Bayesian Indoor Navigation Aid for Users with Limited Perceptual Input

Brian J. Stankiewicz
Department of Psychology
Center for Perceptual Systems & Consortium for Computational Cognition
The University of Texas at Austin
Austin, TX 78712-0233 USA
bstankie@mail.utexas.edu

Abstract

Indoor spatial navigation is a challenge for individuals with limited visual input (*Low Vision*) and can be a challenge for individuals with normal visual input in unfamiliar, complex environments. Although outdoor navigation can be achieved using *Global Positioning Systems* (GPS), these signals are not available for indoor navigation. We have developed an indoor navigation aid that uses a Bayesian approach (*Partially Observable Markov Decision Process*; POMDP) to localize and guide a user within an unfamiliar building. We describe the fundamental concept here, in addition to empirical evaluation of the system within virtual environments of varying sizes. Our findings show that the system improves performance when visual information is degraded and also improves performance under normal conditions in large, complex environments.

Introduction

Approximately eight-million Americans report having some form of visual disorder defined by difficulty reading print material in a typical newspaper with corrective lenses (McNeil 2001). These visual disorders can range from being completely blind to having some sort of residual vision. Common causes of these vision losses, or *low vision*, include, but are not limited to, macular degeneration, glaucoma, and diabetic retinopathy. Each of these forms of low vision affect an individual's vision in different ways. Although the specifics of the vision loss may vary significantly from one visual disorder to another (e.g., loss of contrast, acuity or loss of portions of the visual field) and the residual vision for a specific individual with the same **type** of visual disorder may vary significantly, it is commonly recognized that significant vision loss typically has a detrimental effect on an individual's ability to navigate independently (Marston & Golledge 2003).

The Effects of Vision Loss on Independent Mobility

Marston et al. (Marston & Golledge 2003) surveyed 30 individuals with low vision to get an idea of the types of activities that they did each week and whether there were activities that they wanted to participate in but did not engage in because of navigation difficulties associated with their visual disorder. They found that individual with low vision did not participate in 31% of the activities that they would have liked to engage in due to their visual disorder.

The current paper presents a description and evaluation of a low-vision navigation aid that is currently being developed. We expand on recent research in our lab (Stankiewicz *et al.* In Press) showing the benefit of the low-vision navigation aid in virtual and real environments. The current studies investigate the benefit of the system in environments that vary in size (i.e., the number of hallways) and the number of dynamic obstacles (i.e., pedestrians).

Description of the NavAid System

NavAid uses a Bayesian algorithm that, given specific possible actions, costs for making those actions and a destination, can choose the optimal action despite uncertainty about the true state (position and orientation) of the user within the environment. This form of Bayesian analysis is referred to as Partially Observable Markov Decision Processes (POMDP) (Kaelbling, Cassandra, & Kurien 1996; Kaelbling, Littman, & Cassandra 1998; Cassandra, Kaelbling, & Littman 1994; Stankiewicz *et al.* 2006; Sondik 1971). Previously, the POMDP algorithm has been used to guide robots in large-scale environments (Kaelbling, Cassandra, & Kurien 1996; Kaelbling, Littman, & Cassandra 1998; Cassandra, Kaelbling, & Littman 1994) and for measuring human efficiency when navigating with uncertainty (Stankiewicz *et al.* 2006). The current system is not a robot, but uses some of the underlying mathematics to guide and instruct a low-vision user to a specific location within a large-scale space.

Components of the NavAid System

NavAid is comprised of four primary components: an *environment model*, a *range finder*, a *POMDP algorithm*, and a *computer*. The environment model makes explicit the set of states in the environment (positions and orientations), the expected observations for each state within the environment and the resulting states for each action from each state. The set of states in the current system are a series of locations within the environment crossed with four orientations at each location (see Figure 1). The expected observations from each state is simply a distance measurement from the user's current pose to the end of the hall (see Figure 1).

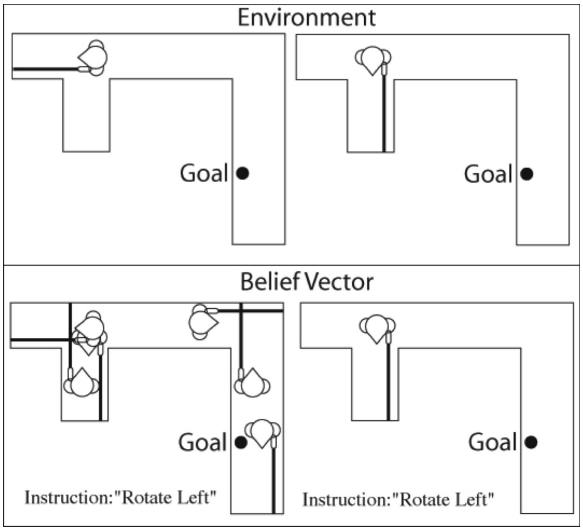

Figure 1: A simple example of how the low-vision navigation concept works. The upper panel illustrates the *true* state of the world, and the lower panel illustrates where the algorithm believes that the user is located. There are 36 states that the observer could be in (9 positions [A-I] x 4 Orientations [N,S,E,W]. The circled triangles in the lower two panels indicate the goal state. The filled triangles indicate the model's current belief about the user's state. In the upper-left panel the user makes his first observation, given that observation the model generates a belief vector specifying the likelihood that the user is in each of the 36 different states (lower-left panel). Given the belief, the model computes that rotate left is the optimal action to do at this time. The user then carries out this action and then takes a second measurement (upper-right panel). Given this measurement in conjunction with the previous measurement and action the model generates a new belief vector (lower-right panel).

Brief Description of the POMDP Algorithm

NavAid is comprised of four primary components: an *environment model*, a *range finder*, a *POMDP algorithm*, and a *computer*. The environment model makes explicit the set of states in the environment (positions and orientations), the expected observations for each state within the environment and the resulting states for each action from each state. The set of states in the current system are a series of locations within the environment crossed with four orientations at each location (see Figure 1). The expected observations from each state is simply a distance measurement from the user's current pose to the end of the hall (see Figure 1).

In addition to the environment model, there is also a range finder that is held by the user to take measurements from the user's current position and orientation within the environment to the end of the hall (see Figure 1). The distance measurements are the *observations* that are given to the POMDP algorithm to inform the system about the user's current state. Given a measurement, the algorithm references the environment model and its current **belief** about where it is in the environment. Thus, given an observation, the model can specify a likelihood that the observer is in any one of the states ($p(state|observation, belief)$); see Figure 1 upper-left and lower-left illustration). Given the likelihood of being in each state, or a *belief vector* (b), the model computes the action that will get the user to the goal state with the largest reward or minimal cost. The NavAid system instructs the user about the optimal action to take. After the user completes the instructed action, the user takes another observation. Given the new observation the model updates its belief about its state within the environment considering the initial observation, the action and the new observation (i.e., $p(s|a, b, o)$). The NavAid system continues through this cycle of observation, belief updating, and action selection until the optimal action is to declare "finished". The "finished" action is given when the model believes that the subject is at the goal state with enough confidence (i.e., likelihood) that outweighs the cost associated for declaring at the wrong state.

To further illustrate the basic concept proposed here, we will reference Figure 1. The upper-left and upper-right figures in Figure 1 illustrate the true-state of the user. In the lower-left and lower-right figures are the corresponding state spaces for the environment. That is, there are nine different positions that the environment model considers (A-I) and four different orientations (N,S,E,W)[1]. This gives us 36 different states that the model will consider. In the upper-left panel, the user is making his initial observation. The vector from the user to the wall represents the range finder measurement from the user to the wall. Given this initial measurement, the model computes the likelihood that the user is in each state. We represent the non-zero likelihoods as solid triangles and the zero likelihoods as open triangles. With the initial observation their are six different states that the user could be in: B-W, B-S, C-N, E-E, G-N, and H-S. The goal state for the system is H-W.

To simplify the illustration, we assume that there is no action or observation noise[2]. Because of this, the likelihood for each non-zero states is 0.1666 (1.0/6.0). Given this belief vector, the model computes the optimal action to reach the goal state with the minimum cost or maximum reward. In this case, the optimal action is "Rotate-Left". This instruction is given to the user and the user then carries out the instruction. The resulting state is illustrated in the upper-right panel of Figure 1. The user then takes a second measurement from his new state. Following the second observation, the model updates its belief vector taking into account the previous belief vector (b), the instructed action (a) ('Rotate-Left') and the new observation (o) using Bayes' Theorem ($p(s|a, b, o)$ where s is a specific *state*) . The updated belief vector is illustrated in the lower-right panel of Figure 1. At this point, the model has no uncertainty about where the user is located. There is only one state in which this sequence of actions and observations could be made which is state B-S. The model continues with this cycle of observation, action

[1]We use N,S,E,W as arbitrary directional units. We could have used up, down, left and right also, or any other notation so long as the orientations are consistent.

[2]When we actually model the environment we model it assuming that the actions are not deterministic.

selection, belief updating until the model was confident that the user had reached the goal state.

The previous illustration provides an intuitive understanding of how the algorithm and system works. However, it does not provide a formal description of the system. If the paper is accepted for presentation, we will request for a full paper (8 pages) that will provide more of the details. Many of the details can be found in (Cassandra, Kaelbling, & Littman 1994; Kaelbling, Littman, & Cassandra 1998; Kaelbling, Cassandra, & Kurien 1996; Thrun *et al.* 1998; Burgard *et al.* 1998; Sondik 1971; Stankiewicz *et al.* In Press; 2006).

Empirical Evaluation

We evaluated the effectiveness of the system in virtual environments of varying sizes. Because we were conducted the current evaluations in virtual, rather than real environments, we could easily manipulate the size and complexity of the virtual indoor environments. Eight different virtual indoor environments were randomly generated by "placing" virtual hallway segments on a cartesian grid. This meant that all of the hallways intersected one another at 90˚angles and the environments were unfamiliar to the participants. The environments were composed of 5, 10, 20, and 40 hallway segments (two environments of each size). The goal of this study was to determine whether there was a benefit for using the NavAid system in environments that were small (e.g., 5 hallways) and large environments (40 hallway segments).

Methods

Undergraduates from the University of Texas at Austin participated in the study and were paid $10.00/hour. In each study, participants started in an unfamiliar environment from an unspecified location within the environment and was instructed to reach a particular goal state within the environment. The goal state was identified by a particular room number within the virtual environment. Within the virtual environment there were signs with numbers placed throughout the environment. The participant's task was to find the goal state as quickly and as efficiently as possible. Participants moved through the environment using a joystick.

In the **NavAid** condition, participants used the **NavAid** system. At the beginning of a trial the NavAid system instructed the user to "Take a measurement". This was completed by making a "trigger" response on the joystick. The computer measured the distance from the user to the farthest object away from the user in the direction that the user was facing. The algorithm used this distance estimation to update its belief (likelihood estimate for each state in the environment) about where it could be within the environment. Using this *belief vector*, the algorithm computed the optimal action that got the user to their destination with the minimum cost. The computer would then verbally instruct the user about this action (e.g., "Turn Around"). The user would generate the action and then take another measurement. The algorithm then updated its current belief, given the previous action and the newly acquired observation. This would continue until the algorithm reached what it believed to be the goal state. At this time the computer would declare "Finished".

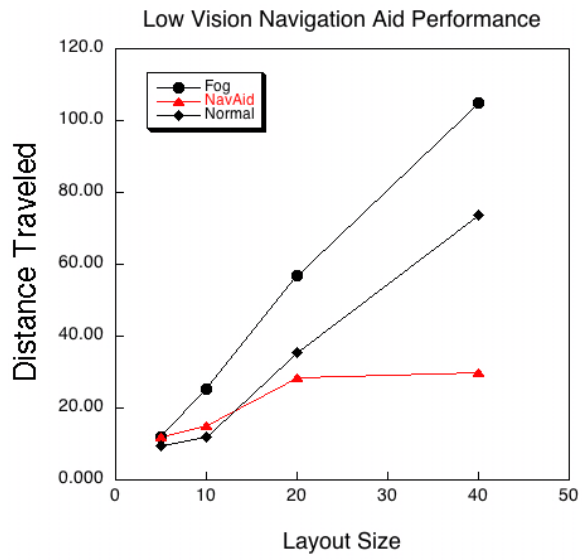

Figure 2: Average distance travelled as a function of the size of the environment. The three curves represent the average distance travelled in the three conditions (Normal Vision, Degraded Vision:NavAid, and Degraded Vision (fog):No NavAid)

Participants were studied in three different conditions: *Normal Vision:No NavAid*, *Degraded Vision:NavAid* and *Degraded Vision: No NavAid*. In the Degraded Vision conditions, fog was added to the environment to reduce visibility. Previous research has shown that degrading vision in this way leads to a reduction in performance (Stankiewicz *et al.* In Press). The Normal Vision condition provides us with a baseline by which to compare performance in the two degraded vision conditions. One interesting comparison is between the Degraded Vision:NavAid and the Normal Vision: No NavAid conditions. This provides will inform us with how well the NavAid system is doing relative to someone with normal vision, and also provides some insight into whether the NavAid system might help individuals who have normal vision.

Results

Our main dependent variable in both studies was the average distance travelled for the participants. Figure 2 shows the average distance travelled for the three conditions as a function of layout size. The primary finding here is that for small environments, five hallway segments or less, there seems to be little to no benefit for the NavAid system. However, as the environment increases in complexity beyond five hallway segments there is a clear advantage for using the NavAid system when visual information is degraded. Participants are navigating with degraded vision with the NavAid system just as well as when they have normal viewing conditions when the environment is ten hallway segments or smaller. However, for environments that have 20 hallways or more,

the participants with degraded vision with the NavAid system are performing better than when they are navigating under normal viewing conditions.

Acknowledgements

This research was supported in part by grants from the National Institutes of Health (EY016089) and from the Air Force Office of Scientific Research (FA9550-04-1-0236 and FA9550-05-1-0321).

References

Burgard, W.; Cremers, A.; Fox, D.; Hahnel, D.; Lakemeyer, G.; Schulz, D.; Steiner, W.; and Thrun, S. 1998. The interactive museum tour-guide robot. *Proceedings of the AAAI Fifteenth National Conference on Artificial Intelligence*.

Cassandra, A. R.; Kaelbling, L. P.; and Littman, M. L. 1994. Acting optimally in partially observable stochastic domains. In *Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI-94)*, volume 2, 1023–1028. Seattle, Washington, USA: AAAI Press/MIT Press.

Kaelbling, L. P.; Cassandra, A. R.; and Kurien, J. A. 1996. Acting under uncertainty: Discrete bayesian models for mobile-robot navigation. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*.

Kaelbling, L. P.; Littman, M. L.; and Cassandra, A. R. 1998. Planning and acting in partially observable stochastic domains. *Artificial Intelligence* 101:99–134.

Marston, J. R., and Golledge, R. G. 2003. The hidden demand for participation in activities and travel by persons who are visually impaired. *Journal of Visual Impairment & Blindness* 97(8):475–488.

McNeil, J. 2001. Americans with disabilities: 1997. Current Population Reports P70-61, U.S. Government Printing Office, Washington, DC.

Sondik, E. 1971. *The Optimal Control of Partially Observable Markov Decision Processes*. Ph.d. thesis, Stanford University.

Stankiewicz, B. J.; Legge, G. E.; Mansfield, J. S.; and Schlicht, E. J. 2006. Lost in virtual space: Studies in human and ideal spatial navigation. *Journal of Experimental Psychology: Human Perception and Performance* 32(3):688–704.

Stankiewicz, B. J.; Cassandra, A. R.; McCabe, M.; and Weathers, W. In Press. Evaluation of a bayesian low-vision navigation aid. *Institute of Electrical and Electronics Engineers Transactions on Systems, Man, and Cybernetic*.

Thrun, S.; Gutmann, S.; Fox, D.; Burgard, W.; and Kuipers, B. J. 1998. Integrating topological and metric maps for mobile robot navigation: A statistical approach. In *Proc. 15th National Conf. on Artificial Intelligence (AAAI-98)*, 989–995. AAAI/MIT Press.