somewhat to the right*), and (3) an assessment of the description (e.g., *the description is satisfactory*). A fourth part describes the Euclidean distance between the object and robot (e.g., *the object is close*). In addition, a high level description is generated that describes the overall environment with respect to the robot. This is accomplished by grouping the objects into 8 (overlapping) regions located around the robot. An example of the generated descriptions is shown in Fig. 3(c). See [3,18] for additional details.

One of the features extracted from the force histograms is the main direction, α, of an object with respect to the robot. The main direction, which is used in later sections, has the highest degree of truth that the object is in direction α of the robot. See also [22] for details.



(a)



(b)

(c)

DETAILED SPATIAL DESCRIPTIONS:
The #1 object is to the right-front of me
-
(the description is satisfactory)
The object is close.

The #2 object is to the right of me
-
(the description is satisfactory)
The object is very close.

The #3 object is in front of me
-
(the description is satisfactory)
The object is close.

The #4 object is mostly in front of me
but somewhat to the right
(the description is satisfactory)
The object is close.

HIGH LEVEL DESCRIPTION:
There are objects in front of me and on my right.

Fig. 3. (a) A robot situated in the grid map. (b) The segmented, labeled map. (c) The generated descriptions. Note the robot heading. Object#2 corresponds to the same pillar in Fig. 2(d).

C. Modeling unoccupied regions for robot directives

To support robot commands such as "Go to the right of the object", we must first compute target destination points in unoccupied space, which are referenced by environment objects. These spatial reference points are computed for the four primary directions, left, right, front, and rear of an object, from the robot's view. In this paper, we consider a method of finding these destination points called the *Intersecting Ray Method*. This method uses the main direction from the constant forces histogram to calculate reasonable target points. The main direction is also used for the viewing perspective of the robot. That is, the spatial reference points are computed, as if the robot is facing the target object along the main direction.

Fig. 4 shows a diagram for the *Intersecting Ray Method*. As shown in the figure, a bounding box is constructed by considering the range of (x, y) coordinates that comprise the object contour. The bounding box is used as a convenient starting point for a search of key points along the object contour.

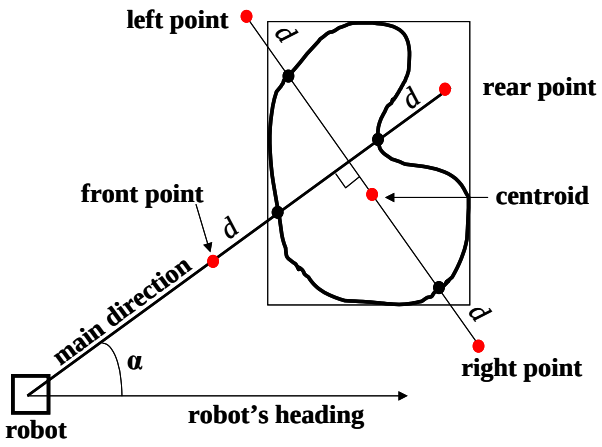


Fig. 4. The Intersecting Ray Method

The front and rear points are computed to lie on the main direction vector, at a specified distance, *d*, from the object boundary. Consider first the front point. Coordinates are calculated along the main direction vector using the following equations:

$$x = r\cos(\alpha)$$
$$y = r\sin(\alpha) \qquad (5)$$

where α is the main direction, (x,y) is a point along the main direction, and r is the distance of the vector from the robot to the (x,y) point. Coordinate points are computed incrementally, starting from the robot and checked for intersection with the object contour until the intersection point is identified. When the intersection point is found, the front point is computed by subtracting the distance, d, from v_F, the vector length of the front intersection point, and computing a new coordinate.

In computing the rear point, we again search for the intersection point of the contour along the main direction vector, this time starting from behind the object. The bounding box of the object is used to compute a starting point for the search. The algorithm first determines the longest possible line through the object by computing l, the diagonal of the bounding box. The starting vector length used in the search is then $v_F + l$. Once the rear contour intersection point is found, the rear point is computed by adding d to the vector length of the rear intersection point and computing a new coordinate.

The left and right points are computed to lie on a vector that is perpendicular to the main direction and intersects the centroid (x_C,y_C) of the object. Again, a search is made to identify the contour point that intersects this perpendicular vector. The starting point for the search of the right intersection point is shown below:

$$x = x_C + l\cos(\alpha - \frac{\pi}{2})$$
$$y = y_C + l\sin(\alpha - \frac{\pi}{2}) \qquad (6)$$

Once the intersection point is found, a new vector length is computed by adding the distance, d, and computing the new coordinate. The left point is found using a similar strategy. Fig. 5 shows some examples. The spatial reference points are marked with the diamond polygons around each object; the vertices of the polygons define the left, right, front, and rear points.

Although this method generally calculates "good" points, at times it will produce non-intuitive spatial reference points for certain objects. An example is shown in Fig. 6 on the object labeled 3. The

problem occurs with odd-shaped objects when the robot is positioned such that the rays for perpendicular directions, e.g., rear and left, both intersect the object on the same side. In this case, the resulting points may not appear to be what we, as humans, would naturally consider left and rear. As shown in Fig. 6 (object 3), having both the left and the rear points lie behind the object is non-intuitive at best. Likewise, the right and front points are computed to be in the front, which is also not a natural configuration.

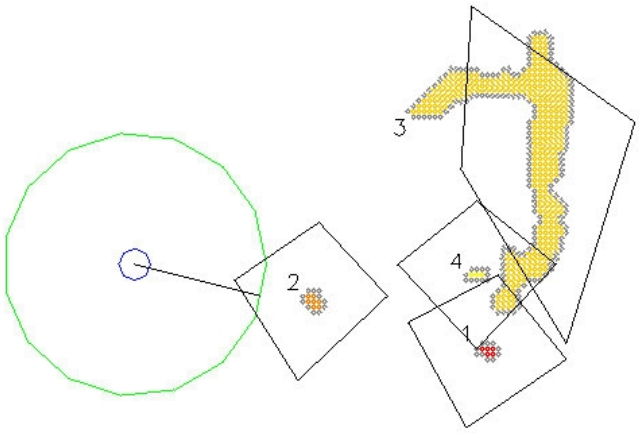


Fig. 5. Computing left, right, front, and rear spatial reference points using the Intersecting Ray Method. The vertices of the polygons mark the positions of the left, right, front, and rear points.

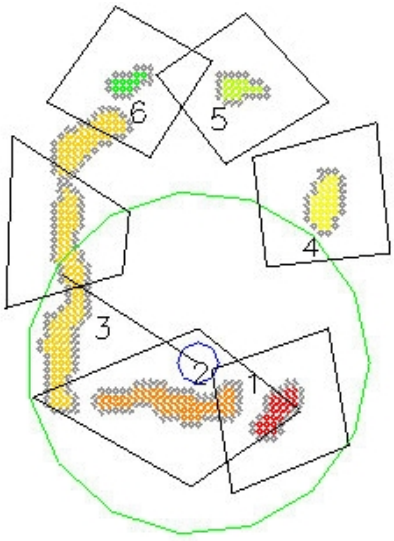


Fig. 6. More examples using the Intersecting Ray Method. Note the non-intuitive reference points for Object 3.

In order to resolve these problems, we must determine a method to evaluate the correctness of the left, right, front, and rear points. This evaluation can be accomplished using the same spatial modeling

algorithms but switching the argument and the referent (the robot and the object). The degrees of truth are then calculated by placing the robot at the spatial reference points computed. For example, we place a virtual robot at the left reference point and then run the analysis to determine whether the robot really is to the left of the object. The resulting degree of truth is interpreted as a confidence level. In fact, by placing a virtual robot at neighboring positions, this technique can be used to investigate regions that are to the left, right, front, and rear of an object, where the areas are segmented by confidence level.

Fig. 7 shows an example of regions created using this technique, computed for Object 2. Regions for left, right, front, and rear are shown in the figure (from the robot's perspective). The medium gray regions (green) represent a high confidence level, where the cell i confidence, $c_i \geq 0.92$. The light gray regions (yellow) have a medium confidence level ($0.8 < c_i < 0.92$). The dark regions (red) have a low confidence level ($c_i \leq 0.8$).

The figure shows that the regions widen as the distance from the object increases, and they join each other in a logical manner. For a relatively small object, the left, right, front, and rear reference points computed by the Intersecting Ray Method lie well within the high confidence region, as shown by the polygon vertices.

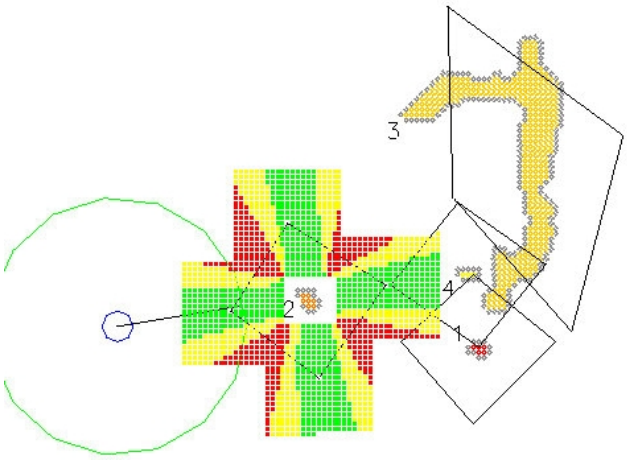
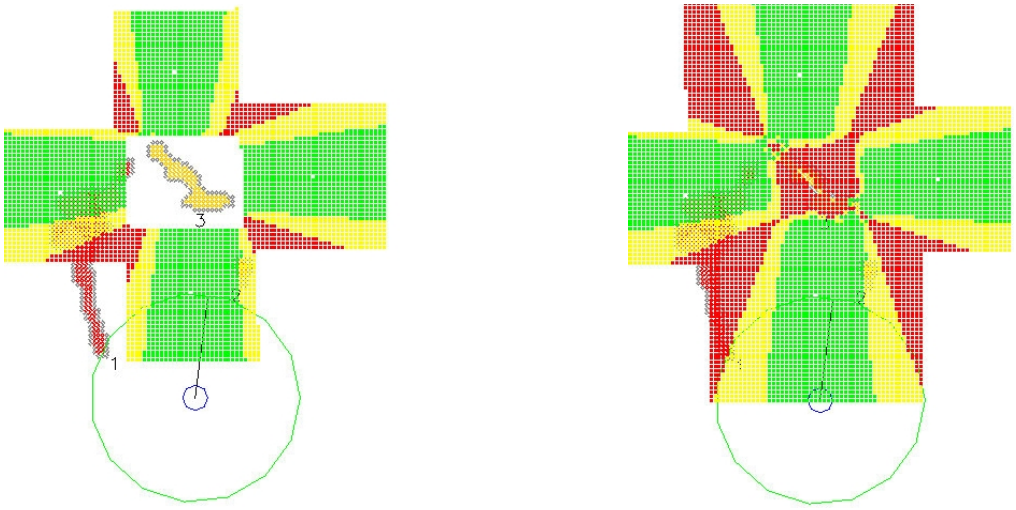


Fig. 7. Confidence Regions around Object 2 for left, right, front, and rear spatial references, from the robot's perspective. Medium gray (green) is high confidence. Light gray (yellow) is medium confidence. Dark gray (red) is low confidence.

Fig. 8 shows another example of spatial regions, this time computed for a long object, from a diagonal perspective. Fig. 8(a) shows the object orientation. Fig. 8(b) shows the extended regions for the

same situation (although the object is no longer visible). Again, the Intersecting Ray Method computes points within the high confidence regions, as long as the specified distance is far enough from the object. Fig. 8(b) shows the confidence levels of the areas close to the object; it is easy to see that if a close distance is specified, the algorithm could compute reference points in the medium or low confidence regions.



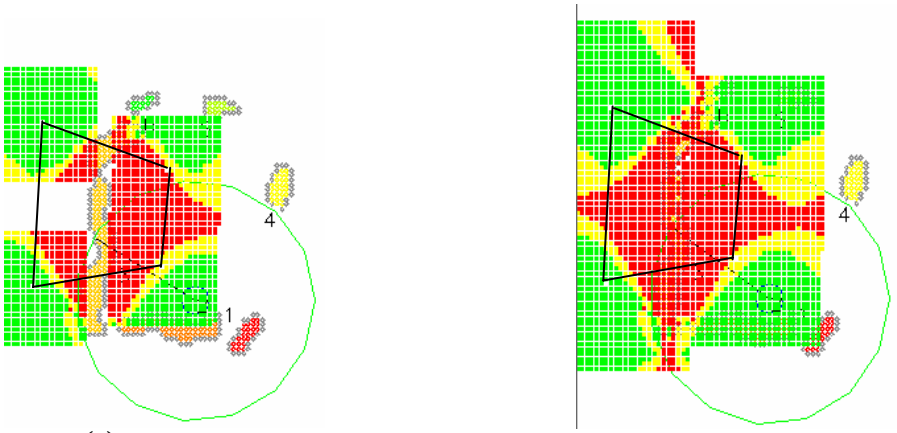
(a) (b)

Fig. 8. Confidence Regions around Object 3 for left, right, front, and rear spatial references. (a) A partial view showing Object 3 (b) Extended regions.

Finally, let us return to the situation in Fig. 6, where non-intuitive points were computed. This case is repeated in Fig. 9 to show confidence regions for object 3. In this case, the distance from the object must be quite high for all four computed reference points to be in the high confidence region. Even in the example shown, the front and right points are outside of the high confidence area. Thus, we still need a method to establish the validity of a reference point and recompute it if the result is not in a high confidence region.

The computed confidence regions offer a more complete picture of the object's spatial regions, but it is not practical to compute them for each point due to the computational complexity. Our final method will most likely employ the Intersecting Ray method to calculate the reference points; then by using those

points as starting values, we will perform a hill-climbing algorithm to move the points into the proper region if necessary, with the robot and object switched in the histogram calculations. Hopefully, this final method will allow us to definitively say that the points are in a region considered to be "right," both from a computational and intuitive perspective.



(a) (b)

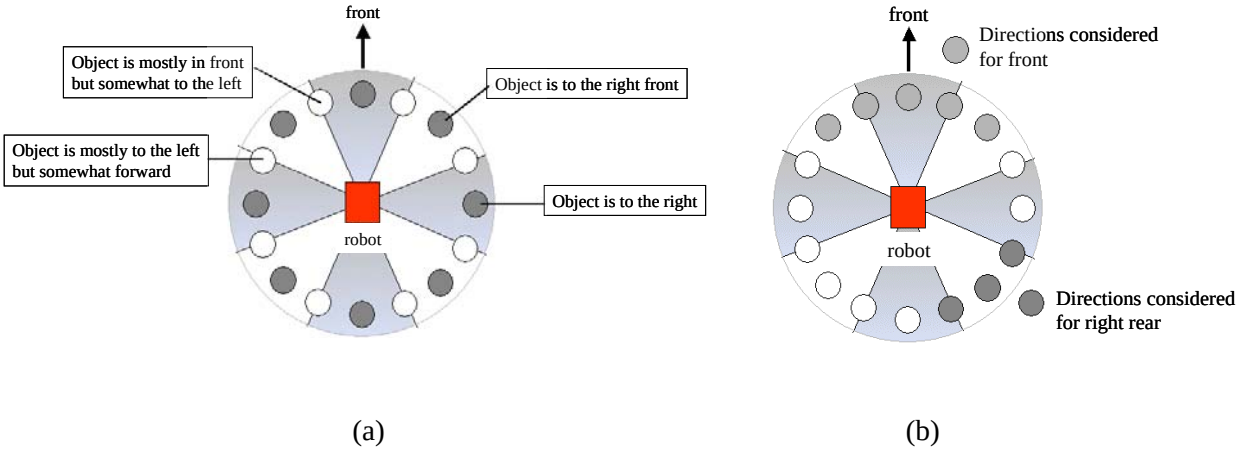
Fig. 9. Confidence Regions for Fig. 6, Object 3. (a) A partial view showing the object. (b) Extended regions.

IV. Integrating Spatial Language into Human-Robot Dialog

The robot control system has been implemented as a distributed system with components for path planning, map processing, localization, navigation, and handling the various interface modalities (PDA, gesture, and speech input). The spatial reasoning capabilities have been integrated into this environment in the form of a server so that any client can request the spatial description of the environment at any given time.

As the descriptions are generated, information is also stored on the relative spatial positions of the environment objects to facilitate a meaningful dialog with the robot. As shown in Fig. 10, sixteen symbolic directions are situated around the robot. The main direction of each object, as computed from the force histograms, is discretized into one of these 16 directions. Examples of some corresponding linguistic descriptions are shown in Fig. 10(a). In addition, the 16 symbolic directions are mapped to a set of 8 overlapping regions around the robot (left, right, front, rear, and the diagonals), which are used for queries. Two examples are shown in Fig. 10(b). An object in any of the 5 light gray directions is

considered to be in front of the robot. An object in any of the 3 dark gray directions is considered to be to the right rear. Thus, an object that is to the right front (as shown in Fig. 10(a)) would be retrieved in queries for three regions: the front, the right, and the right front of the robot.



(a) (b)

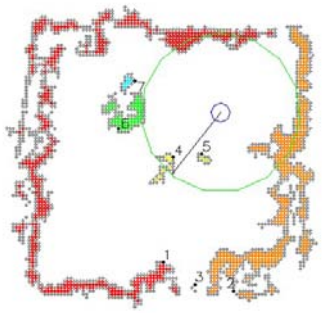
Fig. 10. Sixteen directions are situated around the robot (the small circles). The main direction of each object is discretized into one of these 16 directions. The 8 cone-shaped sections represent the 8 basic regions (front, rear, left, right, and diagonals) used for queries. (a) Examples of the corresponding linguistic descriptions. (b) Examples used for queries. An object is considered to be in front of the robot if it occupies one of the 5 light gray directions. Diagonal directions such as right rear comprise only 3 directions (dark gray).

With this spatial information and linguistic descriptions, we can now establish a dialog using spatial language. Queries can be made with the 8 basic regions illustrated in Fig. 10. Consider the scene in Fig. X for the dialog below.

Fig. X. Querying the robot for the number of objects

Human: "How many objects do you see?"
Robot: "I am sensing 7 objects."
Human: "Where are they?"
Robot: "There are objects in front of me."
Human: "Where is the nearest object in front of you?"
Robot: "The object # 5 is mostly in front of me but somewhat to the left. The object is very close."
Human: "Where is the nearest object on your right?"
Robot: "The object #7 is mostly to the right of me but somewhat forward. The object is close."
Human: "Where is the nearest object on your left?"
Robot: "Object #2 is to the left-front of me. The object is far."

Information about the objects is obtained by querying the Spatial Relations component. Sensor information, graphically represented in Fig. Ya, is mapped to linguistic expressions (Fig. Yb).



DETAILED SPATIAL DESCRIPTIONS:
The object number 1 is mostly in front of me
but somewhat to the left
(the description is satisfactory)
The object is close.

The object number 2 is to the left-front of me
(the description is satisfactory)
The object is far.

The object number 3 is mostly in front of me
but somewhat to the left
(the description is satisfactory)
The object is far.

The object number 4 is in front of me
but extends to the right relative to me
(the description is satisfactory)
The object is very close.

The object number 5 is mostly in front of me
but somewhat to the left
(the description is satisfactory)
The object is very close.

The object number 6 is to the right-front of me
(the description is satisfactory)
The object is close.

The object number 7 is mostly to the right of
me but somewhat forward
(the description is satisfactory)
The object is close.

HIGH LEVEL DESCRIPTION:
There are objects in front of me.

(a) (b)

Fig. Y. Map representation (a) showing the descriptions (b) of sensed objects

While the human in Fig. X is face to face with the robot, note that the user has all of the interface modalities available and can view a graphical display of the scene while talking to the robot if so desired. The robot responds using synthesized speech output [23].

Motion directives can also be given using spatial language, as shown in the dialog below:

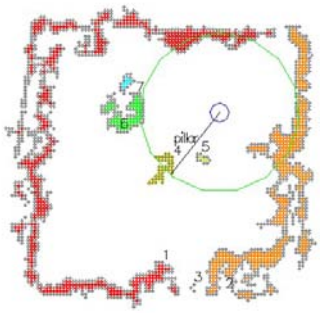
Human: "Go to the nearest object in front of you."
Robot: "Looking for the nearest object in front of me." "I found it. Going to the object."

Such information as "nearest" and "in front of" are calculated from the sensor data but then mapped to linguistic expressions, resulting in a natural exchange of information in the human-robot dialog.

We have also defined a class of persistent objects that have a designated location and are assigned a label by the user. Persistent objects are created from the objects identified in the grid map, through a dialog with the robot. Assuming that the human user is looking at the map representation (Fig. Ya) and knows that object #4 can be identified as a more familiar object, the user can name the object for the robot.

Human: "The object #4 is a pillar."
Robot: "I now know that the object #4 is a pillar."

The robot can confirm the labeling of the object by mapping the information to linguistic structures in the spoken dialog, as well as by drawing information in the graphics (Fig. YYa). The user can also obtain more information about the objects, including the object now labeled "pillar," by querying the system and asking questions like "What objects do you see?" and obtain the output in Fig. YYb.



DETAILED SPATIAL DESCRIPTIONS:
The object number 1 is mostly in front of me
but somewhat to the left
(the description is satisfactory)
The object is close.

The object number 2 is to the left-front of me
(the description is satisfactory)
The object is far.

The object number 3 is mostly in front of me
but somewhat to the left
(the description is satisfactory)
The object is far.

The pillar is in front of me
but extends to the right relative to me
(the description is satisfactory)
The object is very close.

The object number 5 is mostly in front of me
but somewhat to the left
(the description is satisfactory)
The object is very close.

The object number 6 is to the right-front of me
(the description is satisfactory)
The object is close.

The object number 7 is mostly to the right of
me but somewhat forward
(the description is satisfactory)
The object is close.

HIGH LEVEL DESCRIPTION:
There are objects in front of me.
The pillar is in front of me.

(a) (b)

Fig. YY. Map representation (a) showing the identification of object #4 as a *pillar* and (b) the linguistic descriptions of all objects perceived

As the robot moves around the environment, it remembers where the pillar is and will continue to generate spatial information about the pillar.

In Fig. Z and the following dialog, the human user interacts with the robot, who now share additional information about an object in the environment.

[I MOVED Fig.Z: possible another figure or reference back to the original will work]

Human: "Where is the pillar?"
Robot: "The pillar is in front of me but extends to the right relative to me. The object is very close."
Human: "Go to the right of the pillar."

Information in the dialog continues to be obtained through queries to the detailed spatial and high level descriptions in the Spatial Relations component.

To execute the final command in the dialog and to locate the desired position relative to the pillar, the cells occupied by the pillar are used to calculate the left and right sides of the persistent object. Graphically, the corresponding Fig. 11 is constructed, using the same positioning and array of objects in Fig. YYa.

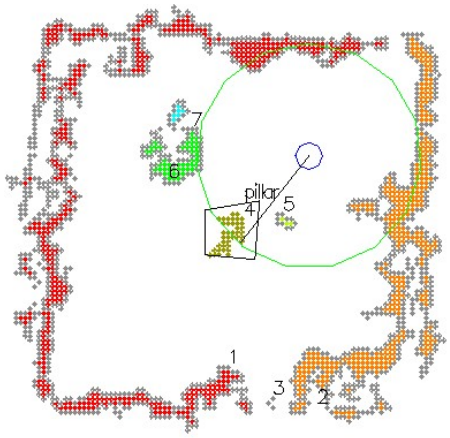
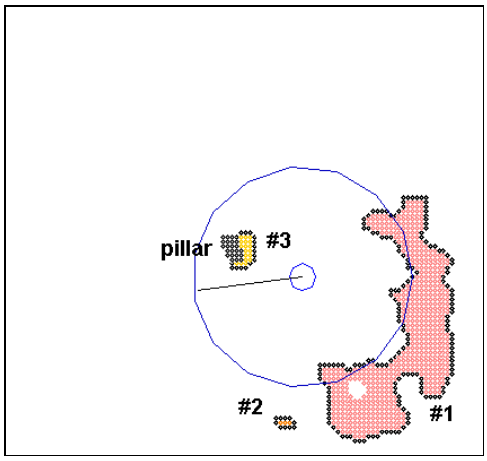


Fig. 11. The robot, situated in the segmented map, generates a persistent object, a pillar.

An appropriate Robot Action ensues (Fig. 111).

Fig. 111. Robot moves to a position to the right of the pillar.

Consider a slightly different arrangement of sensed objects in Fig. 12. The pillar and object #3 are the same although they do not occupy exactly the same grid cells. Note, however, that the description of both object #3 and the pillar, as shown in Fig. 12(b), are exactly the same. A simple algorithm matches the detailed spatial descriptions and the distance values (within a range) to determine that object #3 is in fact the pillar. As shown in Fig. 12(b), a merged High Level Description is generated in which object #3 is replaced with the pillar. In future work, we will explore additional algorithms for connecting persistent objects with those identified dynamically from the grid map.



DETAILED SPATIAL DESCRIPTIONS:
The #1 object is behind me
but extends to the left relative to me
(the description is satisfactory)
The object is very close.

The #2 object is mostly to the left of me
but somewhat forward
(the description is satisfactory)
The object is close.

The #3 object is mostly in front of me
but somewhat to the right
(the description is satisfactory)
The object is very close.

The pillar is mostly in front of me
but somewhat to the right
(the description is satisfactory)
The object is very close.

HIGH LEVEL DESCRIPTION:
The #1 object is behind me.
The #2 object is mostly to the left of me.
The pillar is mostly in front of me.
I am surrounded from the rear.

(a) (b)

Fig. 12. (a) The scene from Fig. 3 with the persistent object "pillar". (b) The generated detailed and High Level descriptions.

Finally, Fig. 12a also shows an example of the *surrounded* relation, which provides additional capabilities of high level spatial reasoning, as it is provided in the High Level Description in Fig. 12b. In

[18], we introduce 3 levels of surrounded based on the width of the force histograms, e.g., (1) *I am surrounded on the right*, (2) *I am surrounded with an opening on the left*, and (3) *I am completely surrounded*.

XY. A Pilot Study—perhaps to be used and/or integrated or not into Evaluation methods in VI.

To obtain some preliminary information about the usefulness and habitability of our multimodal interface, we conducted a pilot study. In this study, human users were asked to interact with a mobile robot in another room. Subjects were seated at a desk in front of a touch screen. The robot's eye view and a mapped representation of its environment appeared on the screen. The subjects were told that they had to find a hidden object using the mobile robot. The object was a yellow sign with the word "FOO" printed on it. Subjects were further told that they could interact with the robot by way of natural language, using a wireless microphone. They could also hear feedback from the robot through headphones and they could touch the displays, thereby using gestures to indicate locations and objects for the robot. Subjects were not told, however, that this was a Wizard-of-Oz experiment.

Two researchers associated with the human-robot interface project for several years were always behind the robot out of camera view for the subjects: one Wizard executed the commands, interpreting the gestures indicated, navigating the robot with a joystick; the other, whose voice was modulated to sound more "robot-like," responded to the subjects, corroborating moves and asking for indicating that a certain command was not do-able. While we wanted to give the subjects the impression that the robot's were extremely robust, and they could perform whatever the human's requested in order to complete the task, several subjects asked the robot to "look up/down." Because the current vision system is stationary, subjects had to be told, "I'm sorry, but I cannot do that."

Since this was not a rigorous experiment, we do not make any claims regarding the validity or statistical significance of these preliminary results. We simply report here our observations from this pilot study, indicating the kind of experiment that we wish to pursue in the near future with greater scientific rigor.

Five subjects participated in the pilot study, but the first person was used as a debugging subject so that equipment, software, instructions and other setup problems could be ironed out. Two of the subjects were female and the others were male; all were adults. In terms of utterances, use of gesture, and use of the two kinds of visual display (the camera view and the map view), there were pronounced individual differences in how the subjects conceptualized the task, used the interface, and interacted with the robot.

All of the subjects quickly became comfortable with talking to the robot, but each chose to speak to the robot in a different way. Two frequently instructed the robot to turn a certain number of degrees one way or another, such as “Turn left twenty-five degrees.” Another instruction used later by one of these subjects was to turn "a little" or "a little more." The other subjects seemed comfortable giving less precise instructions, such as "turn left," "turn around, or “move slightly forward." One subject navigated for the most part without referring to any of the objects in the room.

While the subjects were told that they could point and talk about objects in the room, most of their interactions seemed to be devoid of spatial references, other than “here” and “there.” One subject did refer to some of the objects, such as “computer boxes”, “a pillar,” and “bookshelves.” When asked about the kind of linguistic exchange that occurred with one subject, the subject admitted that the subject’s personal use of speech recognition software in a commercial operating system had biased the user to use simple commands, rather than to explore the capabilities of the current system.

Two subjects showed a strong preference for tightly managing the robot's actions, whereas the other subjects seemed willing to let the robot exercise a degree of autonomy as the interaction progressed. All but one of the subjects made frequent, verbal deictic references such as "go here" or "go there" and corroborated them by pointing to locations on the touch screen. Although the remaining subject also used a number of deictic references, this person made essentially no use of the touch screen.

Subjects seemed to differ as to which kind of visual display they preferred to use. Whereas one subject felt that both displays were extremely helpful, another subject felt that the map view of the task was completely unnecessary. Yet another subject interacted almost entirely with the map view and felt the camera view was only needed to identify the FOO.

All of the subjects seemed to enjoy working with the robot and carrying out the task. When asked, all rated the interaction experience positively and gave no indication that they did not believe they were truly interacting with a machine.

However, we were surprised by the relatively narrow use of the interface modes. We had provided the user with fairly robust linguistic and spatial reasoning modules, but users seemed to approach the interface with some degree of caution or conservatively, not pushing the system beyond what seemed to be a rather narrow range of the system's overall capabilities. It seems, therefore, that we need to design adaptive interfaces that gradually reveal more options in a particular mode of operation to best suit the naive user. While we want to reduce the learning curve for using an interface, some initial instruction—more than we gave—would benefit the naïve user.

Since our subjects seemed to exhibit different ways of interacting, we can't just provide a host of tools for the user and expect them to gravitate naturally to their own preferred modes of interaction and figure out how the interface can meet their needs. On the other hand, an adaptive interface might overcome some of the initial difficulties of a user settling in to the interface, by determining what the user's preferences are, and then somehow suggest alternative modes of interaction or modes not yet explored by the user. Designing and implementing adaptive interfaces seems like a very hard goal to achieve, but we do not believe it is impossible.

The chief purpose of this pilot study was to mature and validate the Wizard-of-Oz techniques that are planned for a forthcoming formal study on human-robot interaction. The result of this pilot study was a number of design improvements made and lessons learned. Critical hardware and software problems were identified. Some were successfully addressed, such as interfacing the various hardware components and writing the software to get the displays up and running on the touch screen. However, others, such as providing bearing information to the map view so that subjects have a better understanding of the robot's position in that view, will require additional work. Iterative evaluation of the whole process between subjects allowed us to catch numerous omissions, and to refine the consistency of the Wizards' interactive behaviors and the overall conduct of the experiment. Careful consideration of the preparation and exit

processes also allowed us to improve the subject training phase, but problems still abound. We were also able to develop a more thorough exit questionnaire. All of this work will ensure the integrity of the formal study.

VI. Concluding Remarks

In this paper, we showed how an evidence grid map is processed so that linguistic expressions can be generated to describe the environment with respect to a robot for robot-to-human feedback. We also showed how spatial terms can be used in robot directives, such as modeling unoccupied space to allow for human-to-robot commands like "Go to the right of the object." We illustrated how spatial language can be integrated into a multi-modal robot interface to provide capabilities for a natural, human-robot dialog. As a result of our experience using spatial language in human-robot dialogs, we also discussed semantic representation for spatial and locative information in a dialog. The work thus far illustrates further questions that need to be addressed, e.g., what is the most useful spatial language needed for a dialog; and what is the best frame of reference for different types of tasks.

In the future, we intend to address these problems. We also want to explore the use of spatial information in robot behaviors, to facilitate additional commands with respect to objects in the environment, e.g., "Move forward until the pillar is behind you". This continued work in supporting and developing spatial language contributes to the natural, multi-modal human-robot interface.

Finally, observations from a Wizard-of-Oz pilot study involving human users directing a mobile robot to find a hidden object have given us pause. Providing users with a variety of natural modes of interaction may be important, but users need to be aware of both the interface's and the robot's capabilities. Perhaps the best way to present these to the user is through an adaptive interface.

Acknowledgements

This research has been supported by ONR and DARPA. The authors would also like to acknowledge the help of Scott Thomas and Myriam Abramson at NRL and Dr. Pascal Matsakis and. Dr. Jim Keller at UMC.

References

[1] F.H. Previc, "The Neuropsychology of 3-D Space", Psychological Review, 1998, vol. 124, no. 2, pp. 123-164.

[2] C. Schunn, T. Harrison, (2001, June) Personal communication, Available through email to: skubicm@missouri.edu

[3] M. Skubic, G. Chronis, P. Matsakis and J. Keller, "Generating Linguistic Spatial Descriptions from Sonar Readings Using the Histogram of Forces", in *Proc. of the 2001 IEEE Intl. Conf. on Robotics and Automation*, Seoul, Korea, May, 2001, pp. 485-490.

[4] M. Skubic, P. Matsakis, B. Forrester and G. Chronis, "Extracting Navigation States from a Hand-Drawn Map", in *Proc. of the 2001 IEEE Intl. Conf. on Robotics and Automation* , Seoul, Korea, May, 2001, pp. 259-264.

[5] M. Skubic, G. Chronis, P. Matsakis and J. Keller. "Spatial Relations for Tactical Robot Navigation", in *Proc. of the SPIE, Unmanned Ground Vehicle Technology III*, Orlando, FL April, 2001.

[6] D. Perzanowski, A.C. Schultz, W. Adams, E. Marsh, M. Bugajska, "Building a Multimodal Human-Robot Interface", *IEEE Intelligent Systems*, pp. 16-20, Jan./Feb, 2001.

[7] W. Adams, D. Perzanowski, A.C. Schultz, "Learning, Storage and Use of Spatial Information in a Robotics Domain", *Proc. of the ICML 2000 Workshop on Machine Learning of Spatial Language*, Stanford Univ.: AAAI Press, 2000, pp. 23-27.

[8] M.C. Martin, H.P. Moravec, "Robot Evidence Grids", Carnegie Mellon University, Pittsburgh, PA, Technical Report #CMU-RI-TR-96-06, Mar., 1996.

[9] D. Perzanowski, A.C. Schultz, W. Adams, M. Bugajska, E. Marsh, G. Trafton, D. Brock, M. Skubic, and M. Abramson, *Multi-Robot Systems: From Swarms to Intelligent Automata: Proceedings from the 2002 NRL Workshop on Multi-Robot Systems*, eds. A.C. Schultz, A.C., and L.E. Parker. Kluwer: Dordrecht, The Netherlands, 2002, pp. 185-193.

[10] R.D. Bolt, "Put-that-there: voice and gesture at the graphics interface". *Computer Graphics*, vol. 14, no. 3, 1980, pp. 262-270.

[11] T.W. Fong, F. Conti, S. Grange and C. Baur, "Novel Interfaces for Remote Driving: Gesture, haptic, and PDA", *SPIE 4195-33, SPIE Telemanipulator and Telepresence Technologies VII*, Boston, MA, November 2000.

[12] D. Kortenkamp, E. Huber and P. Bonasso, "Recognizing and Interpreting Gestures on a Mobile Robot", *Proceedings of AAAI*, 1996.

[13] C. Rich, C.L. Sidner, and N. Lesh, "COLLAGEN: Applying collaborative discourse theory to human-computer interaction", *AI Magazine* vol. 22, no. 4, pp. 15-25, 2001.

[14] J.F. Allen, D.K. Byron, M, Dzikovska, G. Ferguson, L. Galescu and A. Stent, "Toward conversational human-computer interaction, *AI Magazine* vol. 22, no. 4: 27-37, 2001.

[15] K. Wauchope, *Eucalyptus: Integrating Natural Language Input with a Graphical User Interface*, Naval Research Laboratory, Washington, DC, Technical Report NRL/FR/5510-94-9711, 2000.

[16] D. Perzanowski, A. Schultz, W. Adams, and E. Marsh, "Goal Tracking in a Natural Language Interface: Towards Achieving Adjustable Autonomy," *Proc. of the 1999 IEEE Intl. Symp. on Computational Intelligence in Robotics and Automation*, Monterey, CA, 1999, pp.208-213.

[17] M. Skubic, D. Perzanowski, A. Schultz, and W. Adams, "Using Spatial Language in a Human-Robot Dialog," in *Proceedings of the IEEE 2002 International Conference on Robotics and Automation*, Washington, D.C., May, 2002.

[18] M. Skubic, P. Matsakis, G. Chronis, and J. Keller, "Generating Multi-Level Linguistic Spatial Descriptions from Range Sensor Readings Using the Histogram of Forces," Accepted conditionally to *Autonomous Robots.*

[19] A. Schultz, W. Adams and B. Yamauchi, "Integrating Exploration, Localization, Navigation and Planning with a Common Representation," *Autonomous Robots*, vol.6, no.3, May 1999.

[20] P. Matsakis and L. Wendling, "A New Way to Represent the Relative Position between Areal Objects", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 21, no. 7, pp. 634-643, 1999.

[21] J. Keller, P. Matsakis, and M. Skubic, "Beyond 2001: The Linguistic Spatial Odyssey", to appear as a book chapter in *Computational Intelligence Beyond 2001: Real and Imagined*, C. Robinson, ed, Wiley, 2002. Also presented by J. Keller as a Plenary Address, World Congress on Computational Intelligence, Honolulu, Hawaii, May, 2002.

[22] M. Skubic, S. Blisard, and P. Mataskis, "Analyzing Sketched Route Maps for Robot Navigation," Submitted to *IEEE Transactions on SMC, Part B.*

[23] D. Perzanowski , M. Skubic, A. Schultz, W. Adams, M. Bugajska, K. Wauchope, and E. Marsh, "Multi-Modal Navigation of Robots Using Spatial Relations: A Videotaped Demonstration", *Video Proceedings, IEEE 2002 International Conference on Robotics and Automation*, Washington, D.C., May, 2002

[24] M. Kracht, "On the Semantics of Locatives," *Linguistics and Philosophy vol.* 25, pp. 157-232, 2002.

[1] Presently, we use a structured light rangefinder tuned to a laser's wavelength. This is our sole source for gesture information. We are not using binocular vision, although we hope to advance to this more robust capability shortly.

[2] For expositional purposes, we include the logical representation (D) here, although it is not used in checking gestures *per se*. On the other hand, it is used for further linguistic analysis where necessary, such as pronominal dereferencing and quantifier scoping. Given (D), therefore, it is possible to interpret such utterances as "Go to the left of it," where *it* is analyzed as a pronoun in a larger discourse, and "How many objects do you see?" where it is necessary to process the number of objects.

[3] Currently our vision system does not permit fine distinctions such as triangulating on a point indicated by the human gesture. Consequently, we simply pass information to the robot module indicating that the robot should move in some general direction indicated by the gesture. The use of the "obstacle" element in the string simply informs the system of the termination of the command. In future, we hope to incorporate a more robust vision system where triangulation is possible.

[4] For expositional purposes here, the representation of the context predicate is somewhat simplified.