Transparency in Robot Autonomy

Manuela M. Veloso

Joint work Francesca Rossi, Stephanie Rosenthal, Sai Selvaraj, Vittorio Perera

Machine Learning Department
Computer Science Department
Robotics Institute
School of Computer Science
Carnegie Mellon University

Workshop – Beneficial AI, January 5, 2017

Mobile Service Robots: CoBots
(PhD Theses at www.cs.cmu.edu/~mmv)

Navigation: > 1,200kms

Stephanie Rosenthal, 2012
Human-Centered Planning

Brian Coltin, 2014
Multi-Robot Pick up and Deliver Task Scheduling with Transfers

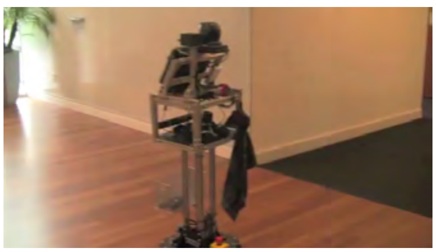
Joydeep Biswas, 2014
Vector-Based Episodic Non-Markov Localization

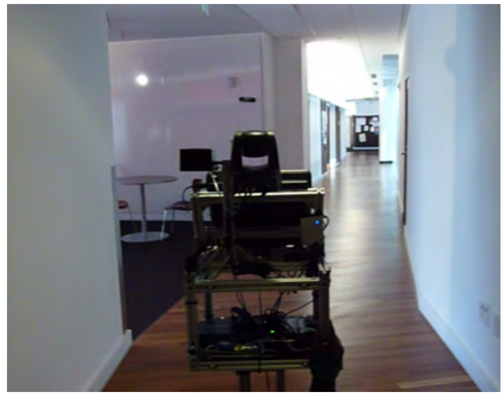
Richard Wang, 2016
Collecting, Analyzing, and Using Fine-Grain Sensor Data with Mobile Platforms

Vittorio Perera, ongoing
Learning from Human-Robot Interaction through Language

Autonomous Mobile Service Robots

- ***Tasks***
  - Go to a location
  - Deliver a message
  - Transport object between locations
  - Escort a visitor

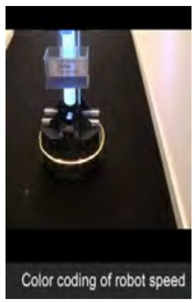




Transparency in Autonomy

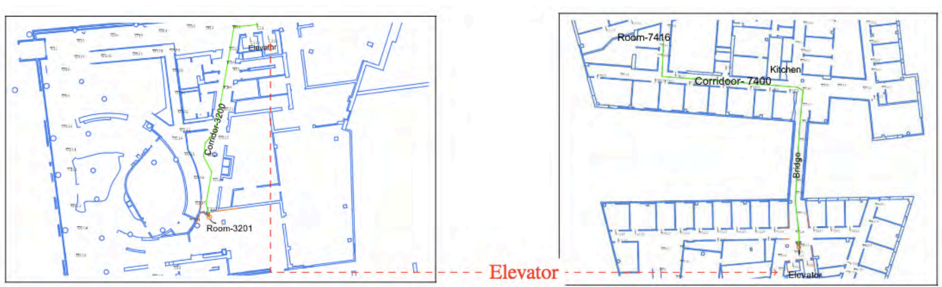
- What are you going to do next?
- What is your internal state?
- Which path did you take?
- What happened by the elevator?
- How long did it take to arrive here?
- Did you successfully escort Eric?
- Why are you late?
- …

Explanation through Expressive Lights



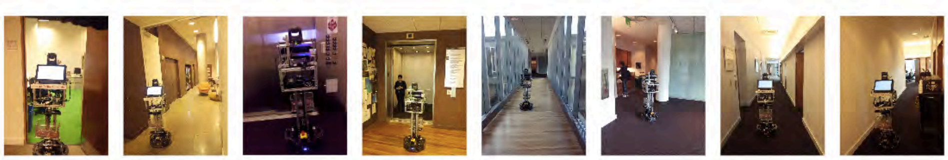
Kim Baraka

Experience – Sequence of Actions



3rd Floor map

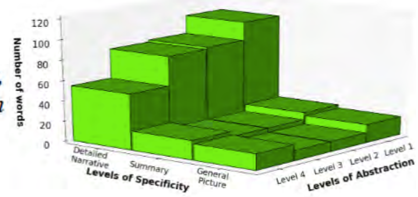
7th Floor map



Generating Variable Level of Explanation

I traveled 26 meters and took 152 seconds on the 7^{th} floor.

I went Straight for 15 meters and Turned Left, then Straight for 51 meters and Turned Left, then Straight for 10 meters to reach the destination.



I started from room 3201, I went through the 3200 corridor, then I took the elevator and went to the seventh floor, then I took the 7th floor bridge, then I passed the kitchen, then I went through the 7400 corridor, then I reached room 7416.

Verbalization and Verbalization Space

Verbalization: the process by which an autonomous robot converts its own experience into language

Verbalization space: to capture different nature of explanations **Specificity, locality, abstraction**

		Abstraction, A			
		Level 1	Level 2	Level 3	Level 4
Specificity, S	General Picture	Start point, end point and average speed for the complete L	Total distance and time taken for the complete L	Total distance and time taken for all new environments in L	Starting and ending landmark for the complete L
	Summary	Start and end point, and average speed for all new environments in L	Total distance and time taken for all new environments in L	Total distance and time taken for all new environments in L	Starting and ending landmark for all new environments in L
	Detailed Narrative	Start and end point, and time taken between neighboring p's in L	Angle turned at the N_I, total distance and time taken between neighboring p's in L	Directions at each N_I required to reach the next p in L, along with distance to move	Starting and ending landmark between neighboring p's in L

S. Rosenthal, S. P. Selvaraj, and M. Veloso, "Verbalization: Narration of Autonomous Mobile Robot Experience", Proceedings of IJCAI'16.

Papers as Attached

- CoBots: Robust Symbiotic Autonomous Mobile Service Robots, Manuela Veloso, Joydeep Biswas, Brian Coltin, and Stephanie Rosenthal, In *Proceedings of IJCAI'15.*
- Verbalization: Narration of Autonomous Mobile Robot Experience, Stephanie Rosenthal, Sai P. Selvaraj, and Manuela Veloso In *Proceedings of IJCAI'16.*
- Dynamic Generation and Refinement of Robot Verbalization, Vittorio Perera, Sai P. Selvaraj, Stephanie Rosenthal, and Manuela Veloso. In *Proceedings of RO-MAN'16.*
- Enhancing Human Understanding of a Mobile Robot's State and Actions using Expressive Lights, Kim Baraka, Stephanie Rosenthal, and Manuela Veloso. In *Proceedings of RO-MAN'16.*

Conclusion – Autonomous Robots
www.cs.cmu.edu/~mmv/Veloso.html

- ***Peer*** **interaction with humans**
  - **Symbiotic autonomy**
  - **Learning from humans**
    - **Environment groundings**
  - **Transparency**
    - **Automatic annotation of videos**
    - **Expressive lights**
    - **Verbalization**
- **Current work**
  - **Learning explanations in Deep learning and DeepRL**

Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI 2015)

CoBots: Robust Symbiotic Autonomous Mobile Service Robots

Manuela Veloso, Joydeep Biswas, Brian Coltin, and Stephanie Rosenthal
School of Computer Science
Carnegie Mellon University
mmv@cs.cmu.edu, joydeepb@ri.cmu.edu, brian@coltin.org, steph.rosenthal@gmail.com

Abstract

We research and develop autonomous mobile service robots as Collaborative Robots, i.e., CoBots. For the last three years, our four CoBots have autonomously navigated in our multi-floor office buildings for more than 1,000km, as the result of the integration of multiple perceptual, cognitive, and actuations representations and algorithms. In this paper, we identify a few core aspects of our CoBots underlying their robust functionality. The reliable mobility in the varying indoor environments comes from a novel episodic non-Markov localization. Service tasks requested by users are the input to a scheduler that can consider different types of constraints, including transfers among multiple robots. With symbiotic autonomy, the CoBots proactively seek external sources of help to fill-in for their inevitable occasional limitations. We present sampled results from a deployment and conclude with a brief review of other features of our service robots.

1 Introduction

We research, develop, and deploy multiple autonomous mobile robots capable of performing tasks requested by users in our multi-floor office building. To successfuly perform service tasks, our robots have several core capabilities, namely:

- To autonomously localize and navigate in the diverse types of indoor space, including corridors, elevators, and open areas with movable furniture and people.
- To schedule conflict-free plans for multiple robots to satisfy constrained tasks specified and requested by users.
- To overcome the robots' own limitations, in particular in actuation, by proactively ask for help from humans.

Our current four Collaborative Robots, CoBot-1 through CoBot-4 (see Figure 1) can be viewed as mobile, computing, and sensing platforms, that behave as service robots.[1]

Mobile robots, by definition, need to be able to move, in our case, in indoor environments. Such capability has been extensively investigated. The fact that our research goal includes the persistent deployment of the CoBots led to the introduction of novel mapping, sensing, and localization approaches [Biswas, 2014]. The robots classify sensed obstacles as map-known long-term features (walls) or map-missing short-term (furniture) and dynamic (people) features. This explicit distinction enables the overall effective episodic non-Markovian localization approach [Biswas *et al.*, 2011; Biswas and Veloso, 2012; 2013].

Figure 1: CoBots with omnidirectional motion, onboard computation, interaction interfaces, carrying baskets, and different combinations of depth sensing (cameras and LIDAR).

Service robots need to be able to perform service tasks. The CoBot robots can perform multiple classes of tasks, as requested by users through a website [Ventura *et al.*, 2013], in person through speech [Kollar *et al.*, 2013] or on the robot's touch screen. All tasks can be represented as pick up and delivery tasks of objects or people. The task scheduler takes into account time and location constraints, as well as the multiple available robots, and issues plans that can include transfers [Coltin, 2014].

As can be seen in Figure 1, a CoBot has no hands but has a basket, so it can carry, but not manipulate items. To overcome this actuation limitation, and inevitably other types of limitations, the robots proactively ask for help from humans [Rosenthal *et al.*, 2010; 2011], and from the web [Samadi *et al.*, 2012]. They can gather and use models of human help and preferences in a human-centered planning approach [Rosenthal, 2012].

[1]Thanks to Mike Licitra for designing and building the CoBots, and to Joydeep Biswas for keeping them functional.

2 Episodic non-Markov Localization

A variety of early robots, such as Shakey [Nilsson, 1984]), Xavier [Simmons *et al.*, 1997], and museum tour guide robots [Burgard *et al.*, 1999; Fong *et al.*, 2003], and more recent ones [Chen *et al.*, 2012; Randelli *et al.*, 2013; Christensen *et al.*, 2010; Hawes *et al.*, 2007; Dias and Ventura, 2013; Zhang and Stone, 2015; Visser and Burkhard, 2007]. All of these efforts include variations of localization algorithms [Dellaert *et al.*, 1999]. Our CoBot robots, as deployed in a multi-floor university building setting, now for more than 1,000km, have faced new challenges.

Over the course of their regular deployments, the CoBots are exposed to a variety of types of environments. Some environments like corridors remain largely invariant over time, with little or no changes. Other environments like cafe areas and open atria, exhibit significant changes over time, with objects like tables and chairs being moved around frequently, and numerous dynamic obstacles like humans. Such environments pose a challenge to localization algorithms that assume that the world can be represented by a static map.

To localize in the presence of frequently observed movable and moving objects, we introduce Episodic non-Markov Localization [Biswas and Veloso, 2014] that explicitly reasons about observations of non-mapped objects without saving locally static maps. Episodic non-Markov localization maintains a belief of the history of pose estimates of the robot over "episodes" of observations of unmapped objects. For every time-step, it classifies observations into those arising from the map ("Long Term Features", LTFs), from unmapped static objects ("Short Term Features", STFs), or from moving objects ("Dynamic Features", DFs). The correlations between poses of the robot due to the presence of STFs and DFs are represented by a "Varying Graphical Network" (VGN), which we introduce next.

2.1 The Varying Graphical Network

As in a Dynamic Bayesian Network, a VGN includes certain periodically repeating nodes and edges that do not change with the belief. We term these the non-varying nodes and edges. A VGN includes two additional structural elements: varying nodes and varying edges. The presence and structure of the varying nodes and varying edges are not known a-priori, and are estimated jointly with the belief. Since the estimates of the structure may change with the belief, the structure is likely to change as new observations become available.

VGNs provide an accurate representation for non-Markov localization. The presence of LTFs and their relations to the map, and the correlations between successive poses of the robot due to odometry observations are encoded by the non-varying edges and nodes. The presence of STFs and DFs is encoded by the presence of associated varying nodes. The correlations between STFs observed at different time-steps is encoded by the varying edges. The Belief of the robot's localization, $Bel(x_{1:n})$ is maintained over a history of n poses $x_{1:n}$. For each timestep i, odometry u_i corresponds to the robot's relative motion between poses x_{i-1} and x_i, and observation s_i, made at pose x_i, includes observations of LTFs that match the map, as well as unexpected observations of STFs and DFs.

Since the VGN for non-Markov localization has no predefined structure, it might seem that the computation of the belief would require storing the complete history of all states and observations since the robot was turned on. However, in practice this is not necessary, as we rely on the existence of "episodes" in non-Markov localization. Suppose there exists a time step t_i such that all observations and state estimates made after t_i, given x_i, are independent of all prior observations and state estimates:

$$P(x_{1:n}|x_0, s_{1:n}, u_{1:n}, M) = \\ P(x_{1:i}|x_0, s_{1:i}, u_{1:i}, M) \times P(x_{i+1:n}|x_i, s_{i+1:n}, u_{i+1:n}, M). \quad (1)$$

This conditional independence implies that there are no STF observations after t_i that correspond to STF observations before t_i. In such a case, the history of states and observations prior to t_i, called the "episode" $t_{0:i-1}$, can be discarded when estimating $Bel(x_{i:n})$ over the episode $t_{i:n}$. We assume such episode-boundary time-steps like t_i exist, allowing real-time non-Markov localization with limited computational resources. Figure 2 shows an example VGN near an episode boundary, highlighting the absence of any varying edges crossing the episode boundary.

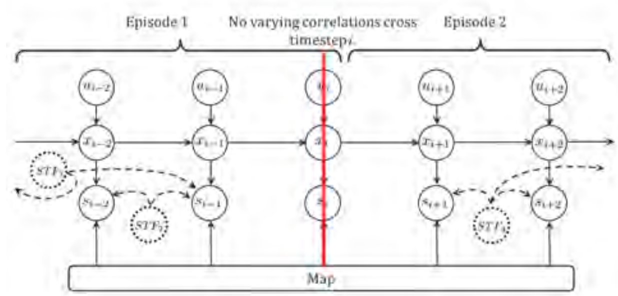


Figure 2: An example VGN demonstrating the presence of an episode in non-Markov localization. Note the absence of any varying edges that cross the red line indicating the episode boundary. Hence the pose x_i is an episode boundary, where all previous poses up to x_{i-1} are in the previous episode, and poses x_i and later are in the latest episode. Observations s_{i-1} and older thus no longer need to be stored.

The exact structure of the VGN depends on the specific LTFs and STFs that are observed by the robot, and we next present how the observations are classified.

2.2 Classification of Long-Term and Short-Term Features

For every time-step, the structure of the VGN, based on the classification of the observations into LTFs, STFs and DFs, is re-evaluated prior to updating the MLE of the belief. In this work the sensor we use is a laser rangefinder, so each observation s_i is a set of n_i 2D points $s_i = \{p^i_j\}_{j=1:n_i}$ observed by the robot. We represent the pose x_i of the robot on the map at time-step i as an affine transform T_i that consists of a 2D rotation followed by a 2D translation.

We use a vector map [Biswas *et al.*, 2011] representation $M_{\text{vector}} = \{l_i\}_{1=1:s}$ for the permanent map, consisting of a set of s line segments l_i. To evaluate which of the observed points p_k^i are LTFs, an analytic ray cast [Biswas and Veloso, 2012] is performed from the latest MLE of x_i. The result of the analytic ray cast is a mapping from $p_j^i \rightarrow l_j \in M_{\text{vector}}$, indicating that the line segment l_j from map M_{vector} is the most likely line in the map to be observed by the point p_j^i. Let $\mathrm{d}ist(p,l)$ denote the perpendicular distance of point p from the line segment l where both p and l are in the reference frame of the map. The observation likelihood $P(p_j^i|T_i, M_{\text{vector}})$ of the point p_j^i is then given by

$$P(p_j^i|x_i, M_{\text{vector}}) = \exp\left(-\frac{\mathrm{d}ist(T_i p_j^i, l_j)^2}{\Sigma_s}\right), \quad (2)$$

where Σ_s is the scalar variance of observations, which depends on the accuracy of the sensor used. Thus, observations are classified as LTFs if the observation likelihood of the point given the map is greater than a threshold, $P(p_j^i|x_i, M_{\text{vector}}) > \epsilon_{\text{LTF}}$.

Observed points that are classified as non-LTFs could potentially be STFs. To check if an observed point $p_j^i \in \overline{\text{LTF}}_i$ is an STF, it is compared to all non-LTF points observed prior to time-step i to check if they correspond to observations of the same point. Given a point $p_j^i \in \overline{\text{LTF}}_i$ observed at time-step i and another point $p_k^l \in \overline{\text{LTF}}_l$ observed at a previous time-step l, the probability that both the observations correspond to the same point is given by the STF observation likelihood function,

$$P(p_j^i, p_k^l|x_i, x_l) = \exp\left(-\frac{||T_i p_j^i - T_l p_k^l||^2}{\Sigma_s}\right). \quad (3)$$

Therefore, a non-LTF point $p_j^i \in \overline{\text{LTF}}_i$ is classified as an STF if there exists a point $p_k^l \in \overline{\text{LTF}}_l$ from a time-step $l, l < i$ such that $P(p_j^i, p_k^l|x_i, x_l) > \epsilon_{\text{STF}}$.

Given the classifications, and the form of the observation likelihoods of the LTFs and STFs, episodic non-Markov localization solves for the maximum likelihood estimate of the belief by representing the Belief as a cost function and optimizing over it, instead of keeping multiple estimates represented as a particle filter [Biswas, 2014].

We convert the belief from a probability distribution representation to a cost function representation C such that

$$\begin{aligned} Bel(x_{1:n}) &= P(x_{1:n}|x_0, s_{1:n}, u_{1:n}, M) \\ &\propto \exp(-C(x_{1:n}|x_0, s_{1:n}, u_{1:n}, M)). \quad (4) \end{aligned}$$

The cost function C consists of a sum of m sub-cost functions c_j^{STF} corresponding to the STF terms $P(s_{1:n}^{\text{STF}_j}|x_{1:n})$, n sub-cost functions c_i^{LTF} corresponding to the LTF terms $P(s_i^{\text{LTF}}|x_i, M)$, and n sub-cost functions c_i^{odom} corresponding to the odometry terms $P(x_i|x_{i-1}, u_i)$.

The Maximum Likelihood Estimate $x_{1:n}^*$ is therefore computed by minimizing the cost function as:

$$x_{1:n}^* = \arg\min_{x_{1:n}} \left(C(x_{1:n}|x_0, s_{1:n}, u_{1:n}, M)\right). \quad (5)$$

Thus, Episodic non-Markov Localization updates the maximum likelihood location estimates of the robot via functional non-linear least squares optimization of Equation 5.

2.3 Results

Episodic non-Markov Localization has been deployed on all the CoBots over part of a 1,000km Challenge [Biswas, 2014], and has been used to localize the robots in many different environments spanning multiple floors across multiple buildings. In particular, it has been instrumental in increasing the robustness of localization on floors with challenging open areas, like a large atrium on the floor GHC4. Figure 3 illustrates different placements of the STFs, namely movable furniture.

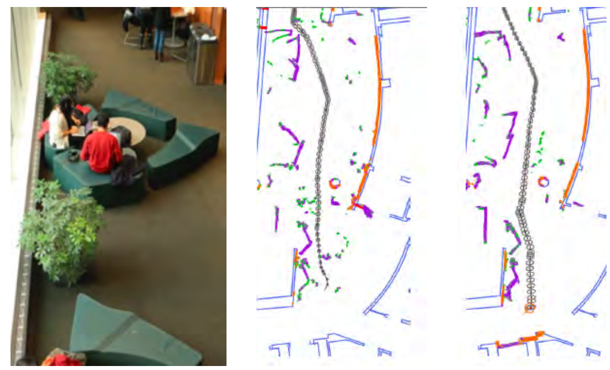


Figure 3: View of the challenging varying space in GHC4 atrium and snapshots of enML at two different times. The trajectory of the robot over the episode is shown in grey, along with the covariance elipses. LTF observations are shown as orange points, STF observations as purple points, and DF observations as green points. The long-term static map is shown as blue lines.

To highlight the contribution of the robustness of Episodic non-Markov localization to the deployments of the CoBots, we tabulate the mean distance traversed autonomously by the CoBots between operator interventions in Table 1.

	CGR	EnML
GHC4	0.62	4.42
GHC6	8.61	9.48
GHC7	5.58	9.02
GHC8	6.04	19.36
GHC9	5.33	20.05
NSH4	0.56	2.65
All	4.79	8.13

Table 1: Mean distance (in km) traversed between interventions using CGR (a variant of Markov Localization) and EnML per map over the 1,000km Challenge.

The CoBots traversed a mean distance of $4.42km$ between interventions while using EnML for localization as opposed to $0.62km$ when using Corrective Gradient Refinement (CGR) [Biswas *et al.*, 2011], a variant of Markov Localization. Overall, EnML allowed the CoBots to traverse a mean of $8.13km$ as opposed to $4.79km$ when using CGR. The increased robusteness is attributed to the ability of EnML to reason better about observations of unmapped objects, and hence its robustness to changes in the environment.

3 Scheduling for Transfers with CoBots

Tasks requested by users are processed by a *scheduler* that computes an ordered assignment of tasks to the multiple robots. The scheduler needs to satisfy various constraints stated by the users, including location, time windows, transportation capacities of the robots, and maximum delivery times. The goal of scheduler is to find a valid schedule which minimize the total distance traveled by the robots and, or the completion times of the tasks. The scheduler outputs task execution times for each robot, and sends lists of tasks to the robots. During execution, the robots update the scheduler of their progress.

We realized that the robots can perform their tasks more efficiently by *transferring* items between one another [Coltin, 2014]. For example, the scheduler, without considering transfers, could assign CoBot-1 and CoBot-2 both to pick up items on the seventh floor that they need to deliver to the ninth floor. Instead of both taking an elevator ride, CoBot-1 could transfer its item to CoBot-2, which could deliver both items.

Initially, we introduce a scheduler that generates an optimal schedule for the CoBots using mixed integer programming (MIP) [Coltin, 2014]. Finding the optimal schedule is NP-hard, so the MIP solver scales poorly, although for our typical usage of less than fifteen tasks at once solving the problem optimally is feasible.

To scale to larger problems, we developed an approximation algorithm for a variant of the scheduling problem in which all the items share the same destination and there are no time constraints. This is a common scenario for the CoBots, when they pick up mail for delivery to the central office, or hand out candy to building occupants for Halloween. The approximation algorithm is based on an approximation for the traveling salesman problem, and returns a solution that is guaranteed to be within a factor of two of optimal in terms of total distance traveled [Coltin and Veloso, 2014a].

Expanding to the more general problem in which items have distinct destinations, we introduced three heuristics to from schedules, still without considering time: a greedy approach, an algorithm based on auctions, and an algorithm where an item's entire trajectory is inserted into a graph of transfers. The heuristics reduce the search space by inserting transfers into existing schedules, and hence may not find the optimal solution. Transfers were shown to reduce the solution cost compared to similar heuristics without transfers [Coltin and Veloso, 2014b].

We extended the auction heuristic to work with time windows, by determining execution times through the use of simple temporal networks. The auction algorithm is applied online as new tasks come in from users, so that the CoBots replan online. If a CoBot is delayed or disabled, the other CoBots replan so that the tasks are still completed as quickly as possible (see Figure 4).

The CoBots also take advantage of the fact that there are multiple robots to replan better schedules. If a robot is blocked in a hallway, it will inform the other robots it is blocked. The other robots will then replan to avoid the blocked hallway, if possible, as shown in Figure 5. Additionally, robots detect if doors are open or closed when they drive past. If one robot happens to drive by a closed door that another robot is planning to pick up or deliver an item from, it will tell the other robot, and the scheduler will attempt to delay the task at that room until a later time when the occupant has hopefully returned to their office [Coltin and Veloso, 2013]. In general, each robot can be aware of and check the *rationale* of the plans of other robots.

Figure 4: (a) Deliveries are scheduled with three robots, including two transfers; (b) When one of the robots fails, the tasks are rescheduled. Squares indicate pickups, circles indicate deliveries, triangles indicate transfers, and stars indicate robot starting points. The numbers inside indicate either robot and item numbers.

Finally, we introduce an algorithm based on simulated annealing which finds high quality non-optimal schedules with transfers. While more computationally expensive than the previous heuristics, this algorithm outperforms the best previous solutions to benchmark scheduling problems by incorporating transfers. In addition to the CoBots, the idea of transfers are also applied to transportation and ridesharing problems to reduce fuel costs [Coltin and Veloso, 2014c].

Multiple CoBots continue to autonomously perform tasks in the Gates-Hillman building. The scheduling algorithms, with transfers, allow the CoBots to complete more tasks, more quickly, while prolonging battery life.

4 Human-Centered Planning for Symbiotic Autonomy

Rather than limiting robots' tasks to those that only include actions that robots can perform autonomously, CoBot instead reasons about, plans for, and overcomes its limitations by proactively asking humans in the environment for help [Rosenthal *et al.*, 2010].

We introduced a human-centered planning algorithm that asks for help when CoBot is uncertain of its location or when it is uncertain of which action to take [Rosenthal *et al.*, 2010]. Robots and humans are in a symbiotic relationship, as robots perform service tasks for humans, and humans may need to help the robots. The underlying assumption for the symbiotic robot autonomy is that the requests for help from the robot, e.g., pressing an elevator button, are simple for humans.

The symbiotic autonomy approach leads to adding *ask-for-help* action primitives to the robots' plans. The robots autonomously perform such actions. Figure 6 shows a high-level partial conditional plan for the robot to navigate to a

(a) CoBot-2 to Office (b) CoBot-4 Blocked (c) CoBot-2 Replans (d) Obstacle Avoided

Figure 5: (a) CoBot-2 heads towards an office to make a delivery, and shares with CoBot-4 the path that it needs to traverse; (b) CoBot-4 detects that a hallway of relevance to CoBot-2's path is blocked; (c) The scheduler replans for CoBot-2; and (d) CoBot-2 takes an alternate round to avoid the blocked hallway.

room, where it asks for help from a human to push the elevator buttons.

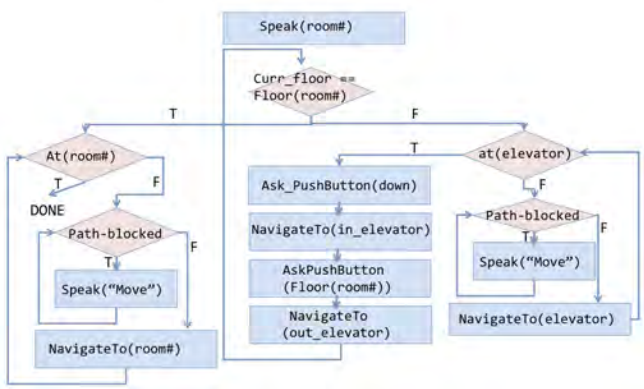


Figure 6: High-level partial conditional plan for symbiotic autonomy to navigate with actuation limitations and asking for help if needing to take the elevator.

The conditional plan in Figure 7 is partial in the sense that it does not include what happens if any of the actions fails. We have developed several approaches to handle the additional contingencies of symbiotic autonomy, namely (i) no human helps, e.g., the robot keeps waiting by the elevator; (ii) a human provides the wrong help, e.g., it tells the robot is on an incorrect floor. In the latter situation, as soon as the robot recognizes that it is not at its desired location, it continues its execution by replanning or recognizing that it cannot perform its task for any reason. In both situations, the robot is unable to proceed its task. We have developed two approaches: timeout-based one and a proactive-seek for help one. In the timeout-based approach, the robot waits for help for a predefined amount of time, after which it sends email to its developers using a template that it fills in describing the location and situation where it finds stalled. This step represents an action to ask for help from remote humans.

In the proactive-seek for help approach, we studied who, whether, and where to *proactively* ask building occupants for help, concretely to use the elevator. We made five hypotheses based on our intuitions about what human state attributes matter in determining where and who to ask for help. The first two hypotheses represent the spatial considerations that CoBot robot should take into account.

- Cost of Help: Asking someone for help who is already at the elevator is preferred over finding someone in an office. A benefit of asking the elevator person is that they are already performing the action themselves and should have little cost to helping the robot
- Distance to Help Location: If someone in an office must be asked because it is unlikely that anyone will be at the help location, there should be a preference for asking someone close to the location to avoid making someone travel too far. Although CoBot is mobile and are capable of traveling to find help, an in-office helper would have to travel back to the help location.

The second three hypotheses represent the considerations the robot should make to increase the likelihood that people are willing to comply and help the robot, because the robot need help performing these actions over a long period of time.

- Interruption: The robot should avoid requesting help from people in offices that are likely to be busy.
- Recency of Last Question and Frequency of Questions: The robot should take into account how recently it asked different helpers to avoid asking too often.
- Availability: If a robot travels with a person to the help location and there is someone already at the location of help, the traveling person may feel that they were asked unnecessarily.

Through user studies, we confirmed all five hypotheses [Rosenthal and Veloso, 2012]. Robots should consider the cost of help, distance to help location, availability, interruptibility, and frequency and recency of questions. However,

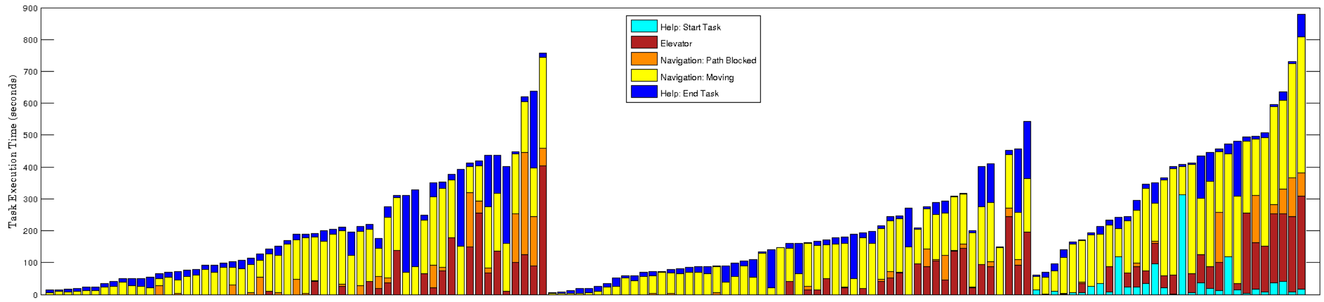


Figure 7: Execution times for, from left to right, Deliver Message tasks, Go to Room tasks, and Transport tasks. The breakdown includes 1) waiting for help to start the task, 2) riding the elevator, 3) navigating (not including time blocked by obstacles), 4) waiting blocked by an obstacle, and 5) waiting for help to end the task.

some participants were willing to help irrespective of the distance to the help location. We use these human state attributes in our human-centered planning algorithm to determine who to ask for help and where to navigate.

When CoBot needs help to use the elevator, it first asks at the elevator hall if anyone is available to help. The people at the elevator hall location have the lowest cost of helping the robot because they are at the elevator anyway. However, if no one helps CoBot, it plans where to seek for help by computing the decision-theoretic expected cost of asking a person in their office based on our user study findings to come to help the robot to get to the desired floor.

The goal of our human-centered proactive replanning algorithm is to simultaneously reduce the time to complete the task while also limiting the in-office help [Rosenthal and Veloso, 2012]. We were able to show that CoBot could complete tasks 4mn faster on average with the proactive replanning algorithm compared to waiting at the elevator only.

The CoBot robots have been deployed for more than 1,000km, in our multi-floor buildings successfully navigating, using their symbiotic autonomy, in particular to move between floors. We present a sample of the results to illustrate the impact of the symbiotic autonomy in the timing of the tasks. The results correspond to a deployment of one robot on the upper four floors of our office building for a two week period. CoBot was deployed for two hours every weekday and made available to the building occupants.

The response to CoBot's deployment was positive: over 100 building occupants registered to use CoBot. Users found creative ways to exploit the robot's capabilities, including, but not limited to sending messages to friends, reminding occupants of meetings, escorting visitors between offices, delivering printouts, inter-office mail, USB sticks, snacks, owed money, and beverages to other building occupants.

We found that occupants scheduled the robot to transport objects between multiple floors of the building more often than they used the multi-floor functionality for other tasks (see Table 2). In particular, the transport task saved the task solicitors time because they did not have to travel between floors themselves. However, even the other scheduled tasks utilized the elevator 40% of the time.

Figure 7 shows how much time CoBot took to execute each task, and how that time was apportioned. A total of 140 tasks were completed during the two-week deployment, which took 9 hours and 13 minutes. Based on these times, we find that task solicitors quickly responded to the robot's request for help at the start and end of tasks. Building occupants (even those that had never scheduled a task) were willing and able to help the robot in and out of the elevator. This finding supports our model of symbiotic autonomy, namely that humans are willing to help a robot complete its tasks so that the robot is available and capable of performing tasks for them as well at another time.

Table 2: Total number of task requests per task type and the respective number that used the elevator.

Task Type	Total Requests	# Multi-floor
Escort	3	2
GoToRoom	52	22
DeliverMessage	56	20
Transport	29	22

5 Conclusion

The CoBot robots have been successfully deployed in multi-floor buildings for over three years. We summarized some of the core contributions. The episodic non-Markovian localization to effectively handle environments whose depth appearance varies over time. Long-term features, e.g., walls, match existing floor-plan maps, while short-term features, e.g., furniture, match previous observations in an episodic non-Markovian manner. The multi-robot task scheduler considers transfers among robots to optimize the travel time performance, and replans to handle online requests and changing conditions. Symbiotic autonomy enables the robot to ask for help from humans at the place needed or proactively search for help from near-by humans. Human-centered planning uses models of humans to generate robots' plans.

Current and future work include learning to improve service performance, including human-preference and environment learning and exploration. We also continue to research on detection of anomalies for safety of use. We are also focused on task instruction and correction through natural language [Mericli *et al.*, 2014], to enable any user to requests new tasks from the robot.

Acknowledgements

We thank Aijun Bai, Steven Klee, Thomas Kollar, Max Korein, Juan Pablo Mendoza, Cetin Mericli, Daniele Nardi, Vittorio Perera, Mehdi Samadi, Yichao Sun, and Rodrigo Ventura for their contributions to the CoBot robots.

This research is partly sponsored by the NSF under awards NSF IIS-1012733 and IIS-1218932, and the ONR under grant number N00014-09-1-1031. The views and conclusions contained herein are those of the authors only.

References

[Biswas and Veloso, 2012] J. Biswas and M. Veloso. Depth camera based indoor mobile robot localization and navigation. In *Proc. of ICRA*, pages 1697–1702, 2012.

[Biswas and Veloso, 2013] J. Biswas and M. Veloso. Localization of the CoBots over Long-term Deployments. *Int. Journal of Robotics Research*, 32(14):1679–1694, 2013.

[Biswas and Veloso, 2014] J. Biswas and M. Veloso. Episodic non-markov localization: Reasoning about short-term and long-term features. In *Proc. ICRA*, 2014.

[Biswas *et al.*, 2011] J. Biswas, B. Coltin, and M. Veloso. Corrective gradient refinement for mobile robot localization. In *Proceedings of IROS*, pages 73–78, 2011.

[Biswas, 2014] Joydeep Biswas. *Vector Map-Based, Non-Markov Localization for Long-Term Deployment of Autonomous Mobile Robot*. PhD thesis, Carnegie Mellon University, 2014. CMU-RI-14-25.

[Burgard *et al.*, 1999] W. Burgard, A. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 114:3–55, 1999.

[Chen *et al.*, 2012] X. Chen, J. Xie, J. Ji, and Z. Sui. Toward open knowledge enabling for human-robot interaction. *J. of Human-Robot Interaction*, 1-2:100–117, 2012.

[Christensen *et al.*, 2010] H. I. Christensen, G.J. Kruijff, and J. Wyatt, editors. *Cognitive Systems*. COSMOS. Springer Verlag, Berlin, DE, May 2010.

[Coltin and Veloso, 2013] B. Coltin and M. Veloso. Towards replanning for mobile service robots with shared information. In *Proc. of ARMS Workshop, AAMAS*, 2013.

[Coltin and Veloso, 2014a] B. Coltin and M. Veloso. Optimizing for transfers in a multi-vehicle collection and delivery problem. In *Proc. of DARS*, 2014.

[Coltin and Veloso, 2014b] B. Coltin and M. Veloso. Ridesharing with passenger transfers. In *IROS*, 2014.

[Coltin and Veloso, 2014c] B. Coltin and M. Veloso. Scheduling for transfers in pickup and delivery problems with very large neighborhood search. In *AAAI*, 2014.

[Coltin, 2014] Brian Coltin. *Multi-agent Pickup and Delivery Planning with Transfers*. PhD thesis, Carnegie Mellon University, 2014. CMU-RI-14-05.

[Dellaert *et al.*, 1999] F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte carlo localization for mobile robots. In *Proc. of ICRA*, 1999.

[Dias and Ventura, 2013] D. Dias and R. Ventura. Absolute localization for low capability robots in structured environments using barcode landmarks. *J. of Automation, Mobile Robotics and Intelligent Systems*, 7(1):28–34, 2013.

[Fong *et al.*, 2003] T. Fong, I. Nourbakhsh, and K. Dautenhahn. A survey of socially interactive robots. *Robotics and Autonomous Systems*, 42:143–166, 2003.

[Hawes *et al.*, 2007] N. Hawes, A. Sloman, J. Wyatt, M. Zillich, H. Jacobsson, G.-J. Kruijff, M. Brenner, G. Berginc, and D. Skocaj. Towards an Integrated Robot with Multiple Cognitive Functions. In *Proceedings of AAAI*, 2007.

[Kollar *et al.*, 2013] T. Kollar, V. Perera, D. Nardi, and M. Veloso. Learning Environmental Knowledge from Task-Based Human-Robot Dialog. In *Proc. of ICRA*, 2013.

[Mericli *et al.*, 2014] C. Mericli, S. Klee, J. Paparian, and M. Veloso. An Interactive Approach for Situated Task Specification through Verbal Instructions. In *Proc. of AAMAS'*, 2014.

[Nilsson, 1984] N. J. Nilsson. Shakey the robot. Technical Report 323, SRI International, 1984.

[Randelli *et al.*, 2013] G. Randelli, T. Bonanni, L. Iocchi, and D. Nardi. Knowledge acquisition through humanrobot multimodal interaction. *Intelligent Service Robotics*, 6:19–31, 2013.

[Rosenthal and Veloso, 2012] S. Rosenthal and M. Veloso. Mobile Robot Planning to Seek Help with Spatially-Situated Tasks. In *Proc. of AAAI*, 2012.

[Rosenthal *et al.*, 2010] S. Rosenthal, J. Biswas, and M. Veloso. An Effective Personal Mobile Robot Agent Through Symbiotic Human-Robot Interaction. In *Proc. of AAMAS*, 2010.

[Rosenthal *et al.*, 2011] S. Rosenthal, M. Veloso, and A. Dey. Learning Accuracy and Availability of Humans who Help Mobile Robots. In *Proc. of AAAI*, 2011.

[Rosenthal, 2012] Stephanie Rosenthal. *Human-Centered Planning for Effective Task Autonomy*. PhD thesis, Carnegie Mellon University, 2012. CMU-CS-12-110.

[Samadi *et al.*, 2012] Mehdi Samadi, Thomas Kollar, and M. Veloso. Using the Web to Interactively Learn to Find Objects. In *Proc. of AAAI*, 2012.

[Simmons *et al.*, 1997] R. Simmons, R. Goodwin, K. Haigh, S. Koenig, and J. O'Sullivan. A layered architecture for office delivery robots. In *Proc. of AGENTS*, 1997.

[Ventura *et al.*, 2013] R. Ventura, B. Coltin, and M. Veloso. Web-Based Remote Assistance to Overcome Robot Perceptual Limitations. In *AAAI Workshop on "Intelligent Robotic Systems"*, 2013.

[Visser and Burkhard, 2007] U. Visser and H.-D. Burkhard. RoboCup: 10 years of achievements and future challenges. *AI Magazine*, 28(2):115–132, Summer 2007.

[Zhang and Stone, 2015] S. Zhang and P. Stone. CORPP: Commonsense Reasoning and Probabilistic Planning, as Applied to Dialog with a Mobile Robot. In *Proc. of AAAI*, 2015.

Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)

Verbalization: Narration of Autonomous Robot Experience

Stephanie Rosenthal
Software Engineering Institute
srosenthal@sei.cmu.edu

Sai P. Selvaraj
Robotics Institute
spandise@andrew.cmu.edu

Manuela Veloso
Computer Science Department
veloso@cs.cmu.edu

Carnegie Mellon University, Pittsburgh USA

Abstract

Autonomous mobile robots navigate in our spaces by planning and executing routes to destinations. When a mobile robot appears at a location, there is no clear way to understand what navigational path the robot planned and experienced just by looking at it. In this work, we address the generation of narrations of autonomous mobile robot navigation experiences. We contribute the concept of *verbalization* as a parallel to the well-studied concept of visualization. Through verbalizations, robots can describe through language what they experience, in particular in their paths. For every executed path, we consider many possible verbalizations that could be generated. We introduce the *verbalization space* that covers the variability of utterances that the robot may use to narrate its experience to different humans. We present an algorithm for segmenting a path and mapping each segment to an utterance, as a function of the desired point in the verbalization space, and demonstrate its application using our mobile service robot moving in our buildings. We believe our verbalization space and algorithm are applicable to different narrative aspects for many mobile robots, including autonomous cars.

1 Introduction

Service robots can autonomously generate and execute plans to successfully perform tasks for humans, appropriately handling the uncertainty of their surroundings. With mobile robots performing more autonomous behaviors without human intervention, humans in the environment may wonder what exactly the robot was perceiving, predicting, planning, and doing. Robotics researchers have developed logging approaches to enable the recording of the robot experience. For debugging purposes, such developers must dig through the accumulated robot logs to find out about the robot experience in great detail. In addition to researchers, an office worker may want the robot to identify why it was late in completing its task. And a person accompanying the robot may want the robot to summarize its speed and distance traveled. To the authors' knowledge, there are no robots that currently narrate in plain English their planned and executed experiences through a translation of sensor data and plans into natural language. In this work, we introduce *verbalization* as the process of converting or narrating robot experiences via natural language. A robot that verbalizes its experiences could help each of the above example users resolve questions they have about autonomous robot behavior.

Different humans interacting with autonomous robots, as exemplified above, are interested in different specific information, for specific parts of the robot's experience, and at different levels of detail. A one-size-fits-all verbalization will not satisfy all users. We contribute the concept of the *verbalization space* to represent ways in which verbalizations may vary for different reasons, including user preferences and needs. We define our verbalization space across three orthogonal parameters that prior research has indicated per-user needs or preferences over [Dey, 2009; Bohus *et al.*, 2014; Thomason *et al.*, 2015]. The first parameter, *abstraction*, varies the vocabulary and concepts used in the narrative from concrete robot concepts, such as distances, speed, and time to abstract concepts, such as hallways, rooms, landmarks. Second, *specificity* varies the total number of concepts or words used in the summaries, allowing the robot to generate single-sentence general, or multi-sentence detailed, narratives. Finally, *locality* varies the particular parts of the experience that the narration focuses on, from the global path to a local region or landmark of interest. Our verbalization space is general and can be extended to many other parameters.

We first formalize the concept of verbalizing experiences, as well as each of the parameters of our verbalization space with a focus on navigation tasks. We contribute our algorithm for generating narratives using the three verbalization space parameters, and we provide examples of how to combine these parameters. Our algorithm can be adapted to use other natural language generation techniques or verbalization space parameters. Finally, we demonstrate the use of our verbalization space to narrate our mobile robot's experiences through our building, and validate that it generates narratives of different abstraction, specificity, and locality.

2 Related Work

Prior work in automatically generating explanations or summaries of planned behavior can be roughly divided into three categories: 1) intelligibility or explanation of machine learning algorithms, 2) summarizing perceived behavior, and 3) generating directions for humans to follow.

As machine learning gains popularity in many different applications, much human-computer interaction research has focused on ways machine learning applications can *intelligibly* explain their reasoning algorithms to users (e.g., for context-aware systems [Dey, 2009]). HCI intelligibility studies have focused on ways that users can query applications for information or explanations (e.g., [Lim *et al.*, 2009]) as well as how those explanations can affect users' mental models of how the applications work (e.g., [Kulesza *et al.*, 2012; 2013]). The studies find that explanations increase trust of machine learning applications [Bussone *et al.*, 2015] as well as improve users' mental models. Due to the success of intelligibility across many applications, intelligibility toolkits have been implemented for consistency of explanation across different machine learning algorithms [Lim and Dey, 2010]. While prior work shows that varying the focus of explanations is important and useful to users, no one implements it.

Another growing area of research is in summarizing or generating narratives of perceived behavior. For example, RoboCup soccer commentators aim to use the input of simulated RoboCup games [Voelz *et al.*, 1999] or live RoboCup games [Veloso *et al.*, 2008] to generate realtime summaries of the actions in the games. Activity recognition algorithms and natural language generation have also been used to produce annotated accounts of wartime exercises [Luotsinen *et al.*, 2007], video conferencing sessions [Yengui and Mahmoud, 2009], and sports games [Allen *et al.*, 2010]. While some work generates a variety of summaries to maintain human interest (e.g., [Veloso *et al.*, 2008]), the work does not vary the length or depth of summaries as we do.

Finally, and perhaps most closely related to our work, GPS applications (e.g., [Belvin *et al.*, 2001]) and robot applications (e.g., [Kirby *et al.*, 2005; Bohus *et al.*, 2014; Thomason *et al.*, 2015]) are automatically generating navigation instructions and dialog for people to follow and understand. In the prior work, a path is converted into language and ideally presented in an easy-to-understand yet accurate way for the person to follow it seamlessly every time. While these navigation directions do not vary in the language used, recently [Bohus *et al.*, 2014] found that navigation directions should 1) provide differing levels of specificity at different locations in the route and 2) use abstract landmarks in addition to more concrete details. Similarly, prior work on human direction givers shows that humans do not generate the same directions for every person [MacFadden *et al.*, 2003].

We note that none of the prior work focuses on summarizing both perception and plans of a robot or other autonomous vehicle. And while the prior work extensively documents the need for parameterized summaries, none of the prior work, to our knowledge, measures those parameters and contributes an algorithm for actually varying them. In this work, we first contribute verbalization as a method of summarizing what robots actually experience. Based on the findings from prior work as well as the needs of our robots' users, we then propose and formalize our verbalization space that represents the variability in narratives, and we provide an algorithm for generating variable verbalizations of route plans.

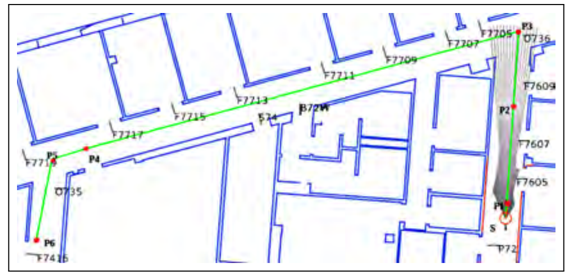


Figure 1: Robot route plan (green lines), nodes $\{S, P_1, ... P_6\}$, Starting node S, and finish node P_6.

3 Route Verbalization

We define *verbalization* as the process by which an autonomous robot converts its own experience into language. In this work, we consider mobile navigation experience in the physical world, and verbalize what the robot experienced while traversing its route. We define *route verbalization* as the process by which an autonomous robot converts its own route experience into language. A robot can generate route verbalizations mentioning the planned route that will be traversed or the route that has been traversed (*i.e.,* a narrative in the future tense is equivalent to GPS driving directions, while a narrative of the past traversed route describes the actual experience). At this time, we do not distinguish between the future and past tenses, exemplifying the applicability across language generation domains.

We first define simple route verbalizations over common robot map and route representations. Then, we describe our annotations to the map and route to accommodate the variation in verbalization that humans require.

3.1 Robot Map and Route Plan

We define an indoor mobile robot's map $M = \langle P, E \rangle$ as set of points $p = (x, y, b, z) \in P$ representing unique locations (x, y) in our buildings b for each floor z and edges $e = \langle p_1, p_2, d, t \rangle \in E$ that connect that connect two points taking time t to traverse distance d.

The points on the map are annotated with semantic *landmarks* represented as *room numbers* (e.g., 7412, 3201) and *room type* (office, kitchen, bathroom, elevator, stairs, other). Points could be annotated with additional information, including the occupants of the office or the names of laboratory spaces (e.g., as in [Rosenthal *et al.*, 2010]). We also maintain lists of *corridors* and *bridges* as points that reside within them (e.g., "7400 corridor" contains office 7401, office 7402, office 7404, etc. and the "7th floor bridge" contains other 71, other 72, etc.). Some points may not appear in any corridor or bridge list if they are in open areas, and some points may reside in two hallways if they occur at hall intersections.

Using our map, our route planner produces plans as trajectories through the environment composed of:

- a starting point S,
- a finish point F,
- an ordered list of intermediate waypoints $W \subset P$, and

		Abstraction, A			
		Level 1	Level 2	Level 3	Level 4
Specificity, S	General Picture	Start and finish point of the complete route	Total distance and time taken for the complete route	Total distance and time taken for the complete route	Starting and ending landmark of complete route
	Summary	Start and finish point for subroute on each floor of each building	Total distance and time taken for subroute on each floor of each building	Total distance and angles for subroute on each floor of each building	Starting and ending landmark for subroute on each floor of each building
	Detailed Narrative	Start and finish points of complete route plus time taken for each edge of route	Angle turned at each point plus the total distance and time taken for each edge of route	Turn direction at each point plus total distance for each edge of route	All landmarks encountered on the route

Table 1: Narrated information depends on preferred Verbalization Space parameters. Information for Abstraction A and Specificity S are shown assuming Locality L is Global. For a different Locality, a subset of the route is generated, and the information provided is computed in terms of the subset.

- a subset of straight line edges in E that connect S to F through W.

Our planner labels waypoints as *turning points* representing the only places the robot turns after traversing straight edges. Figure 1 shows a route plan, the starting point S, and finish point $F = P_6$, as the destination of a task requested by a user. The figure shows turning points $W = \{P_1, P_2, P_3, P_4, P_5\}$, connected by straight line edges (as pictured in green).

3.2 Simple Route Verbalization

Using the map and route plan described above, a simple route verbalization algorithm could interleave turn angles at each point p and distances traversed for each edge e between waypoints. For the route depicted in Figure 1, this simple route verbalization algorithm would produce:

> *I went straight for 8.5 meters and turned left, then straight for 24.9 meters and turned left, then straight for 3.62 meters to reach the destination.*

While this verbalization successfully describes the robot's route, different people in the environment may be expecting more or different information to be provided. For example, we as robotics researchers could be interested in the exact (x, y, b, z) coordinates of the points where the robot turns. Other people in the environment may find landmarks such as room numbers to be useful. We next describe the use of our semantic annotations within our verbalization space.

4 Verbalization Space

We represent the variations in possible narratives of the same route as the *verbalization space*. Each region of the verbalization space represents a different way to generate text to describe the route plan. A user may specify their personalized preferences for verbalization within this space, or the preferences may be inferred from some other source. Our verbalization space contains three orthogonal parameters – abstraction, locality, and specificity – that are well-documented as personal preferences in the literature (e.g., [Dey, 2009; Bohus *et al.*, 2014; Thomason *et al.*, 2015]). Our verbalization space is general and could be extended to include more parameters as needed.

4.1 Verbalization Space Definitions

Table 1 details the way we instantiate verbalizations for specified parameters $(a, l, s) \in (A, L, S)$.

Abstraction A: Our abstraction parameter represents the vocabulary or corpus used in the text generation. In the most concrete form (Level 1), we generate text in terms of the robot's world representation, directly using points (x, y, b, z) from the route plan. Our Level 2 derives turn angles and uses expected or actual traversal time and distances from the points and edges in the plan. Level 3 abstracts the angles and distances into right/left turns and straight segments. And finally, in the highest level of abstraction, Level 4 contains the semantic annotations described above.

Locality L: Locality describes the segment(s) of the route the user is interested in. In the most general case, the user is interested in the route through the entire Global Environment including all buildings and floors. However, an office occupant may only be interested in a particular predefined Region of the route composed of multiple points in the maps (e.g., we limit our regions by building b or building floor b, z). Finally, the occupant may specify a single particular point or landmark for the robot to summarize its route around (e.g., a constant distance around the 8th floor kitchen or Office 4002).

Specificity S: Specificity indicates the number of concepts or details to discuss in the text: the General Picture, the Summary, and the Detailed Narrative. The General Picture contains the most general description of the robot's route, namely the start and finish points (or landmarks), the total distance covered, and/or the time taken (see Table 1). Our Summaries contain this same information for the subroute on each floor of each building. The Detailed Narrative contains a description of each edge of the robot's route.

Next we describe how these verbalization space parameters are used to generate verbalization text.

Algorithm 1 Variable Verbalization Algorithm

Input: *route, verb_pref, map* **Output:** *narrative*

//The verbalization space preferences
1: $(a, l, s) \leftarrow$ *verb_pref*
//Choose which abstraction vocabulary to use
2: *corpus* $\leftarrow$ ChooseAbstractionCorpus(a)
//Annotate the route with relevant map landmarks
3: *annotated_route* $\leftarrow$ AnnotateRoute(*route, map*, a)
//Subset the route based on preferred locality
4: *subset_route* $\leftarrow$ SubsetRoute(*annotated_route*, l)
//Divide the route into segments, one per utterance
5: *route_segs* $\leftarrow$ SegmentRoute(*subset_route, s*)
//Generate utterances for each segment
6: *utterances* $\leftarrow$ NarrateRoute(*route_segs, corpus, a, l, s*)
//Combine utterances into full narrative
7: *narrative* $\leftarrow$ FormSentences(*utterances*)

4.2 Variable Verbalization Algorithm

The Variable Verbalization (VV) algorithm pseudocode is presented in Algorithm 1. The algorithm directly translates the robot's route plan into plain English given the map and the incorporated annotations described above. It takes as input a *route*, a verbalization space preference *verb_pref* = $(a, l, s) \in (A, L, S)$, and a *map* of the environment with locations labeled as above. It starts by choosing what corpus (Level 1-4) to use when generating utterances depending on abstraction preference a (Line 2). Then, the VV algorithm annotates the given route by labeling each point with landmarks and corridor/bridge names using the map (Line 3).

Once the route is annotated with relevant locations, the algorithm extracts the subset of the route that is designated as relevant by the locality preference l (Line 4). We subset Regions by building and floor and Landmarks by a threshold distance around a given point. Both of these subset types can be directly computed from our point representation - Regions using b, z and Landmarks using a distance function around x, y for the given building/floor. The output of this step is another annotated route that is a copy of the route if l=Global Environment. Otherwise, the output is a subset of the route with a new start and finish point.

Using the *subset_route*, the VV algorithm then computes route segments to narrate with respect to the specificity preference s (Line 5). If the specificity preference is a General Picture, our algorithm computes the required abstraction information for a single route segment from S to F. For Summaries, it computes one route segment for each floor of each building and then computes the relevant abstraction information for those segments. In Detailed Narratives, all edges are included in the narrative.

The Algorithm then translates the route segments from Line 5 into plain English using the corpus vocabulary from the annotated map and template sentences (Line 6, examples described next). Finally, after the sentences have been generated for each route segment, the VV algorithm stitches them together (Line 7). The final narrative is returned as the output of the function.

In the next section, we describe our implementation of our algorithm on our mobile robot and its routes.

5 Mobile Robot Route Verbalizations

Our mobile service robot plans and executes tasks autonomously in our buildings [Biswas and Veloso, 2013; 2014], such as accompanying visitors to their meetings and carrying objects to offices [Veloso *et al.*, 2015]. It regularly interacts with humans in the environment through dialog and symbiotic interactions to ask for help [Rosenthal *et al.*, 2010; Perera *et al.*, 2015; Perera and Veloso, 2015]. We found many different people in our environment are interested in what our robot is doing and experiencing as it acts. We as researchers tend to be interested in high specificity, detailed narratives about the global environment. Other people may be interested in narratives about their own office locations at a general picture level. The Variable Verbalization algorithm is implemented on our robot and allows each person to receive a personalized narrative based on their priorities and interests.

We first describe our annotated map and corpus for verbalizations that are input into our Variable Verbalization algorithm. Then, we describe two narratives based on different verbalization space preferences for the same route. Finally, we test our algorithm on different routes through our building to demonstrate how the number of words and numbers changes with each instantiation of our verbalization space.

5.1 Robot Map and Language Corpus

Our robot's environment includes three buildings connected by bridges. Each floor of each building has its own coordinate system. The individual floor maps are linked to each other via the elevators and bridges, so that the robot can use multiple floors while planning and executing. The set of all floors and all buildings is defined as our map M. Our map contains points p representing any arbitrary location on the map. Points can be labeled as landmarks representing specific room numbers and room types including office, lab, kitchen, bathroom, elevator, stairs, printers, and other. We also maintain lists of corridors and bridges as outlined above. Given any two points, start S and finish F, our route planner computes a set of edges and waypoints to travel from S to F.

Our corpus of landmarks on the map (exerpt below) is used for Level 4 of our Abstraction parameter. Our other corpora for our other levels of abstraction are much smaller and include (x, y) "points", "angle" degrees, distance in "meters", "left turns", "right turns", and "u-turns".

$$\begin{cases} \ldots & \ldots \\ \text{Office-3201} & (x, y, Gates, 3^{rd}\ floor) \\ \text{Bathroom-3} & (x, y, Gates, 3^{rd}\ floor) \\ \text{Stairs-34} & (x, y, Gates, 3^{rd}\ floor) \\ \text{Kitchen-71} & (x, y, Gates, 7^{th}\ floor) \\ \text{Office-7401} & (x, y, Gates, 7^{th}\ floor) \\ \text{Office-7412} & (x, y, Gates, 7^{th}\ floor) \\ \ldots & \ldots \end{cases}$$

5.2 Route Experience Variable Verbalization

Using our map, our mobile robot plans routes between points in our building. Figure 2 Top shows one example route (in green) from the 3rd Floor Office 3201 to the 7th Floor Office 7416 in our Gates building. We have labeled in black our annotations over the map including the corridors, the elevators, a bridge, and a kitchen. Figure 2 Bottom shows a

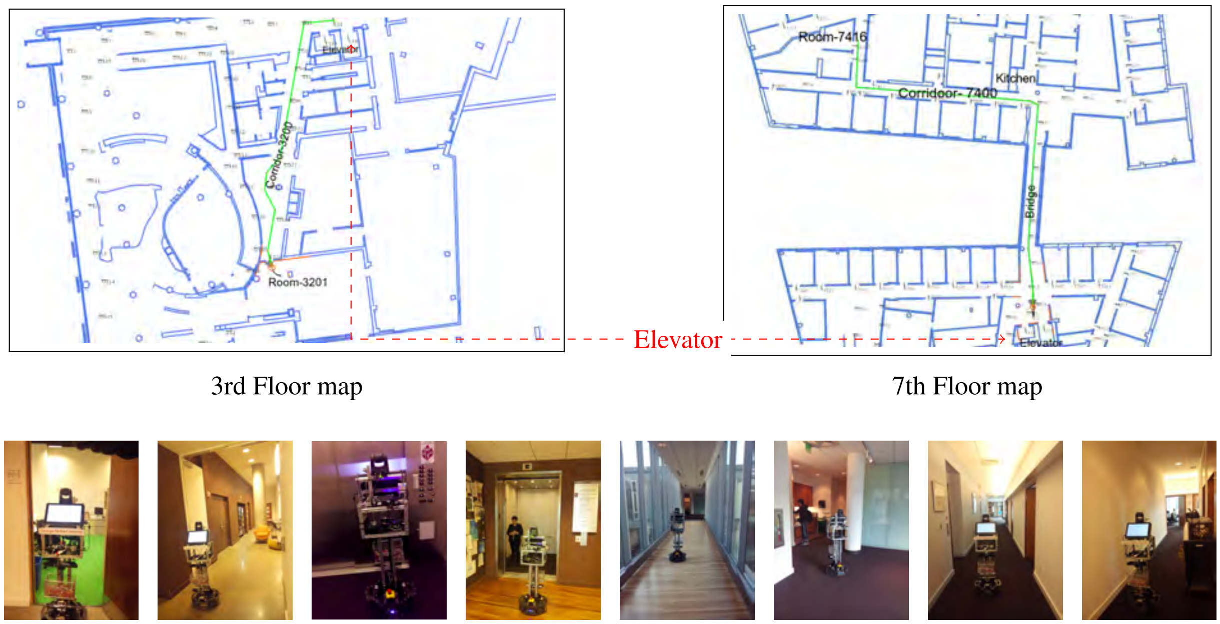


Figure 2: Top: Example of our mobile robot's multi-floor plan in our building (blue walls, green route, red connects elevator between floors). Bottom: Images of our robot navigating the route. The robot (1) starts at Office 3201, (2) travels down the 3200 corridor, and turns right to (3) reach the elevator. Once it (4) reaches the 7th floor, it (5) travels straight across the bridge, (6) turns left at the kitchen, (7) travels down the 7400 corridor, and then (8) makes its first right to Office 7416.

visual depiction of the robot traveling along this route. We demonstrate two variations of verbalizations for the route.

Example 1: Long, Detailed Verbalization

With our map and corpus, we consider the preference:

(Level 4, Global Environment, Detailed Narrative)

that represents a researcher in our lab who wants a detailed description of what happens on each edge of the robot's route. We will review our algorithm's analysis of the route plan to generate a verbalization fitting this preference.

Choose Abstraction Corpus: Because the abstraction level preference is Level 4, the VV algorithm chooses the large corpus of room numbers, room types, and corridors and bridges for its language model.

Annotate Route: Next, the input route is annotated with these landmarks from the corpus. In this case, the VV algorithm labels starting point Office-3201; the points leading to the elevator are Corridor-3200; the elevator on the 3rd floor is labeled Elevator-31 and similarly the 7th floor is labeled Elevator-71; points on the bridge are Bridge-7; the Kitchen-71 is labeled; the hallway points are labeled Corridor-7400; and finally the finish point is Office-7416.

Subset Route: The researcher is interested in the Global Environment Locality, and thus the route is not subsetted.

Segment Route: The researcher would like s =Detailed Narrative. Our algorithm merges all same-labels, resulting in seven route segments. We write segments in terms of their meaning here because there are too many points to enumerate; the robot maintains the list of points on the route.

{s1: Office-3201, s2: Corridor-3200, s3: Elevator, s4: Bridge-7, s5: Kitchen-71, s6: Corridor-7400, s7: Office-7416}

Narrate Route: Our algorithm's ability to narrate a route depends on filling in templates matching different route segments. We manually created the following templates for Level 4 abstractions. We note next to the D whether the type of landmark is specific (e.g., the template must be filled in by a corridor, bridge, etc.), and we note with a slash that the choice of verb is random to prevent repetition by replacing the verbs with a synonym (e.g., [Veloso *et al.*, 2008]). We have similar templates for other abstraction levels that include distances and time to complete the route segments.

- "$[\text{I}]_N$ $[\text{visited/passed}]_V$ the $[\_\_\_]_{D:room}$"
- "$[\text{I}]_N$ $[\text{took}]_V$ the elevator and went to the $[\_\_\_]_{D:floor}$"
- "$[\text{I}]_N$ $[\text{went through/took}]_V$ the $[\_\_\_]_{D:corridor/bridge}$"
- "$[\text{I}]_N$ $[\text{started from}]_V$ the $[\_\_\_]_{D:start}$"
- "$[\text{I}]_N$ $[\text{reached}]_V$ $[\_\_\_]_{D:finish}$"

Using the templates, the VV Algorithm generates utterances for each of the segments.

{s1: "I started from Office 3201",
s2: "I went through the 3200 corridor",
s3: "I took the elevator to the seventh floor",
s4: "I took the 7th floor bridge",
s5: "I passed the kitchen",
s6: "I went through the 7400 corridor",
s7: "I reached Office 7416",

Form Sentences: Finally, the algorithm combines the sentences with "then"s (more complex concatenation could be used):

I started from office 3201, then I went through the 3200 corridor, then I took the elevator and went to the seventh floor, then I took the 7th floor bridge, then I passed the kitchen, then I went through the 7400 corridor, then I reached office 7416.

Example 2: Short Overview Verbalization

To contrast the long detailed landmark-based narrative, a short verbalization can be achieved with preference

(Level 2, Gates 7th Floor Region, General Picture)

Here, a person accompanying the robot wants to know how far they traveled only on the 7th floor. The VV algorithm first annotates our entire route with abstraction Level 2, adding distances to the edges in the route between each pair of points. Since the required locality is Region, the algorithm subsets the route containing only the required Gates 7th floor points. As the specificity is General Picture, a single route segment is generated as the combination of all edges from the new 7th floor start node S to the finish node F. The route is annotated with the total distance and time taken for the route. Next the algorithm narrates the route using the template “$[I]_N$ [traveled] [x] meters in [t] seconds on the $[\_\_]_{D:floor}$”. Finally these utterances could be combined (not necessary here) to form the final narrative:

I traveled 56.18 meters and took 75 seconds on the 7th floor.

5.3 Validation

Given the well-documented need for verbalizations, we focus our experiment on whether we succeed at varying our verbalizations based on those needs. We randomly generated 12 multi-floor routes in our Gates building and 12 single-floor routes, ran the VV algorithm over the route plans, and analyzed the content of the 36 × 24 verbalizations that were generated.

Figure 3 shows the average number of words for two of our parameters: abstraction and specificity. There are many more words in Detailed Narratives (55-104 words) compared to Summaries (14-21) or General Pictures (10-18). We note that the number of words is nearly the same for Summaries and General Pictures. Because our VV implementation creates one phrase per floor of the building for Summaries, it generates the same narrative as the General Picture for single-floor navigation routes. Given that half of our routes are single-floor, the average number of words for Summaries is similar to that of General Picture rather than Detailed Narratives.

Additionally, there are more words generated for Summary/General Picture Level 4 Abstraction than Level 3 or 2. This is due to the landmark descriptions that are more verbose than the time and distances reported. In contrast, for Abstraction Level 4, there are no numbers in most of our narratives as the landmarks are entirely made up of words (Figure 4). The exception is Level 4 Abstractions with Detailed Narratives, which do include office numbers.

The addition of the locality parameter reduces the overall number of words and numbers but shows the same patterns. As the narratives become more focused around a region and then a landmark, there are fewer route segments to describe. We conclude that overall we do successfully vary narratives within our verbalization space.

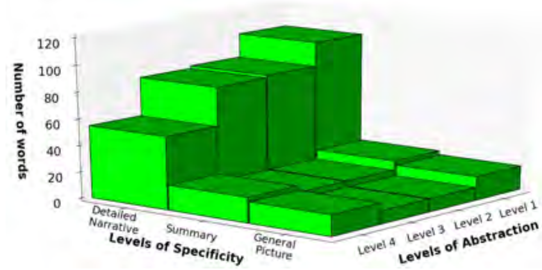


Figure 3: Average number of words generated.

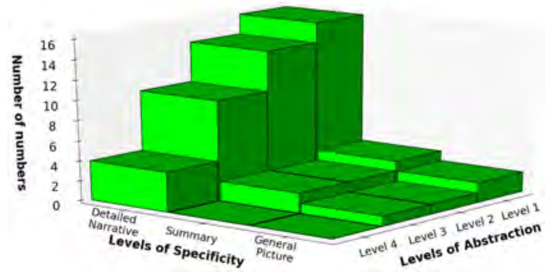


Figure 4: Average number of numbers generated.

6 Conclusion

It is hard, if not impossible, for humans to understand the experience of an autonomous mobile robot. In this paper, we have contributed a novel approach to capture verbalization by a robot as a way for the robot to narrate its experience in natural language. Our mobile robot translates its route experiences in into verbalization utterances. We contribute the verbalization space as a formalization of multiple levels of detail in which narrations can be generated. We introduce different axes of the space to represent different dimensions of verbalization, namely abstraction, locality, and specificity, though the space can be extended.

The approach we present aims at being applicable beyond mobile robots to other planning algorithms, allowing language to be adjusted to the desired levels of detail. For autonomous vehicles, we can imagine using a new map and semantic landmark labels with our same verbalization space and the same verbalization algorithm to produce narrations of driven routes. Autonomous vehicles would reason over points in GPS space, and use landmarks such as buildings, roads, and street signs to create a variety of narrations. Other intelligible machine learning applications could also produce new formalisms for the verbalization space to produce variable narrations.

We demonstrate the use of verbalizations on our mobile service robot. We present two examples of narrations corresponding to different points in the verbalization space for one multi-floor route through our building environment. Then, we validate on 24 routes that a variety of narrations that can be generated from any single plan. Future work will focus on studying techniques for the personalization of verbalization preferences among our building occupants.

Acknowledgments

Copyright 2016 Carnegie Mellon University. This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT. [Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution. DM-0003247

References

[Allen *et al.*, 2010] Nicholas D Allen, John R Templon, Patrick Summerhays McNally, Larry Birnbaum, and Kristian J Hammond. Statsmonkey: A data-driven sports narrative writer. In *AAAI Fall Symposium: Computational Models of Narrative*, 2010.

[Belvin *et al.*, 2001] Robert Belvin, Ron Burns, and Cheryl Hein. Development of the hrl route navigation dialogue system. In *Proceedings of the first international conference on Human language technology research*, pages 1–5. Association for Computational Linguistics, 2001.

[Biswas and Veloso, 2013] Joydeep Biswas and Manuela Veloso. Localization and Navigation of the CoBots over Long-term Deployments. *International Journal of Robotics Research*, 32(14):1679–1694, 2013.

[Biswas and Veloso, 2014] Joydeep Biswas and Manuela Veloso. Episodic Non-Markov Localization: Reasoning about Short-Term and Long-Term Features. In *Proceedings of the International Conference on Robotics and Automation*, 2014.

[Bohus *et al.*, 2014] Dan Bohus, Chit W Saw, and Eric Horvitz. Directions robot: In-the-wild experiences and lessons learned. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems*, pages 637–644, 2014.

[Bussone *et al.*, 2015] Adrian Bussone, Simone Stumpf, and Dympna O'Sullivan. The role of explanations on trust and reliance in clinical decision support systems. *IEEE International Conference on Healthcare Informatics*, 2015.

[Dey, 2009] Anind K Dey. Explanations in context-aware systems. In *ExaCt*, pages 84–93, 2009.

[Kirby *et al.*, 2005] Rachel Kirby, Frank Broz, Jodi Forlizzi, Marek Piotr Michalowski, Anne Mundell, Stephanie Rosenthal, Brennan Peter Sellner, Reid Simmons , Kevin Snipes, Alan Schultz, and Jue Wang. Designing robots for long-term social interaction. In *Proceedings of the International Conference on Intelligent Robots and Systems*, pages 2199 – 2204, 2005.

[Kulesza *et al.*, 2012] Todd Kulesza, Simone Stumpf, Margaret Burnett, and Irwin Kwan. Tell me more?: the effects of mental model soundness on personalizing an intelligent agent. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1–10. ACM, 2012.

[Kulesza *et al.*, 2013] Todd Kulesza, Simone Stumpf, Margaret Burnett, Songping Yang, Irwin Kwan, and Weng-Keen Wong. Too much, too little, or just right? ways explanations impact end users' mental models. In *Visual Languages and Human-Centric Computing (VL/HCC), 2013 IEEE Symposium on*, pages 3–10. IEEE, 2013.

[Lim and Dey, 2010] Brian Y Lim and Anind K Dey. Toolkit to support intelligibility in context-aware applications. In *Proceedings of the 12th ACM international conference on Ubiquitous computing*, pages 13–22. ACM, 2010.

[Lim *et al.*, 2009] Brian Y Lim, Anind K Dey, and Daniel Avrahami. Why and why not explanations improve the intelligibility of context-aware intelligent systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2119–2128. ACM, 2009.

[Luotsinen *et al.*, 2007] Linus J Luotsinen, Hans Fernlund, and Ladislau Bölöni. Automatic annotation of team actions in observations of embodied agents. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, page 9. ACM, 2007.

[MacFadden *et al.*, 2003] Alastair MacFadden, Lorin Elias, and Deborah Saucier. Males and females scan maps similarly, but give directions differently. *Brain and Cognition*, 53(2):297–300, 2003.

[Perera and Veloso, 2015] Vittorio Perera and Manuela Veloso. Handling Complex Commands for Service Robot Task Requests. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2015.

[Perera *et al.*, 2015] Vittorio Perera, Robin Soetens, Thomas Kollar, Mehdi Samadi, Yichao Sun, Daniele Nardi, René van de Molengraft, and Manuela Veloso. Learning task knowledge from dialog and web access. *Robotics*, 4(2):223–252, 2015.

[Rosenthal *et al.*, 2010] Stephanie Rosenthal, Joydeep Biswas, and Manuela Veloso. An Effective Personal Mobile Robot Agent Through Symbiotic Human-Robot Interaction. In *Proceedings of AAMAS'10, the Ninth International Joint Conference on Autonomous Agents and Multi-Agent Systems*, Toronto, Canada, May 2010.

[Thomason *et al.*, 2015] Jesse Thomason, Shiqi Zhang, Raymond Mooney, and Peter Stone. Learning to interpret natural language commands through human-robot dialog. In *Proceedings of the 2015 International Joint Conference on Artificial Intelligence (IJCAI)*, July 2015.

[Veloso *et al.*, 2008] Manuela Veloso, Nicholas Armstrong-Crews, Sonia Chernova, Elisabeth Crawford, Colin McMillen, Maayan Roth, Douglas Vail, and Stefan Zickler. A team of humanoid game commentators. *International Journal of Humanoid Robotics*, 5(03):457–480, 2008.

[Veloso *et al.*, 2015] Manuela Veloso, Joydeep Biswas, Brian Coltin, and Stephanie Rosenthal. CoBots: Robust Symbiotic Autonomous Mobile Service Robots. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2015.

[Voelz *et al.*, 1999] Dirk Voelz, Elisabeth André, Gerd Herzog, and Thomas Rist. Rocco: A robocup soccer commentator system. In *RoboCup-98*, pages 50–60. Springer, 1999.

[Yengui and Mahmoud, 2009] Ameni Yengui and NEJI Mahmoud. Semantic annotation formalism of video-conferencing documents. *Cognition and Exploratory Learning in Digital Age*, 2009.

Dynamic Generation and Refinement of Robot Verbalization

Vittorio Perera, Sai P. Selveraj, Stephanie Rosenthal, Manuela Veloso

***Abstract*— With a growing number of robots performing autonomously without human intervention, it is difficult to understand what the robots experience along their routes during execution without looking at execution logs. Rather than looking through logs, our goal is for robots to respond to queries in natural language about what they experience and what routes they have chosen. We propose verbalization as the process of converting route experiences into natural language, and highlight the importance of varying verbalizations based on user preferences. We present our verbalization space representing different dimensions that verbalizations can be varied, and our algorithm for automatically generating them on our CoBot robot. Then we present our study of how users can request different verbalizations in dialog. Using the study data, we learn a language model to map user dialog to the verbalization space. Finally, we demonstrate the use of the learned model within a dialog system in order for any user to request information about CoBot's route experience at varying levels of detail.**

I. INTRODUCTION

We have been investigating autonomous mobile service robots for several years. Our robots perform services that involve moving between locations in our buildings, just traveling to a destination, transporting items from one place to another, or accompanying visitors to offices. Our novel and robust solutions to many challenges of such autonomous behavior have led to the autonomous navigation of more than 1,000km by the robots within the last 3-4 years [1].

Because of the success of the autonomous algorithms, our and other robots consistently move in our environments and they persistently perform tasks for us without any supervision. With robots performing more autonomous behaviors without human intervention, we do not know much about their paths and experience when they arrive at their destinations without delving into the extensive log files. In this work, we propose a new research challenge, namely how to have robots respond to queries, in natural language, about their autonomous choices including their routes taken and experienced. We are interested in ways for robots to *verbalize* (an analogy to visualization) their experiences via natural language.

We notice that different people in the environment may be interested in different specific information, for specific parts of the robot's experience, at different levels of detail, and at different times. A one-size-fits-all verbalization will not satisfy all users. For example, as robotics researchers interested in debugging our robots' behaviors, we often would like our robot to recount its entire path in great detail. On the other hand, an office worker may only want a robot to identify why it arrived late. These variances in preferences are echoed in prior literature in which autonomous systems explain their behavior [2], [3], [4].

In prior work, we have introduced *verbalization spaces* as a way to capture the fact that descriptions of the robot experience are not unique and can greatly vary in a space of different dimensions. We introduced three dimensions of our verbalization space, namely abstraction, specificity, and locality, and associate different levels to each dimension. Based on the underlying geometric map of an environment used for route planning in addition to semantic map annotations, our automated verbalization algorithm generates different explanations as a function of the desired preference within the verbalization space. We present a summary of this prior work including an example verbalization for our CoBot robot in our environment.

In this work, we pursue our research addressing the fact that people will want to request different types of verbalizations through dialog, and may even want to revise their requests through dialog as the robot verbalizes it's route experiences. We present a crowdsourced online study in which participants were told to request types of information represented in our verbalization space. We then provide the robot's verbalization response and asked the participants to write a new request to change the type of information in the presented verbalization. Using the verbalization requests collected from the study, we learn a mapping from the participant-defined language to the parameters in our verbalization space. We show the accuracy of the learned language model increases in the number of participants in our study, indicating that while the vocabulary was diverse it also converged to a manageable set of keywords with a reasonable participant sample size (100 participants). Finally, we demonstrate human-robot dialog that is enabled by our verbalization algorithm and our learned verbalization space language classifier.

II. RELATED WORK

We identify three main categories in the literature on automatically generating explanations or summaries of planned or perceived behavior: 1) intelligibility or explanation of machine learning algorithms, 2) summarizing perceived behavior, and 3) generating directions for humans to follow.

One of the main focus in Human-Computer Interaction research is developing ways for machine learning applications to *intelligibly* explain their reasoning to users (e.g., for context-aware systems [2]). The studies performed on

V. Perera and M. Veloso are with the Computer Science Department, S. Selveraj with the Robotic Institute and S. Rosenthal with Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA 15203 (e-mails: vdperera@cs.cmu.edu, spandise@andrew.cmu.edu, srpomerantz@sei.cmu.edu, mmv@cs.cmu.edu).

intelligibility focus in multiple directions. In [5], the authors look at how users can query applications for information or explanations. The focus of [6], [7] is to explore how the generated explanation can affect the users' mental model of how the applications work. Last, [8] shows how automatically generated explanation can increase the users trust. Another relevant problem is providing summaries or generating narrative of perceived behavior. This problem has been addressed in many different scenarios such as: Robocup soccer games [9], [10], wartime exercises [11], video conferencing sessions [12], or sports games [13]. Finally, automatically generating navigation instructions and dialog for people to follow and understand has become, more and more, a relevant problem in GPS applications (e.g., [14]) and robotics (e.g., [15], [3], [4]).

A common aspect of prior work is the need to vary explanations and summaries according to the user's preference. In [16] the authors show how human direction givers do not generate the same directions for every person. Recently, [3] found that navigation directions *should*: 1) provide differing levels of specificity at different locations in the route and 2) use abstract landmarks in addition to more concrete details. Although the need for parametrized summaries is well documented, none of the prior work, to our knowledge, measures those parameters and contributes an algorithm for varying them.

Our previous work on verbalization space and verbalization algorithm [17] is briefly summarized in Section III. The focus of this work is on how a user might request a variety of verbalizations. The literature of both Human-Human and Human-Robot Interaction focus on how to request additional information when the instructions provided are not clear [18], [19], [20], [21]. Our approach differs as, rather than focusing on changing or repairing instructions when there is a communication breakdown, we allow users to proactively request language variation based on preferences. The contribution of our experiment is twofolds, first to understand the user's language for specifying what information they would want in a verbalization, and second to understand user's language to *change* a verbalization to receive new or different information. We then create a predictive models and demonstrate how we can use them to predict the verbalization preference.

III. ROUTE VERBALIZATION

Previously, we have defined *verbalization* as the process by which an autonomous robot converts its own experience into language. We represent the variations in possible explanations for the same robot experience in the *verbalization space* (VS). Each region in verbalization space represents a different way to generate explanations to describe a robot's experience by providing different information as preferred by the user. Specifically, given an annotated map of the environment, a route plan through the environment, and a point in our verbalization space, our Variable Verbalization Algorithm generates a set of sentences describing the robot's experience following the route plan. We summarize each of these aspects in turn and then provide example verbalizations for our indoor mobile robot CoBot.

A. Environment Map and Route Plans

Our robot maintains an environment map with semantic annotations representing high level landmarks of interest. We define the map $M = \langle P, E\rangle$ as set of points $p = (x, y, m) \in P$ representing unique locations (x, y) locations for each floor map m, and edges $e = \langle p_1, p_2, d, t\rangle \in E$ that connect two points p_1, p_2 taking time t to traverse distance d.

The map is annotated with semantic *landmarks* represented as room numbers (e.g., 7412, 3201) and room type (office, kitchen, bathroom, elevator, stairs, other). The map is also annotated with lists of points as *corridors* which typically contain offices (e.g., "7400 corridor" contains (office 7401, office 7402, ...)) and *bridges* as hallways between offices (e.g., "7th floor bridge" contains (other 71, other 72)).

Using our map, a route planner produces route plans as trajectories through our map. The route plan is composed of a starting point S, finish point F, an ordered list of intermediate waypoints $W \subset P$, and a subset of edges in E that connect S to F through W. Our route planner annotates route plans with *turning points* (e.g., [22]) to indicate the locations where the robot turns after moving straight.

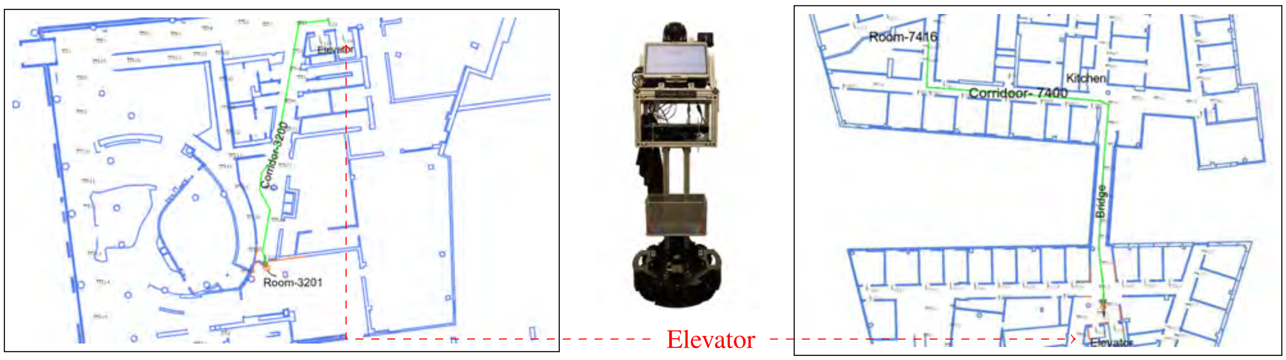
B. Verbalization Space Components

For any given route plan, many different verbalization summaries can be generated. We formalize the space of possible verbalizations as the *verbalization space* (VS) consisting of a set of axes or parameters along which the variability in the explanations are created. For the purpose of describing the path of the CoBot, our VS contains three orthogonal parameters with respect to the environment map and route plan – abstraction, locality, and specificity. These parameters are well-documented in research, though they are not exhaustive ([2], [3], [4]).

Abstraction A: Our abstraction parameter represents the vocabulary or corpus used in the text generation. In the most concrete form (Level 1), we generate explanations in terms of the robot's world representation, directly using points (x, y, m) in the path. Our Level 2 derives angles, traversal time and distances from the points used in Level 1. Level 3 abstracts the angles and distances into right/left turns and straight segments. And finally at the highest level of abstraction, Level 4 contains location information in terms of landmarks, corridors, and bridges from our annotated map.

Locality L: Locality describes the segment(s) of the route plan that the user is interested in. In the most general case, the user is interested in the plan through the entire Global Environment. They may only be interested in a particular Region defined as a subset of points in our map (e.g., the 8th floor or Building 2), or only interested in the details around a Location (e.g., 8th floor kitchen or office 4002).

Specificity S: Specificity indicates the number of concepts or details to discuss in the text. We reason about three levels of specificity, the General Picture, the Summary, and the Detailed Narrative. The General Picture contains a short



3rd Floor map 7th Floor map

Fig. 1. Example of our mobile robot's planning through our buildings. Building walls are blue, the path is green, the elevator that connects the floors is shown in red and shown in black text are our annotations of the important landmarks.

description, only specifying the start and end points or landmarks, the total distance covered and the time taken. The Summary contains more information regarding the path than General Picture, and the Detailed Narrative contains a complete description of the route plan in the desired locality, including a sentence between every pair of turning points.

C. Variable Verbalization Algorithm

Given the route plan, the verbalization preference in terms of (A, L, S), and the environment map, our Variable Verbalization (VV) Algorithm translates the robot's route plan into plain English (pseudocode in Algorithm 1). We demonstrate algorithm with an example CoBot route plan from starting point "office 3201" to finish point "office 7416" as shown in Figure 1. In this example, the user preference is (Level 4, Global Environment, Detailed Narrative).

Algorithm 1 Variable Verbalization (VV) Algorithm

Input: *path, verb_pref, map* **Output:** *narrative*

//The verbalization space preferences
1: $(a, l, s) \leftarrow$ *verb_pref*
//Choose which abstraction vocabulary to use
2: *corpus* $\leftarrow$ ChooseAbstractionCorpus(a)
//Annotate the path with relevant map landmarks
3: *annotated_path* $\leftarrow$ AnnotatePath(*path, map*, a)
//Subset the path based on preferred locality
4: *subset_path* $\leftarrow$ SubsetPath(*annotated_path*, l)
//Divide the path into segments, one per utterance
5: *path_segments* $\leftarrow$ SegmentPath(*subset_path, s*)
//Generate utterances for each segment
6: *utterances* $\leftarrow$ NarratePath(*path_segments, corpus, a, s*)
//Combine utterances into full narrative
7: *narrative* $\leftarrow$ FormSentences(*utterances*)

The VV Algorithm first uses abstraction preference a to choose which corpus (points, distances, or landmarks) to use when generating utterances (Line 2). Since the abstraction preference in the example is Level 4, the VV algorithm chooses corpus of landmarks, bridges and corridors from the annotated map. The VV algorithm then annotates the route plan by labeling the points along the straight trajectories by their corridor or bridge name and the route plan turning points based on the nearest room name.

Once the path is annotated with relevant locations, the algorithm then extracts the subset of the path that is designated as relevant by the locality preference l (Line 4). In this case, the locality is Global Environment and the algorithm uses the entire path as the subset. The VV algorithm then determines the important segments in the path to narrate with respect to the specificity preference s (Line 5). For Detailed Narratives, our algorithm uses edges between all turning points, resulting in descriptions of the corridors and bridges, landmarks, and the start and finish points:

{s1: Office 3201, s2: Corridor 3200, s3: Elevator, s4: 7th Floor Bridge, s5: 7th Floor Kitchen, s6: Corridor 7400, s7: Office 7416}

The VV Algorithm then uses segment descriptions and phrase templates to compose the verbalization into English utterances (Line 6). Each utterance template consists of a noun N, verb V, and route plan segment description D to allow the robot to consistently describe the starting and finish points, corridors, bridges, landmarks, as well as the time it took to traverse the path segments. The templates could also be varied, for example, to prevent repetition by replacing the verbs with a synonym (e.g., [10]). The following are the templates used on CoBot for the Level 4 abstractions. We note next to the D whether the type of landmark is specific (e.g., the template must be filled in by a corridor, bridge, etc), and we note with a slash that the choice of verb is random.

- "[I]$_N$ [visited/passed]$_V$ the [___]$_{D:room}$"
- "[I]$_N$ [took]$_V$ the elevator and went to the [___]$_{D:floor}$"
- "[I]$_N$ [went through/took]$_V$ the [___]$_{D:corridor/bridge}$"
- "[I]$_N$ [started from]$_V$ the [___]$_{D:start}$"
- "[I]$_N$ [reached]$_V$ [___]$_{D:finish}$"

The template utterances are joined using "then"s but could also be kept as separate sentences. Using the filled-in templates, the VV Algorithm generates the following verbalization (Line 7):

I started from office 3201, I went through the 3200 corridor, then I took the elevator and went to the seventh floor, then I took the 7th floor bridge, then I passed the 7th floor kitchen, then I went through the 7400 corridor, then I reached office 7416.

IV. DIALOG TO REVISE VERBALIZATIONS

Our Variable Verbalization Algorithm takes as input a user's explanation request (a, l, s) in terms of level a of Abstraction, l of Locality, and s of Specificity. We further envision the user to engage in a dialog with the robot to incrementally revise their verbalization preferences. In this section, we contribute an approach for mapping the user's dialog onto a verbalization preference, along the dimensions of the Verbalization Space (VS).

As an example, consider the following request to the robot for an explanation: "Please, tell me exactly your experience for your whole path to get here." Since this sentence refers to the "whole path," the robot uses the Global Environment level in the Locality dimension of the Verbalization Space. Furthermore, as the user uses the term "exactly," the explanation should be at the level of Detailed Narrative in the dimension of Specificity. Finally, although no language feature in the request directly refers to a level of Abstraction, the robot may use a high level of Abstraction, as its default. We then concretely address the problem of dialoguing with the robot to revise an explanation. Once a user asks for and receives a route verbalization, they could be interested in refining such description. If we continue the above example, after the robot offers a detailed description of its path, the user could ask: "OK robot, now tell me only what happened near the elevator." The user is hence asking a revised summary of the task executed, where the language should map the explanation to the same values for Abstraction and Specificity as in the initial description, but now focusing the Locality on the region of the elevator.

Our learned mapping from language-based requests to points in the verbalization space allows the user to dynamically refine previous preferences through dialog.

A. Data Collection

In order to enable a robot to correctly infer the user's initial VS preferences as well as how to move in the VS to refine the preferences, we gathered a corpus of 2400 commands (available at *link*) from a total of 100 participants through an Amazon Mechanical Turk survey (www.mturk.com) in which each participant was asked 12 times to request information about our robot's paths and then refine their request for different information. Table I shows a sample of the corpus.

Please give me a summary of statistics regarding the time that you took in each segment.
Can you tell me about your path just before, during, and after you went on the elevator?
How did you get here?
Can you please eliminate the time and office numbers?
What is the easiest way you have to explain how you came to my office today?
Robot, can you please further elaborate on your path and give me a little more detail?

TABLE I

SAMPLE SENTENCES FROM THE CORPUS

After giving consent to partake in the survey, the users were given instructions in order to complete the survey. These instructions included: 1) a short description of the robot capabilities (i.e., execute task for users and navigate autonomously in the environment) and 2) the context of the interaction with the robot. In particular, we asked the users to imagine the robot had just arrived at their office and they were interested in knowing how it got there. Each time the robot arrived at their office, the participants were given:

- A free-response text field to enter a sentence requesting a particular type of summary of the robot's path,
- An example of the summary the robot could provide, and finally
- A second free-response text field to enter a new way to query the robot assuming their interest changed.

This process was repeated 12 times for different parts of our VS. Figure 2 shows the first page of the survey.

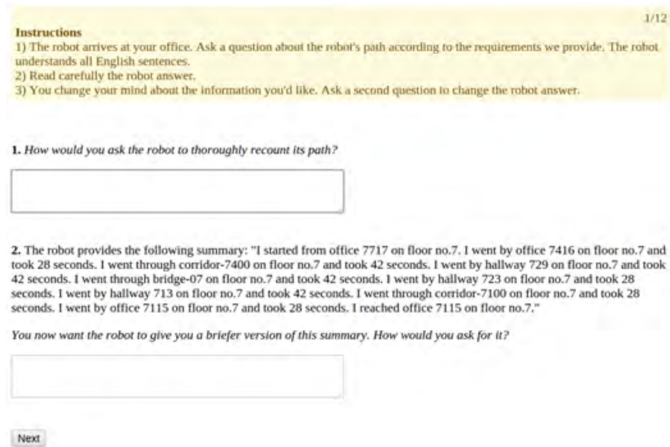


Fig. 2. The survey used to gather out the data corpus. The instructions above the two text fields read: "How would you ask the robot to thoroughly recount its path" and "You now want the robot to give a briefer version of this summary. How would you ask for it?"

We note that the instructions to our survey purposefully did not mention the concept of verbalization and did not introduce any of the three dimensions of the verbalization space. Users hence were not primed to use specific ways to query the robot. However, as the sentences in our corpus should cover the whole verbalization space, when asking for the initial sentence on each page, we phrased our request in a way that would refer to a point on one of the axis of the VS. As an example, in Figure 2, we ask for a sentence matching a point with Detailed Narrative Specificity, and therefore we ask "How would you ask the robot to thoroughly recount its path?". The second sentence we requested on each page refers to a point on the same axis but with opposite value. In Figure 2, we look for a sentence matching a point with General Picture specificity, and we ask the user "You now want the robot to give you a briefer version of this summary. How would you ask for it?". In the first 6 pages of the survey, we asked for an initial sentence matching a point for each possible dimension (Abstraction/Specificity/Locality) at the extreme values. The same questions were asked a second time in the remaining 6 pages of the survey. Table II shows the phrasing for each dimension/value pair.

Abstraction	High Low	"How would you ask the robot for an easy to read recount of its path?" "How would you ask the robot for a recount of its path in terms of what the robot computes?"
Specificity	High Low	"How would you ask the robot to thoroughly recount its path?" " How would you ask the robot to briefly recount its path?"
Locality	High Low	"How would you ask the robot to focus its recounting of the path near the elevator?" "How would you ask the robot to recount each part of its entire path?"

TABLE II
PHRASING OF SURVEY INSTRUCTIONS

V. LEARNING DIALOG MAPPINGS

We frame the problem of mapping user dialog to VS dimensions of Abstraction, Specificity and Locality as a problem of text classification. In particular, we consider the six possible labels corresponding to two levels, high or low extremes, for each of the three axes of the verbalization space. The corpus gathered from the Mechanical Turk survey was minimally edited to remove minor typos (e.g., 'pleaes' instead of 'please') and automatically labeled. The automatic labeling of the corpus was possible since the ground truth was derived directly from the structure of the survey itself.

To perform the classification, we tried several combinations of features and algorithms. Here, we report on the most successful ones. The features considered for our classification task are unigrams, both in their surface and lemmatized form, bigram and word frequency vectors. We also considered two different algorithms, a Naive Bayes Classifier and Linear Regression. Figure 3 shows the results.

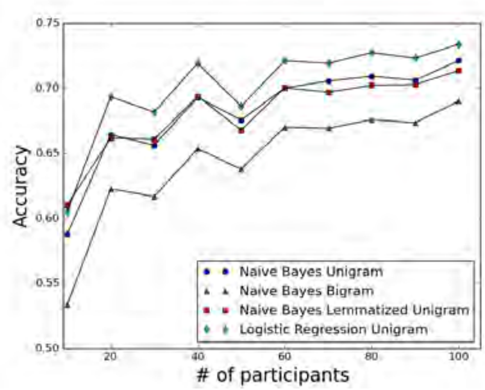


Fig. 3. Experimental results. On the X axis the number of users used to train and test the model, on the Y axis the accuracy achieved.

The X axis shows the number of participants, randomly selected from the pool of 100 survey takers, used to train the model. The Y axis, shows the average accuracy over 10 leave-one-out cross validation tests. As the number of participants increases, all of the proposed approaches improve in performance, as the size of the corpus increases proportionally. Once a robot is deployed and is able to gather more and more sentences asking to verbalize a path, it will then further improve the accuracy of the classification.

When trained on the whole corpus, Logistic Regression achieves the best results with 73.37% accuracy. The accuracy for the Naive Bayes Classifer is 72.11%, 71.35%, and 69% when trained using unigrams, lemmatized unigrams, and bigrams, respectively. Interestingly, the Bayes Classifier and Linear Regression perform similarly with data points differing by less than 2%. Lemmatizing the unigrams does not appear to have a strong effect on the classifier accuracy. Finally, using bigrams negatively effects the classification accuracy. While bigrams encode more information than unigrams, bigrams also naturally produce a more sparse representation of the sentence, which coupled with the size of our corpus, leads to lower accuracy rates.

All the models were trained on the whole corpus and therefore their use is twofold. First, we can use them to infer the correct point in the VS when a user initially queries the robot. Second, we can use the models to dynamically update these values during the dialog. Since our best model achieves 73.37% accuracy the robot is not always going to provide the correct level of information when verbalizing its task. This should not be considered a problem as, using the same models enables the iterative query refinement through dialog.

A. Demonstration

We demonstrate how the learned dialog mapping guides the verbalization process. In this example, we used the Naive Bayes Classifier trained using only unigrams as features to predict the labels of the sentences. We start with the initial sentence given by the user:

USER: Please tell me exactly how you got here.

Our classifier finds the most likely label for the sentence Detailed Narrative, corresponding to point 1 in Figure 4. Accordingly, the robot describes its task:

ROBOT: I started at office 7717. I went by office 7416 and took 28 seconds. I went through corridor 7400 and took 42 seconds. I went by the open area 730 and took 28 seconds. I reached office 7501.

USER: OK, now only tell me about the open area.

Once again the robot classifies the input sentence returning the label Locality Location-specific. Our algorithm now updates the parameters of the verbalization algorithm by increasing the value of locality to produce a new summary, now corresponding to the transitionfrom point 1 to 2 in Figure 4.

ROBOT: Near open area 730. I started from office 7407 on floor and went through corridor-7400.

USER: Can you rephrase in terms of what you computed?

This final command is classified as Level 1 Abstraction. The parameters of the algorithm are updated as shown in the transition from 2 to 3 in Figure 4 and the robot produces a third summary of its task.

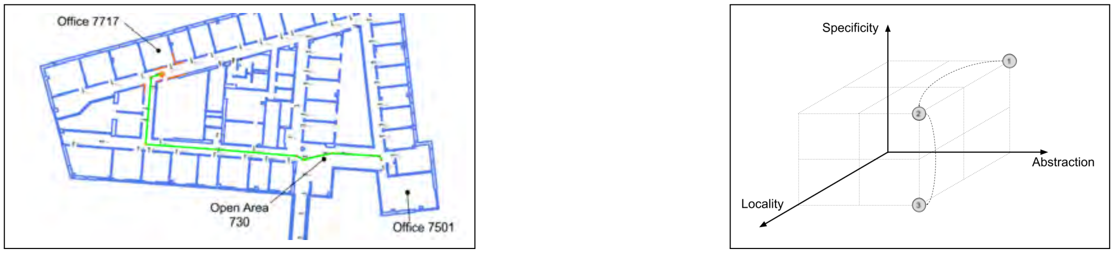


Fig. 4. Demonstration of movements in the verbalization space. On the left the path the robot is going to verbalize. On the right three different point in the verbalization space.

ROBOT: Near O730. Start point is (-1.36, 31.22) on floor GHC7, end point is (10.79, 30.20) on floor GHC7. Start point is (10.79, 30.20) on floor no.GHC7, end point is (16.0, 30.63) on floor no.GHC7.

This dialog sequence demonstrates how the verbalization algorithm, coupled with text classification, allows the robot to understand natural language commands to verbalize its task at different levels. Moreover the user is able to specify what type of information the robot is going to provide through an interactive dialog, to further refine this information without delving into execution logs.

B. *User Study*

Finally, to further demonstrate our approach, we run a user study involving 7 participants. Each participant was first introduced to our CoBot robot and its capabilities, namely to autonomously navigate in the environment and perform item transportation or people guiding tasks. Next, we explained the concept of verbalization space and its three dimensions, Abstraction, Locality, and Specificity. Last, we pointed out that goal of the study was to evaluate the robot's ability to properly explain the path it had traversed. After this explanation phase, the participants were given an initial verbalization of the robot's path. This verbalization was generated by randomly selecting a point in the VS. The subjects were then instructed to provide a sentence to revise the explanation, such that the verbalization would move in a specific direction along one of the three dimensions. The robot then provided a new verbalization by applying the learned classifier, and the users were asked if the revision provided matched, did not match, or almost matched their expectations. Each user dialoged about 4 paths of the robot, and the dialog was repeated 3 times, each for each direction of the VS. There was hence a total of 84 different exchanges between a user and the robot, which were logged.

We first analyze the accuracy of the classifier, in terms of the desired dimension and direction corresponding to the language input. Even if the training from the collected corpus is clearly still limited, the classifier was correct in 54.76% of the cases, i.e., in 46 out of the 84 completely new requests given by the users. In 82,6% of these interactions, the users found that the new verbalization provided matched or almost matched their expectations. In the second step of our analysis, we looked at the remaining 38 interactions where the label returned by the classifier did not match the instructions provided. Surprisingly, the users reported that the verbalization matched their expectations in 21.05% of the cases. By a closer inspection, we found out that the users were confused with the instructions and did not match the directions and dimensions in the verbalization space. For instance, when asked to provide a sentence to move the verbalization towards a *lower specificity* (i.e., a shorter description), one of the users asked "Tell me about your whole path." The classifier labeled this sentence as low locality, the robot extended the verbalization, previously limited in the surroundings of an elevator, to the entirety of the path and matched the users expectations. Table III summarizes the results of the users study.

	Match	Almost Match	Don't Match	
Correct Label	32	6	8	46
Incorrect Label	8	6	24	38
				84

TABLE III
RESULTS OF THE USERS STUDY.

In conclusion, if we consider both the cases where the classifier returned the label meant in the instructions and the cases where the users considered the new verbalization to match their expectations, the dialog was able to provide a correct verbalization in 64.28% of the cases.

VI. CONCLUSIONS

A significant challenge with autonomous mobile robots is understanding what they are doing when there is no human around. We propose verbalization as the process of converting sensor data into natural language to describe a robot's experiences. We review our verbalization space representing different dimensions that verbalizations can be varied, and our algorithm for automatically generating them on our CoBot robot. Then we present our study of how users can request different verbalizations in dialog. Using 2400 utterances collected from the study, we demonstrate that it is possible to learn a language model that maps user dialog to our verbalization space. With greater than 70% accuracy, a robot that uses this model can predict what verbalization a person expects and refine the prediction further through continued dialog. We demonstrate this ability with example verbalizations for CoBot's route experiences.

Acknowledgments

This material is based upon work partially funded by a grant from the Future of Life Institute, by NSF under grant number IIS-1012733; and ONR under grant number N00014-09-1-1031. In addition it was funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN AS-IS BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT. [Distribution Statement A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution. DM-0003431

References

[1] M. Veloso, J. Biswas, B. Coltin, and S. Rosenthal, “CoBots: Robust Symbiotic Autonomous Mobile Service Robots,” in *Proceedings of IJCAI’15*, July 2015.

[2] A. K. Dey, “Explanations in context-aware systems.” in *ExaCt*, 2009, pp. 84–93.

[3] D. Bohus, C. W. Saw, and E. Horvitz, “Directions robot: In-the-wild experiences and lessons learned,” in *Proceedings of the 2014 international conference on Autonomous Agents and Multi-Agent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 2014, pp. 637–644.

[4] J. Thomason, S. Zhang, R. Mooney, and P. Stone, “Learning to interpret natural language commands through human-robot dialog,” in *Proceedings of IJCAI’15*, July 2015.

[5] B. Y. Lim, A. K. Dey, and D. Avrahami, “Why and why not explanations improve the intelligibility of context-aware intelligent systems,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2009, pp. 2119–2128.

[6] T. Kulesza, S. Stumpf, M. Burnett, and I. Kwan, “Tell me more?: the effects of mental model soundness on personalizing an intelligent agent,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2012, pp. 1–10.

[7] T. Kulesza, S. Stumpf, M. Burnett, S. Yang, I. Kwan, and W.-K. Wong, “Too much, too little, or just right? ways explanations impact end users’ mental models,” in *Visual Languages and Human-Centric Computing (VL/HCC), 2013 IEEE Symposium on*. IEEE, 2013, pp. 3–10.

[8] A. Bussone, S. Stumpf, and D. O’Sullivan, “The role of explanations on trust and reliance in clinical decision support systems,” *IEEE International Conference on Healthcare Informatics*, 2015.

[9] D. Voelz, E. André, G. Herzog, and T. Rist, “Rocco: A robocup soccer commentator system,” in *RoboCup-98: Robot Soccer World Cup II*. Springer, 1999, pp. 50–60.

[10] M. Veloso, N. Armstrong-Crews, S. Chernova, E. Crawford, C. McMillen, M. Roth, D. Vail, and S. Zickler, “A team of humanoid game commentators,” *International Journal of Humanoid Robotics*, vol. 5, no. 03, pp. 457–480, 2008.

[11] L. J. Luotsinen, H. Fernlund, and L. Bölöni, “Automatic annotation of team actions in observations of embodied agents,” in *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*. ACM, 2007, p. 9.

[12] A. Yengui and N. Mahmoud, “Semantic annotation formalism of video-conferencing documents,” *Cognition and Exploratory Learning in Digital Age (CELDA 2009), Rome (Italy)*, 2009.

[13] N. D. Allen, J. R. Templon, P. S. McNally, L. Birnbaum, and K. J. Hammond, “Statsmonkey: A data-driven sports narrative writer.” in *AAAI Fall Symposium: Computational Models of Narrative*, 2010.

[14] R. Belvin, R. Burns, and C. Hein, “Development of the hrl route navigation dialogue system,” in *Proceedings of the first international conference on Human language technology research*. Association for Computational Linguistics, 2001, pp. 1–5.

[15] R. Kirby, F. Broz, J. Forlizzi, M. P. Michalowski, A. Mundell, S. Rosenthal, B. P. Sellner, R. Simmons , K. Snipes, A. Schultz, and J. Wang, “Designing robots for long-term social interaction,” in *Proceedings of IROS’05)*. IEEE, August 2005, pp. 2199 – 2204.

[16] A. MacFadden, L. Elias, and D. Saucier, “Males and females scan maps similarly, but give directions differently,” *Brain and Cognition*, vol. 53, no. 2, pp. 297–300, 2003.

[17] S. Rosenthal, S. P. Selvaraj, and M. Veloso, “Verbalization: Narration of Autonomous Mobile Robot Experience,” in *Proceedings of IJCAI’16, the 26th International Joint Conference on Artificial Intelligence*, New York City, NY, July 2016.

[18] J. G. Trafton, N. L. Cassimatis, M. D. Bugajska, D. P. Brock, F. E. Mintz, and A. C. Schultz, “Enabling effective human-robot interaction using perspective-taking in robots,” *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 35, no. 4, pp. 460–470, 2005.

[19] C. Torrey, A. Powers, M. Marge, S. R. Fussell, and S. Kiesler, “Effects of adaptive robot dialogue on information exchange and social relations,” in *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*. ACM, 2006, pp. 126–133.

[20] M. K. Lee, S. Kiesler, J. Forlizzi, S. Srinivasa, and P. Rybski, “Gracefully mitigating breakdowns in robotic services,” in *Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on*. IEEE, 2010, pp. 203–210.

[21] B. M. Allison Saupp, “Effective task training strategies for human and robot instructors,” *Autonomous Robots*, vol. 39, pp. 313–329, 2015.

[22] J. Biswas and M. Veloso, “WiFi Localization and Navigation for Autonomous Indoor Mobile Robots,” in *Proceedings of ICRA’10*, Anchorage, AL, May 2010.

Enhancing Human Understanding of a Mobile Robot's State and Actions using Expressive Lights

Kim Baraka[1], Stephanie Rosenthal[2], and Manuela Veloso[3]

***Abstract*— In order to be successfully integrated into human-populated environments, mobile robots need to express relevant information about their state to the outside world. In particular, animated lights are a promising way to express hidden robot state information such that it is visible at a distance. In this work, we present an online study to evaluate the effect of robot communication through expressive lights on people's understanding of the robot's state and actions. In our study, we use the CoBot mobile service robot with our light interface, designed to express relevant robot information to humans. We evaluate three designed light animations on three corresponding scenarios for each, for a total of nine scenarios. Our results suggest that expressive lights can play a significant role in helping people accurately hypothesize about a mobile robot's state and actions from afar when minimal contextual clues are present. We conclude that lights could be generally used as an effective non-verbal communication modality for mobile robots in the absence of, or as a complement to, other modalities.**

I. Introduction

Mobile robots are entering our daily lives and are expected to carry out tasks with and around humans in environments such as hospitals, supermarkets, hotels, offices, and shops. For effective operation of these robots, it is important that humans have an understanding of some of the processes, states, and actions taking place on the robot pertaining to the tasks performed and to the robot itself. Verbal communication combined with on-screen display is the typical communication mechanism for communicating with humans. However, for autonomous mobile robots in human environments, humans are not always in close proximity to the robot and these communication mechanisms may fail.

Dynamic visual cues [1], and more specifically dynamic lighting [2], have been shown to elicit interactive social responses. These results potentially suggest that *expressive lights* on a robot are likely to create more engaging interactions with humans. These persistent lights might also serve as a complement to existing modalities of interaction which are often transient (e.g., speech) or that require close proximity (e.g., on-screen text). Moreover, in the work mentioned on dynamic visual cues [1], an important part of the social response observed was attributed to the fact that the cues expressed a tangible property of the real world (namely the level of interaction in the environment) in an abstracted way. This observation particularly suggests that an abstracted expression of a robot's state through visual lighting cues may also increase social engagement.

We hypothesize that expressive lights on robots would provide an opportunity to communicate information about robots' state at a distance without the verbal or written cues that are typically used. We focus our study on light expressions for three classes of states in which our autonomous mobile service robot, CoBot, typically finds itself: (1) progress through a task with a fixed goal, (2) obstruction by an obstacle, and (3) need for human intervention.

In our prior work, we conducted a design study in which we captured participants' preferences on how a robot should use lights to express an instance of each of these classes of states [3]. We demonstrated participant consensus on a light animation for each of these instances, with specific animation patterns, speeds and colors associated with them.

In this paper, we present our subsequent study to evaluate the effect of these light expressions on people's understanding of the robot's state/actions when viewed at a distance. In the study, we presented online participants with videos of our robot in nine different scenarios related to the three classes of states mentioned above. These videos shot at a distance emulate the viewpoint of a human observing the robot at a distance, where speech and on-screen text would be imperceptible. Half of the participants saw the robot performing with its expressive lights on and half saw the robot performing with the lights off. Our results show that, even though the particular scenario in which the robot is shown affects the accuracy of participants' understanding of the robot, *the presence of lights significantly increases that understanding regardless of the scenario*. We conclude that using expressive lights to symbolically represent robot states is a promising way to intelligibly communicate this information to humans from afar.

II. Related Work

Light signals have been widely used in the history of mankind to convey information at a distance or in low visibility environments, such as in aviation and maritime navigation [4], but these signals often need to be learned. In contrast, personal electronic devices make use of more intuitive, walk-up-and-use light patterns to convey information to the user. We see light indicators on all sorts of devices from cell phones to toasters, and their expressivity can be greatly

[1]Kim Baraka is with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA. kbaraka@andrew.cmu.edu

[2]Stephanie Rosenthal is with the Software Engineering Institute, Carnegie Mellon University. srosenthal@sei.cmu.edu

[3]Manuela Veloso is with the Computer Science Department, Carnegie Mellon University. mmv@cs.cmu.edu

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. [Distribution Statement A] This material has been approved for public release and unlimited distribution. DM-0003430

exploited to convey diverse information about the device's operation [5]. Expressive lights have been used as well on apparel [6] and interactive art installations [7] [8]. Stage and scene lighting also share common expressive features with indicator lights like color, intensity and time-varying patterns [9], but there the purpose is to illuminate rather than using the light source itself as an expressive modality.

The use of lights for non-verbal communication on robots remains rudimentary. Most of these uses do not have a direct functional role but rather create abstract impressions (such as "artificial subtle expressions" [10]), express emotions [11], or serve as very basic indicators (e.g., battery level). To the best of our knowledge, the only instances of functional light communication in robots are for human-robot speech synchronization [12] and for communicating intent in robot navigation [13]. In [12], an animated LED is used to avoid utterance collisions in verbal human-robot communication by making it subtly blinking between the user's speech end and the robot's speech start. In [13], a LED array is used to communicate direction of navigation on a quadcopter. This last work fits our idea of expressing robot state information through light animations. However, in the prior work, the expressed feature (directionality) remains a low-level one and the light expression has little abstraction to it.

In contrast, we focus on higher-level, task-related features of the robot's state. To map such high-level features to appropriate light expressions, we first needed to understand what animation parameters would accurately represent such abstract concepts. Color theory [14] provides a good starting point for the design of colored light animations carrying meaning (in our case related to robot state). However, it remains difficult to predict the appropriateness of animations for light sources extending in space beyond a single point (e.g., light strips) and expressing meaning in relation to a complex machine such as a robot. In our previous work, we conducted a specialized study to select appropriate designs for robot expression through lights [3], described in the next section. Using our designed animations, in this work we evaluate the effect of the lights on people's understanding of the state and actions of our mobile service robot, CoBot, moving in the real world. We also look at how the effectiveness of our animations generalizes to other scenarios sharing common features with the ones used to design them.

III. Mobile Robot States and Light Expressions

Our autonomous mobile service robot, CoBot, is capable of performing many tasks in the environment [15]. In this section, we describe CoBot's capabilities and discuss opportunities for enhancing the understanding of CoBot's state and actions when viewed at a distance. We present details about our prior work to construct a light animation interface used on CoBot which helps to express robot states [3].

A. *Overview of CoBot and Its Tasks*

CoBot can perform a set of tasks for humans in a building across multiple floors. Building locations (offices, kitchens, and elevators), as well as the navigation map of the building, are known to the robot. CoBot navigates autonomously while avoiding obstacles during its navigation. When facing limitations (such as pressing the button of an elevator or loading/unloading an object on/from the robot), the robot asks for help from humans in the environment [16].

The tasks offered by CoBot are the following:

- **Go-to-Location** task, in which the robot goes from its current position to a goal location.
- **Item transport** task, in which the robot transports an item in its basket from a start location to a goal location.
- **Escort** task, in which the robot accompanies a person from the elevator to a goal location.

When CoBot is moving, it is difficult to discern how much progress it has completed in its task. Similarly, when CoBot is stopped, it can be difficult to discern from a distance whether the robot is stopped for its task, whether an obstacle is blocking its path, or whether it requires help from a human. Expressive lights are one way in which CoBot can help clarify its state to humans in the environment.

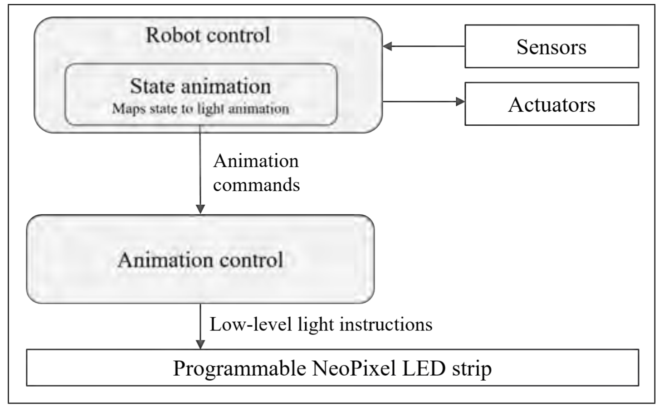


Fig. 1: Diagram of the robot expressive light interface

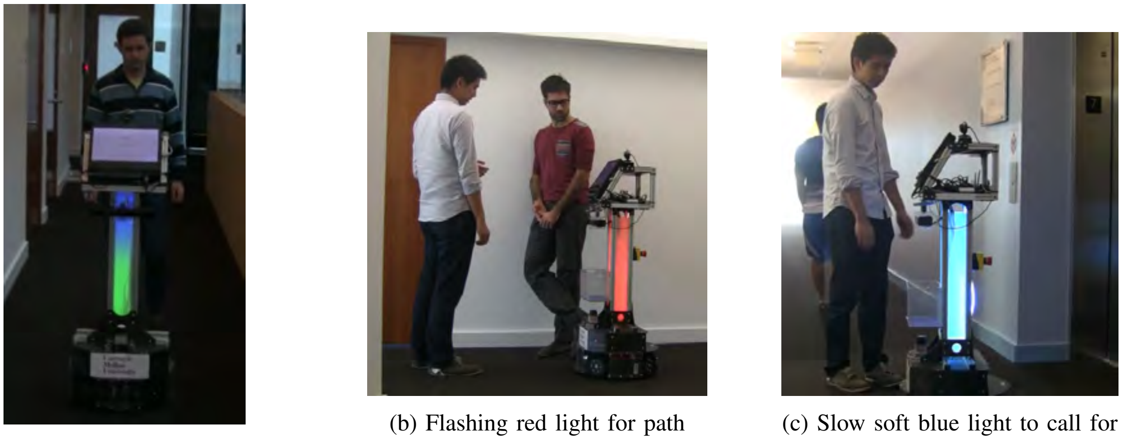
B. *Expressive Light Interface*

For our robot's light interface, we used a programmable, fully addressable NeoPixel LED strip[1] with 91 pixels interfaced to the robot's software through an Arduino microcontroller. The light interface architecture is summarized in Figure 1. A module in CoBot's software translates robot state information into light animation parameters and sends them to the Arduino which performs the hardware-specific light control instructions. Examples of light animations for different scenarios are shown in Figure 2. Our Arduino code[2] is not platform-dependent and is compatible with any robot/device capable of serial USB communication.

C. *Light Expression Design*

The aim of our prior study presented in [3] was to gather "expert" advice about appropriate light animations that could express CoBot's state. Participants were people knowledgeable in one of the following areas: engineering, design, or visual arts. They were provided with information

[1]`https://www.adafruit.com/products/1507`
[2]`https://github.com/kobotics/LED-animation`



(a) Green progress light on an escort task

(b) Flashing red light for path obstructions

(c) Slow soft blue light to call for human help

Fig. 2: Examples of CoBot light animations in different scenarios

about CoBot and its tasks, and were asked how they would demonstrate the following using expressive lights:

- *Progressing to a goal*: CoBot shows its progress on a visitor escort task towards a goal location.
- *Blocked by an obstacle*: CoBot indicates that its path is blocked during task execution.
- *Waiting for human help*: CoBot indicates that it needs human help to press the elevator button.

Participants were asked to vote for the best choice of animation parameters along the following three dimensions for each scenario: (1) animation pattern, (2) speed, and (3) color. For each scenario, the choices presented as videos consisted of three animation patterns, three animation speeds and six colors. Our results showed that participants were consistent in their choices, generally strongly preferring one of the proposed options along each dimension (or two for the color choices). The winning animations, also used in the study of this paper and generalized to more diverse scenarios, are summarized below and depicted in Figure 2.

- "Progressing" animation: a bottom-up progress bar showing the completed distance traveled as a growing portion of the strip lit in bright green.
- "Blocked" animation: a fast red asymmetric fade in / fade out pattern on the whole strip.
- "Waiting" animation: a soft blue slow fade in / fade out pattern on the whole strip.

Note that this study did not control for color blindness (particularly red/green). Though none of the color choices included both red and green in the same animation, the animations should be tested on this population. While this prior study showed consistent results in how an "expert" would design the light animations, in the remainder of the paper we present a study to test those animations on people viewing the robot from afar during a variety of tasks.

IV. Study on the Effect of Light Expressions

In order to evaluate the effectiveness of the chosen expressive light animations, we conducted an online survey in which participants watched videos of a robot performing tasks from afar. At the end of each video, participants were asked to hypothesize about the robot's current *state*, but also about its *actions* (i.e., reasons for performing a specific action such as stopping or being unresponsive). Questions were in a multiple choice format, with four possible answers. Half of the participants saw the robot performing tasks with its expressive lights on ("Lights on" condition), and the other half saw the robot without its expressive lights off ("Lights off" condition). Participants were randomly assigned to one of the two conditions. We analyzed participants' hypothesis choices to demonstrate that those who saw the robot with the lights on were more accurate and gained a higher level of trust in robots from watching the videos.

A. Participants

A total of 42 participants (recruited through email and online advertising), of which 14 were male and 28 were female, took part in the study. Ages ranged from 20 to 67 ($M = 32.4$, $SD = 13.7$). Out of the 42 participants, 33 live in the United States; the rest live in different countries across Asia and Europe. Even though computer usage was relatively high amongst participants (31 out of 42 used computers 30+ hours per week), experience with robots was generally low. Only 5 out of the 42 participants reported having worked with robots before, and 20 reported that they have never seen a robot in person before (3 participants had seen our particular robot, CoBot, before taking the survey). Finally, we ensured that none of the participants were colorblind, since our light animations included color and it could have an effect on our results.

B. Survey Design

Our online video-based survey comprised nine video scenarios of CoBot acting in our environment followed by a multiple choice question asking participants to choose a hypothesis aboutwhat the robot was doing. Four plausible hypotheses about the robot's state/actions were presented as choices for each video, of which one had to be selected. The

TABLE I: Scenarios used in the study

Scenario class	Progressing to a goal (P)	Blocked (B)	Waiting for human input (W)
Scenario 1	Navigation task with human presence (P1)	Human obstacle facing the robot (B1)	Symbiotic autonomy (elevator button) (W1)
Scenario 2	Speech task (P2)	Human obstacles looking away from the robot (B2)	Object loading (W2)
Scenario 3	Battery charging (P3)	Non-human obstacle (B3)	Confirming task completion (W3)

video order, as well as the choices for each answer, were randomized to avoid any order effects.

Each of the video scenarios was recorded using our autonomous robot with lights on and lights off. Although the robot was acting autonomously, the videos were replicated as close as possible for the two conditions. We can reasonably assume that the only notable difference between the two videos for a given scenario is the presence or absence of lights on the robot. The videos did not include any robot speech or any visible information on the robot's screen.

After viewing all nine videos, some relevant background and related information, including trust questions about this particular robot and robots in general, was also collected. A copy of the full survey can be accessed online[3].

C. Scenario Descriptions

The nine scenarios shown in the videos were specifically chosen based on actual tasks that the robot performs while it is deployed in our buildings. We focused our scenarios on the same three common *scenario classes* studied in our prior work – "progressing to a goal", "blocked", and "waiting for human input". For each scenario class, we produced three distinct scenarios in which the robot's state or actions are ambiguous, which are summarized in Table I and described below.

The **"progressing to a goal"** scenarios represent the robot taking actions for a long duration. For each of these scenarios, the progression was modeled as the light expression of a progress bar (see section III-C). The scenarios chosen in this class are:

- *Navigation task with human presence (P1)*: A person is being escorted by the robot to a goal location. When present, the lights show the progress on the distance traveled.
- *Speech task (P2)*: The person asks a question to the robot, which provides no immediate answer, as it is searching the web for the required information. The video ends before the robot responds. When present, the lights show the progress on the web query task.
- *Charging (P3)*: The robot is charging inside the laboratory (the video doesn't show the power plug). When present, the lights show the battery level increasing progressively (video sped up 10 times).

The **"blocked"** scenarios represent the robot being interrupted in its navigation by obstacles of different kinds. The blockage is supported by the fast red flashing light (see section III-C). The scenarios chosen in this class are:

- *Human obstacle facing the robot (B1)*: The robot is blocked in its navigation by a person standing in a narrow corridor, facing the robot.
- *Human obstacles looking away from the robot* (B2): The robot is blocked in its navigation by a person standing in a narrow corridor, facing away from the robot.
- *Non-human obstacle (B3)*: The robot, navigating down a narrow corridor, detects a person walking towards it and changes its navigation path to avoid the person. As a result, it finds itself in front of a branch of plant, which it considers as an obstacle, causing it to stop.

The **"waiting for human input"** scenarios represent the stopped robot waiting for different types of human actions to be taken. For each of these scenarios, the robot is waiting patiently as represented by the slow flashing blue light (see section III-C). The scenarios chosen in this class are:

- *Waiting for help at an elevator (W1)*: The robot is stopped in front of the elevator, waiting for someone to press the elevator button and let it in. People are passing by, ignoring the robot's presence.
- *Object loading (W2)*: The robot is stopped in the kitchen area, facing a counter on which we can see a cup of coffee. Next to the counter area, a person is washing the dishes, presumably unaware of the robot's presence.
- *Confirming task completion (W3)*: The robot is stopped in front of an office door, with coffee in its basket. A person shows up from inside the office and takes the coffee. The robot doesn't react to the person's action and remains still. The person looks at the robot with a confused look on their face.

For each scenario, when lights are present, the default animation on the robot (when no expression is desired) is a static soft blue color.

D. Multiple Choice Questions

After viewing each video, the participants were given choices to explain the robot's state or actions. As discussed earlier, each of the scenarios can be ambiguous to a person viewing CoBot from afar either because of lack of contextual information or because of mixed signals in the robot's behavior. The corresponding answer choices for each video scenario were specifically chosen to reflect many of the possible hypotheses that could correspond to the robot's behaviors. Given our prior work, we theorize that the light expressions will reduce the uncertainty that people have in

[3]https://github.com/kobotics/Surveys/blob/master/survey_printable.pdf

understanding robot's behavior, leading to more accurate answers to our multiple choice questions.

Survey question examples

Scenario B1: *In the video above, why did the robot stop?* (a) The robot recognizes the person, who was expecting it, (b) The robot sees the person as an obstacle, (c) The robot needs help from the person, (d) The robot is inviting the person to use its services. (Scenario B1)

Scenario W3: *In the video above, why is the robot not moving after the person has taken the coffee?* (a) It is waiting for the person to confirm the task is over, (b) It has nothing to do, (c) It is low on battery, (d) It is trying to get inside the room but the door is too narrow.

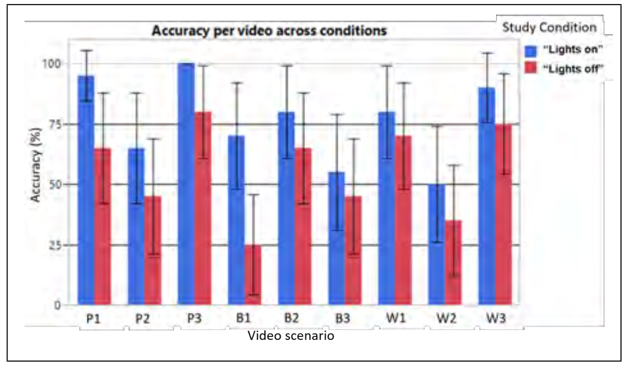


Fig. 3: Comparison of the accuracies on each scenario across the two study conditions with corresponding error bars

V. Results

Responses to the survey multiple choice questions in the nine scenarios were coded in a binary fashion – three answers were coded as wrong and one answer was coded as the correct answer. The resulting dependent variable *accuracy* was modeled as binary categorical. Additionally, we coded the responses to our questions about robot *trust* (5-point Likert scale). We analyzed the effects of our independent variables – experimental *condition* (binary categorical variable "Lights on" and "Lights off") and *scenario* (nine categories) – on the dependent variables. While our scenarios had a range of difficulty resulting in a range of accuracies, our light animations have a statistically significant effect across all scenarios on participant's accuracy. The participants who saw the robots with lights on also indicated an increase in their overall trust in robots more than those who saw the robot with lights off. Detailed results are presented next.

A. Participant Accuracy

In order to analyze our categorical dependent variable *accuracy*, we used a McNemar's chi-square test in a combined between- and within-subject design. The "Lights on/off" *condition* is our between-subject variable. All nine video *scenarios* were shown to all participants (therefore a within-subject variable). The *participant* is modeled as a random variable within the model as each person may be more or less accurate in general. The McNemar's chi-square tested whether the participants' answers depend on the presence/absence of lights, video scenario, and/or the interaction effects of both the lights and video scenario together.

Our results indicate that there is a statistically significant difference in the accuracy based on the presence/absence of lights ("Lights on" $M = 75.66\%$, $SD = 18.20$; "Lights off" $M = 56.08\%$, $SD = 19.16$, $\chi^2(1) = 22.34$, $p < 0.0001$). The accuracy was significantly higher for participants who saw the lights. Additionally, there is a statistically significant difference in participants' accuracy based on the video scenario (see Figure 3 for means and standard deviations, $\chi^2(8) = 51.22$, $p < 0.0001$) (i.e., some videos were harder to determine the robot's state/actions than others for each participant). However, there was no statistically significant effect by the interaction of the light condition and the video scenario ($\chi^2(8) = 8.26$, $p = 0.41$), indicating that the increased effectiveness of the "Lights on" condition was the same across scenarios. Based on these results, we conclude that while the choice of a correct robot state/actions hypothesis does depend on the scenario in which humans see the robot, the tested light animations universally help increase their accuracy.

Figure 3 shows the average accuracy of the participants for each scenario and each light condition. The error bars represent a 95% confidence interval of the mean. We note that the "Lights on" condition accuracies (shown in blue) are universally higher than the "Lights off" accuracies (shown in red). Additionally, the graph clearly shows our result that the video scenarios have different average accuracy, but the accuracy change between conditions per video scenario is not reflective of the scenario.

B. Participant Trust in Robots

On average, participants reported that their trust in robots had increased after watching the videos shown in the survey. (To the question: "Do you agree with the following statement? 'After watching these videos, I will not trust robots as much as I did before.' ", participants in both conditions answered above 3 over 5 on average on a 5-point Likert scale, where 1 meant "Strongly Agree" and 5 meant "Strongly Disagree".) The reported increase in their trust in robots was significantly more pronounced for participants in the "Lights on" condition ($M = 4.29$, $SD = 0.90$) compared to those in the "Lights off" condition ($M = 3.52$, $SD = 0.87$) ($t(40) = 2.02$ two-tailed, $p = 0.008$).

However, there was no statistically significant difference between the two conditions in the reported absolute level of trust in both CoBot and in robots in general ($t(40) = 2.02$ two-tailed, $p > 0.05$), only in the change in trust discussed above did the results differ across conditions.

VI. Discussion

Our results show that our three animations generalize well across several scenarios despite being designed for only a single scenario. Some of our new scenarios (like P3 and W3) even outperform the original scenario. We highlight several aspects of our study design and findings that demonstrate the generalizability of our work to real-world scenarios.

First, in designing our study, we identified three scenarios that fit each of the classes we had previously studied. The significant effect of scenario on response accuracy shows that some survey questions were harder than others. We can attribute some of the differences to the ambiguity of the scenarios - it is sometimes easier to determine CoBot's state and actions than others. However, it is also possible that the four answer choices we designed were more obvious to choose or eliminate depending on the scenario and the question asked. The fact that the lights universally helped participants distinguish CoBot's state better indicates that the effect of our question choices was relatively low.

Next, in designing our study videos, all of them intentionally lacked obvious contextual clues. Lack of such clues is a usual situation when encountering a mobile robot like CoBot. Visitors often encounter the robot for the first time and interact with it with no knowledge about its capabilities, current state, or expectations from humans. Even for people familiar with CoBot, it is difficult to discern whether CoBot is waiting for help (e.g., at an elevator) or waiting for a new task to perform. In such cases, looking at the robot from afar does not give much insight about the robot's operation, unless other visual cues such as our lights are present.

Since these results rely on the legibility of the animations in the presence of minimal contextual clues, we would expect them to hold for real-world encounters, both at a distance and close-by, as long as the lights are clearly noticeable. In fact, CoBot has been running with its lights for over a year, showing escort progress to visitors, eliciting human obstacles to move away and calling for help at elevators. We have seen noticeable differences in people's behavior around CoBot after adding the lights although as of now we have no quantitative data to assess this behavioral change. It would be useful to measure the impact of these lights in real-world, physical interactions with the robot using measures such as robot waiting time or task completion time.

Furthermore, based on the successful generalization of our three expressions, we hypothesize that such expressions might also generalize to: (1) a broader class of scenarios with similar features; (2) other types of users (participants in this study, having no or limited experience with robots or CoBot, were in some way the "worst case analysis"); (3) other types of robots with similar or comparable domains; (4) other types of light arrays or mounting configurations, as long as the lights are easily noticeable; or even (5) multi-modal interactions in which lights are used in conjunction with speech and on-screen interfaces. In the future, we hope to extract design principles for light expressions in robots which could save design and testing efforts across the aforementioned groups as well as be easily portable to diverse platforms.

VII. Conclusion

We have presented an online study to evaluate the effect of expressive lights on people's understanding of a mobile robot, CoBot, carrying out tasks in an office building. We tested three designed light animations on three corresponding classes of scenarios, for a total of nine scenarios. More than just validating the effectiveness and generalizability of our designed light expressions, our results show that the presence of lights on a mobile robot can significantly help people understand the robot's state and actions. Also, some of our interesting results related to robot trust suggest that meaningful expressive lights could contribute to building more solid relationships between mobile robots and humans.

References

[1] B. Mutlu, J. Forlizzi, I. Nourbakhsh, and J. Hodgins, "The use of abstraction and motion in the design of social interfaces," in *Proceedings of the 6th conference on Designing Interactive systems*. ACM, 2006, pp. 251–260.

[2] S. Seitinger, D. M. Taub, and A. S. Taylor, "Light bodies: exploring interactions with responsive lights," in *Proceedings of the fourth international conference on Tangible, embedded, and embodied interaction*. ACM, 2010, pp. 113–120.

[3] K. Baraka, A. Paiva, and M. Veloso, "Expressive lights for revealing mobile service robot state," in *Robot 2015: Second Iberian Robotics Conference*. Springer, 2016, pp. 107–119.

[4] I. Langmuir and W. F. Westendorp, "A study of light signals in aviation and navigation," *Journal of Applied Physics*, vol. 1, no. 5, pp. 273–317, 1931.

[5] C. Harrison, J. Horstman, G. Hsieh, and S. Hudson, "Unlocking the expressivity of point lights," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2012, pp. 1683–1692.

[6] Y. Choi, J. Kim, P. Pan, and J. Jeung, "The considerable elements of the emotion expression using lights in apparel types," in *Proceedings of the 4th international conference on mobile technology, applications, and systems*. ACM, 2007, pp. 662–666.

[7] K. Holmes, "The mood of the chinese internet lights up the facade of beijing's water cube," http://motherboard.vice.com/blog/video-the-great-mood-building-of-china, 2013.

[8] A. Betella, M. Inderbitzin, U. Bernardet, and P. F. Verschure, "Non-anthropomorphic expression of affective states through parametrized abstract motifs," in *Affective Computing and Intelligent Interaction (ACII), 2013 Humaine Association Conference on*. IEEE, 2013, pp. 435–441.

[9] C. De Melo and A. Paiva, "Expression of emotions in virtual humans using lights, shadows, composition and filters," in *Affective Computing and Intelligent Interaction*. Springer, 2007, pp. 546–557.

[10] K. Kobayashi, K. Funakoshi, S. Yamada, M. Nakano, T. Komatsu, and Y. Saito, "Blinking light patterns as artificial subtle expressions in human-robot speech interaction," in *RO-MAN, 2011 IEEE*. IEEE, 2011, pp. 181–186.

[11] M.-g. Kim, H. Sung Lee, J. W. Park, S. Hun Jo, and M. Jin Chung, "Determining color and blinking to support facial expression of a robot for conveying emotional intensity," in *Robot and Human Interactive Communication, 2008. RO-MAN 2008. The 17th IEEE International Symposium on*. IEEE, 2008, pp. 219–224.

[12] K. Funakoshi, K. Kobayashi, M. Nakano, S. Yamada, Y. Kitamura, and H. Tsujino, "Smoothing human-robot speech interactions by using a blinking-light as subtle expression," in *Proceedings of the 10th international conference on Multimodal interfaces*. ACM, 2008, pp. 293–296.

[13] D. Szafir, B. Mutlu, and T. Fong, "Communicating directionality in flying robots," in *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*. ACM, 2015, pp. 19–26.

[14] B. Wright and L. Rainwater, "The meanings of color," *The Journal of general psychology*, vol. 67, no. 1, pp. 89–99, 1962.

[15] M. Veloso, J. Biswas, B. Coltin, and S. Rosenthal, "CoBots: Robust Symbiotic Autonomous Mobile Service Robots," in *Proceedings of IJCAI'15, the International Joint Conference on Artificial Intelligence*, Buenos Aires, Argentina, 2015.

[16] S. Rosenthal, J. Biswas, and M. Veloso, "An Effective Personal Mobile Robot Agent Through Symbiotic Human-Robot Interaction," in *Proceedings of AAMAS'10, the 9th International Joint Conference on Autonomous Agents and Multi-Agent Systems*, Toronto, Canada, 2010.