A Human-Robot Mentor-Protégé Relationship to Learn Off-Road Navigation Behavior

Ayanna M. Howard
Human-Automation Systems (HumAnS) Lab
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA, USA

Barry Werger, Homayoun Seraji
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109, USA

Abstract - *In this paper, we present an approach to transfer human expertise for learning off-road navigation behavior to an autonomous mobile robot. The methodology uses the concept of humanized intelligence to combine principal component analysis and neural network learning to embed human driving expertise onto mobile robots. The algorithms are tested in the field using a commercial Pioneer 2AT robot to demonstrate autonomous traversal over rough natural terrain.*

Keywords: human-robot interaction, robot navigation, neural networks, humanized intelligence

1. Introduction

Humans have a remarkable ability to make rational decisions in an environment of uncertainty and imprecision. This capability enables them to perform a wide variety of tasks without exact measurements or explicit computations. For instance, a human can drive a vehicle on a rough terrain using perceptions of the physical environment, rather than with precise mathematical models of the environment specifying the exact locations and sizes of local objects. The driver adjusts the speed and steering of the vehicle based on his subjective judgment of the surface conditions, e.g., the vehicle speed is decreased in driving on a bumpy and rough terrain but is increased on a hard, smooth and flat surface. The human driving actions are motivated by broad perceptions of the terrain quality and obstacles. On the other hand, navigation of hazardous terrain by a mobile robot is a difficult task to overcome. Research such as traversability analysis, deliberative path planning with pre-stored terrain maps and embedded reactive behavior [1] have been used to address the problems of off-road navigation, but the process of successfully navigating between two designated points in rough terrain with minimal human interaction is still an open issue [2]. It is therefore highly desirable to capture the expertise of the human driver and to utilize this knowledge base to develop autonomous navigation systems for mobile robots.

2. Background

The concept of human-robot interaction for enhanced robot navigation capability has been addressed in few research efforts. In [3], Lennon and Atkins show how human perception can augment rover navigation capability for Lunar exploration missions. An astronaut's navigation and planning ability is used to free the rover from time-consuming analysis of the terrain and its hazards. A rule-based following behavior is then designed to visually track the astronaut along a path that is assumed safe. In [4], CORGI, a vision-guided mobile robot, is trained by a human operator that controls the robot to demonstrate behavior desired in typical navigation situations. A neural network is used to map human perception to learned behaviors such as obstacle detection and wall-following. Neural networks have also been used to mimic human driving behavior for navigation along a circular racetrack [5] and for reactive navigation based on sonar input data [6]. Using a real-time driving simulator, Nechyba [7] developed discontinuous human control strategies for abstracting models of human skill directly from observed human input-output data. CMU's research projects ALVIN, RAPLH, and ROBIN [2] develop a human-based automated vehicle driving system by using artificial neural networks to model human driving skills from input-output data. In the CMU projects, the vehicle was successfully able to operate on structured surfaces, such as open or dirt roads and trails.

In the UGV research arena, there has been significant progress in the areas of road-following, obstacle detection and traversability analysis for off-road navigation [2], but the ability to successfully navigate off-road between two designated points with minimal human intervention (i.e. in rough terrain) is still an open problem. Other research efforts focused on mapping human driving skill to a mobile robot have also shown success in limited situations, such as on outdoor roads or in indoor settings. Their results though are not applicable to mobile robots that operate on highly unstructured surfaces where there are no roads or trails to follow, such as on the rough terrains found in natural terrain environments.

To address these issues, we present a methodology to embed human driving expertise onto mobile robots using the concept of humanized intelligence. Recent studies in computational intelligence have shown that the next research direction in computational intelligence involves embedding the capability of a human being directly into the computation system. The term humanized CI [8] classifies this burgeoning research trend and is emphasized through an interactive evolutionary computation system. Mobile robots, although classified as a conventional engineering system, does not directly fall into the humanized CI scheme since their ultimate function is to interact autonomously in natural environments, without human intervention. We desire to embed human capabilities, such as perception and reasoning, directly into the system through seamless human-robot interaction – the human teaches and the robot learns through physical interaction even after the interactions are complete. This process of “humanized intelligence” represents this distinct feature of the mentor-protégé relationship found between human and machine. The following sections describe this system in detail. Section 3 describes the learning algorithms for autonomous navigation. Section 4 presents the human-robot interaction system and Section 5 provides preliminary test results.

3. Learning Algorithms for Autonomous Navigation

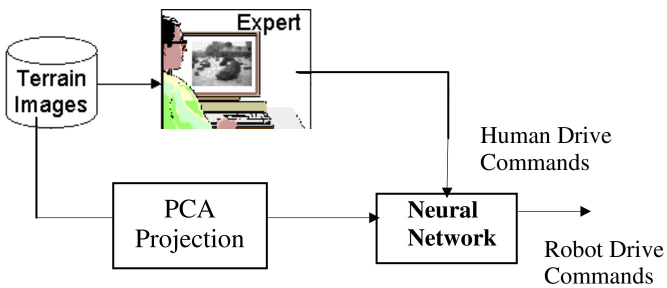

Figure 1: Supervised learning system based on visual terrain input data and human drive commands

In this research, our focus is to develop an autonomous robot navigation system for traversing on rough terrain by using perception-based reasoning and decision-making to embed human expertise directly into the control system (Figure 1). Given a set of terrain images retrieved from camera images, the goal of the navigation algorithm is to determine, in real time, a suitable drive command (speed and turn angle) to navigate a mobile robot through hazardous terrain. In order to accomplish this navigation goal, the following algorithmic steps are implemented:

i. Perception
   - Reduce the environmental input data to a reduced dimensionality, while still maintaining a sufficient data signal to enable transfer of knowledge between human and robot.

ii. Reasoning
   - Train a set of neural network classifiers based on the reduced data set as input, and corresponding human-driving commands as output. The goal of the neural network is to learn speed and turn angle commands based on a human-driver navigating in hazardous terrain.

iii. Decision-Making
   - During field operations and in real time, input a reduced terrain image into the trained neural network and feed the speed and turn command outputs into the mobile robot controller for navigation.

3.1 Perception-Based Capability

To develop human-robot interaction systems, the robot must first develop the ability to mimic the human expert's perception capabilities [9]. In this way, the robotic system can be reasonably confident that decisions made by the system are sound enough to ensure human-equivalent performance. In [10], Zadeh introduces a computational theory to perceptions to enhance the ability of intelligent systems to deal with real-world problems. These systems utilize information consisting of both measurements and perceptions in order to make decisions. The difficulty associated with embedding perceptions in robotic systems is that perceptions are derived from imprecise sensor data. Thus, the first step to perceiving the environment is to extract representative signals that classify the environment in a concise fashion.

Principal Component Analysis (PCA) is a method for data dimensionality reduction, which preserves the most information about a given data set based on a linear construct. PCA involves an orthogonal sub-space projection of the high dimensional terrain images onto a smaller number of dimensions. Once projected, the first principal component (or eigenvector) provides the most information about the data, with the second providing additional information given the value of the first. The last eigenvector accounts for the smallest variance in the data set. The eigenvector approach we utilize for reducing the terrain input data has been validated in relevant real-time applications [11]. We begin by representing a terrain image T_i by a transformed vector of size NxM. Given a set of terrain images $T_1, T_2, \dots T_K$, we then determine the average image A and the covariance matrix R such that:

$$A = \frac{1}{\alpha}\sum_{i=1}^{\alpha} T_i \qquad (1)$$

$$R = \frac{1}{\alpha}\begin{bmatrix} T_1 - A \\ T_2 - A \\ \vdots \\ T_\alpha - A \end{bmatrix}\begin{bmatrix} T_1 - A \\ T_2 - A \\ \vdots \\ T_\alpha - A \end{bmatrix}^{\mathrm{T}} \qquad (2)$$

where α represents the number of terrain images. Based on the covariance calculation, R must be a square matrix, therefore $\alpha = NxM$ for our application. We are now interested in finding a set of eigenvectors V that maximizes the variance found in the input data. We determine the eigenvectors by computing the generalized eigenvector solution:

$$R \times V = V \times D \qquad (3)$$

where R is the covariance matrix, V is the matrix representing the set of eigenvectors, and D is a matrix of eigenvalues derived from the input data. Once the eigenvectors are generated, we use them to create the eigenimage matrix E of our terrain image set such that:

$$E = \begin{matrix} E_1 \\ E_2 \\ . \\ . \\ E_\alpha \end{matrix} = \begin{matrix} T_1 - A & V_1 \\ T_2 - A & V_2 \\ . & . \\ . & . \\ T_\alpha - A & V_\alpha \end{matrix}^{T} \qquad (4)$$

The input data used in this research effort are acquired from a video camera on-board the mobile robot. A sample image set of the terrain environment and a segment of the filter set (i.e. eigenimages) are shown in Figure 2.

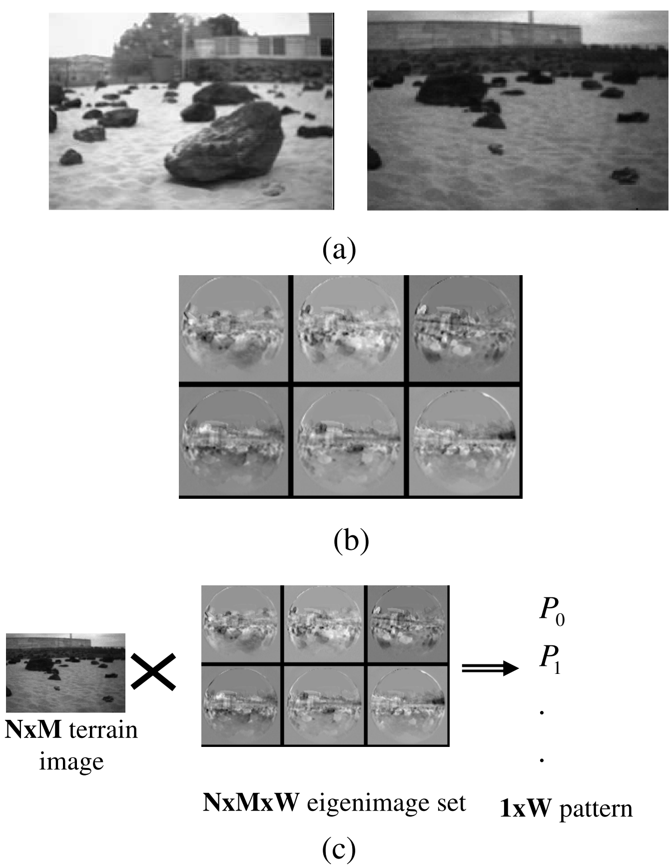

Figure 2. (a) Typical images of a terrain environment (b) Filter set that encodes terrain image signal (c) Pattern created from projecting terrain data onto eigenvector set

3.2 Reason-Based Computation

A neural network allows one to represent arbitrary input-output relationships without being limited to linearity. In this research, we wish to find the relationship between perceptions and human actions. Perceptions are derived from imprecise sensor input data, while corresponding actions are derived from human control variations. Our neural network design consists of a feedforward neural network consisting of an input layer to represent the visual input data and an output layer representing variations of speed and steering commands. Training the network involves finding a set of appropriate weights that mimic the desired human driver action for a given set of visual data input.

Once the eigenimage matrix is calculated, a terrain image T is transformed into a pattern representing the components of the terrain image. Thus, a NxM terrain image can be reduced and transformed into a vector of size W (where W is the number of eigenimages used in the methodology) (Figure 2c). This projection uses a simple operation such that:

$$\begin{matrix} P_1 \\ P_2 \\ . \\ . \\ P_W \end{matrix} = \begin{matrix} E_1 \\ E_2 \\ . \\ . \\ E_W \end{matrix} \left[T_c - A\right]^{T} \qquad (5)$$

where T_c is the NxM terrain vector representing the image acquired by the on-board camera, and P is the projected terrain image that is fed into the neural network classifier (Figure 3). The drive commands consist of speed/turn recommendations and are extracted from the human driver during run-time. We limit each run-time cycle to approximately 10 minutes during training, with an image acquisition rate of 5 frames/sec. The training cycle therefore stores about 3000 images for each run. To limit the noise on the human driving data, turn recommendations are rounded to the nearest 5-degree increment.

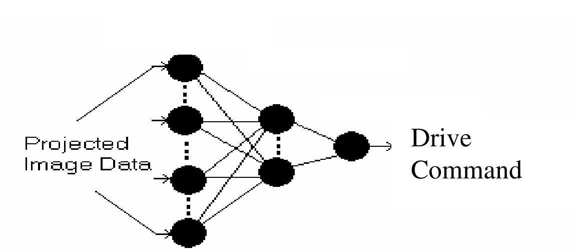

Figure 3. Neural network classifier with the terrain image signal as input and human driving command as output

To determine the optimal size of the neural network, we trained on different sized neural network structures to create a receiver operator curve (ROC) based on drive command error per terrain image, and number of inputs (i.e. eigenimages used in projection) fed into the neural network. For real-time implementation, the smaller the network size, the less computation is required during autonomous control. The ROC analysis allows us to determine a suitable network of minimum size that allows learning of the human-driver commands based on the terrain image data set. Based on this process, we fixed the size of the neural network to 24 inputs nodes and 10 hidden nodes. This corresponds to 24 eigenimages to represent the terrain input data.

4. Human-Robot Interaction System

The human-robot interface system (Figure 4) is used to collect data during the mentor-protégé interaction and consists of a Pioneer 2AT mobile robot that can operate autonomously or be teleoperated (during training) by a human mentor from a base station. The Pioneer is a 4-wheeled skid steering all-terrain commercial robotic platform equipped with an electronic compass, tilt sensor, inertial positioning unit, and FireWire stereo vision system, all connected to an on-board Linux-based laptop computer. The base station has a Macintosh Powerbook running MacOS X, and has an attached joystick that can be used for teleoperation. Images from the robot's vision system are downloaded to the base station to allow the human operator to view the environment from the mobile robot's perspective. The Powerbook communicates, when necessary, to the Linux laptop on-board the mobile robot by way of a wireless Ethernet link. Both the robot and base station laptops run code written in Ayllu [12], a C-based language specialized for Behavior-Based Control; the vision software takes advantage of SRI's Small Vision System for a variety of image processing functions.

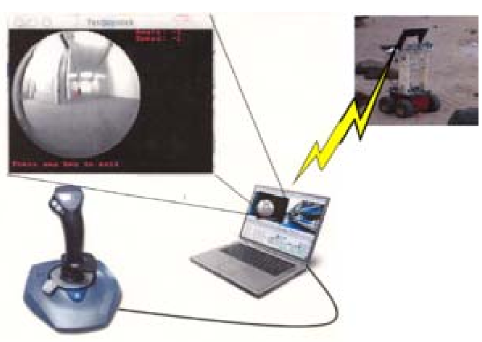

Figure 4. Human-Robot interface system

During human-robot interaction, the interface system sends a continuous stream of images to the base station (Figure 5), where the human mentor watches and sends control inputs via the joystick. The human mentor commands movement of the mobile robot through angular turns of the joystick. The joystick's angular position is than transformed into speed and turn recommendations for control of the mobile robot. Data, consisting of terrain images paired with the mentor's control signals, are collected at the base station for processing by the learning algorithms. After mentor training, the stored data is used to train the neural network on the navigation run. Once robot learning is completed, the learned behavior is used to control the mobile robot to autonomously navigate in the natural terrain environment.

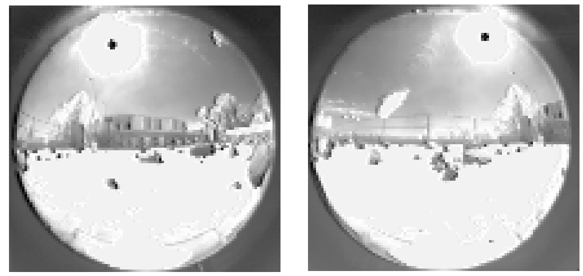

Figure 5. Human viewpoint from mobile robot perspective

5. Decision-Making Test Results

Test runs are used to determine the effectiveness of the human training as the mobile robot autonomously navigates through a test location. The training/testing runs take place in the JPL Mars Yard, a 20m x 20m area that simulates various types of natural terrain environments. A typical scenario is shown in Figure 6 with the associated grid map shown in Figure 7.

Figure 6. Snapshot of Mars Yard terrain environment

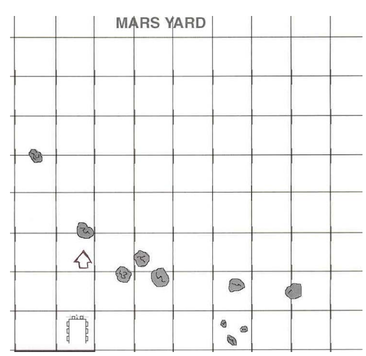

Figure 7. Grid map of Mars Yard terrain environment

The initial training cycle consists of a human operator navigating around local rock obstacles. Four typical scenarios were tested: navigation on a clear straight path toward a large cluster of rock, navigation alongside a rock cluster located to the left, navigation alongside a rock cluster located to the right, and navigation around rock clusters randomly distributed along a path. In these scenarios, we are training the robot to learn the concept of safe navigation, i.e. given a natural terrain environment, find the clearest path of traverse. The human mentor guides the mobile robot safely from a random start to an end goal position corresponding to the desired terrain scenario, while ensuring maximum clearance distance is maintained between the mobile robot and large untraversable rock areas. Human-robot interaction test scenarios of approximately 10-meter traverse distances were used for training the navigation system. To evaluate the capability of the system, the robot was then commanded to autonomously navigate for an additional ten runs at different locations not previously training on, without the human mentor. We were interested in determining whether the robot could perform in any situation, given an environment with similar terrain characteristics. Performance was determined based on comparing the minimum clearance distance between the mobile robot and large obstacle areas, for a human-driven versus autonomous run such that:

$$\mathrm{RobotClearance}_i = \min(\|\mathrm{rockposition}_i - \mathrm{robotposition}(t)\|)$$
$$\mathrm{HumanClearance}_i = \min(\|\mathrm{rockposition}_i - \mathrm{robotposition}(t)\|)$$
$$\mathrm{Error} = \sum \mathrm{HumanClearance}_i / \mathrm{RobotClearance}_i$$

where *rockposition*$_i$ is the x,y position of one of *K* identified untraversable rock areas (labeled a priori for evaluation purposes) over the path, *robotposition* is the x,y position of the mobile robot at time t, *RobotClearance* is the minimum clearance achieved during autonomous traverse, and *HumanClearance* is the minimum clearance achieved during human operation of the mobile robot. An error value nearest to 1.0 represents the ideal learning situation. We select this form of an error metric since we are focusing on learning the concept of *safe* navigation (i.e. obstacle clearance).

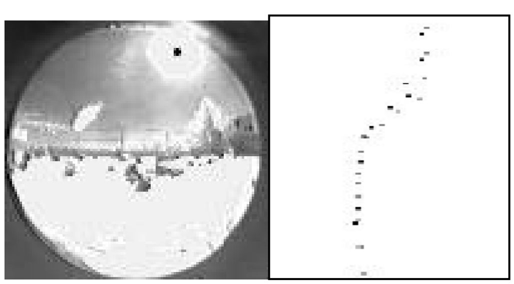

Figure 8. Comparison of driving between mentor (human) and protégé (trained mobile robot)

To compare the performance of the human and robot, we ran the robot autonomously through a test scenario, while having the user provide their command preferences via the joystick. These joystick commands were not sent to the robot, but were stored for performance comparison. A snapshot of a particular instance of one of the robot implementation runs is shown in Table I. Figure 8 shows the corresponding path of the human versus robot. The largest error found in the test runs for the four scenarios was a 0.4 meter clearance difference (*Error* = 1.2) between human directed location and actual robot driven location. In no case did the robot protégé collide with an obstacle during its navigation test runs.

Table I. Comparison of driving commands between mentor (human) and protégé (trained mobile robot)

Mentor		Protégé	
Speed (mm/sec)	Turn (degree/sec)	Speed (mm/sec)	Turn (degrees/sec)
40.5	0	50.2	0.6
37.0	13.2	42.4	12.6
37.7	38.4	37.5	38.4
46.0	4.0	49	9.2

6. Conclusions

In this paper, we discuss the development of a human-robot interaction system to learn off-road navigation behavior. The innovation of our approach is the integration of a learning process that uses data extracted from physical human interaction and modeling of the terrain environment for real-time control and implementation in natural terrain environments. Preliminary test results show that the methodology enables seamless interaction for embedding human expertise in mobile robot systems for safe navigation in natural terrain environments. Future work will focus on expanding the capability of the mobile robot to traverse more complex terrain environments.

7. Acknowledgement

The research described in this paper was performed at the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration.

References

1. E. Tunstel, A. Howard, T. Huntsberger, A. Trebi-Ollenu, J. Dolan: "Applied Soft Computing Strategies for Autonomous Field Robotics," Fusion of Soft Computing and Hard Computing for Autonomous Robotic Systems, Physica-Verlag, 2003.
2. Committee on Army Unmanned Ground Vehicle Technology: Technology Development for Army Unmanned Ground Vehicles, National Academy of Sciences, 2002.

3. J. Lennon, E. Atkins: "A Rule-based Strategy for Astronaut Following Operations," AAAI Spring Symposium on Human Interaction with Autonomous Systems in Complex Environments, March 2003.
4. G. Wyeth: "Training a Vision Guided Mobile Robot," Machine Learning 31(1-3), pp. 201-222, 1998.
5. E. Fix, H. Armstrong: "Modeling human performance with neural networks," Proc. International Joint Conference on Neural Networks, pp. 247--252, 1990.
6. P.K. Pal, A. Kar: "Sonar-based mobile robot navigation through supervised learning on a neural net," Autonomous Robots 3, pp. 355-374, 1996.
7. M. C. Nechyba, Y. Xu: "On learning discontinuous human control strategies," International Journal of Intelligent Systems 16(4), pp. 547-570, 2001.
8. H. Takagi: "Humanized Computational Intelligence with Interactive Evolutionary Computation," Computational Intelligence, the Experts Speak, Wiley-Interscience, 2003.
9. A. Howard, E. Tunstel, D. Edwards, A. Carlson: "Enhancing Fuzzy Robot Navigation Systems by Mimicking Human Visual Perception of Natural Terrain Traversability," Joint 9th IFSA World Congress and 20th NAFIPS International Conference, Vancouver, Canada, July 2001.
10. L. Zadeh: "A New Direction in AI: Toward a Computational Theory of Perceptions," AAAI Magazine, pp. 73 – 84, Spring 2001.
11. A. Howard, C. Padgett, "An Adaptive Learning Methodology for Intelligent Object Detection in Novel Imagery Data," NeuroComputing, vol. 51, pgs. 1-11, March 2003.
12. B. B. Werger: "Ayllu: Distributed port-arbitrated behavior-based control," Proc., 5th Intl. Symp. on Distributed Autonomous Robotic Systems, pp. 25-34, Knoxville, TN, 2000.