Vision-Based Obstacle Run for Teams of Humanoid Robots (Demonstration)

Jacky Baltes
Dept. of Computer Science
University of Manitoba
jacky@cs.umanitoba.ca

Chi Tai Cheng
Dept. of Computer Science
University of Manitoba
tkuggt@cs.umanitoba.ca

Jonathan Bagot
Dept. of Computer Science
University of Manitoba
umbagotj@cs.umanitoba.ca

John Anderson
Dept. of Computer Science
University of Manitoba
andersj@cs.umanitoba.ca

ABSTRACT

This demonstration shows a team of small humanoid robots traverse an environment through a set of obstacles. The robots' brain are implemented using mobile phones for vision, balance, and processing. The robots use particle filters to localize themselves and to map the environment. A frontier-based exploration algorithm is used to direct the robots to overcome obstacles and to explore all regions of the environment.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent Systems

General Terms

Algorithms

Keywords

Visual SLAM, Multiagent Systems, Exploration

This demo shows a team of robots perform a team version of the HuroCup obstacle run event [1]. This event does not only require dexterity and balancing of the humanoid robot, but also the ability simultaneously localize itself and to map a previously unknown environment, the so-called SLAM problem. A SLAM solution gradually builds a map by mapping visible spatial area relative to the current estimated pose of an agent. Our approach to this problem has the following unique features:

Limited Computational Ability: the processors our robots work with are mobile embedded systems of limited processing power. Much of this limited power must be devoted to interpreting visual frames, as well as to the robot application at hand. This both leaves little remaining computational ability to a SLAM algorithm, and compounds the previous problem in that there is a low frame rate for vision and greater noise in visual interpretation.

Cite as: Vision-Based Obstacle Run for Teams of Humanoid Robots (Demonstration), Jacky Baltes, Chi Tai Cheng, Jonathan Bagot, John Anderson, *Proc. of 10th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2011)*, Tumer, Yolum, Sonenberg and Stone (eds.), May, 2–6, 2011, Taipei, Taiwan, pp. 1319-1320.

Vision: The only sensor that our robots use for detecting features in the environment are a single camera. This results in far noisier input data than other sensors such as ladar scanners and also adds a significant computational burden on the robots. The use of vision alone also means that the sensing range of the robots is severly limited, since they can only recognize obstacles in direct line of sight.

Humanoid Robots: This demonstration uses humanoid robots. Humanoid robots pose interesting problems for SLAM, since their motion model has a much wider spread than wheeled robots. For example, the robots often stub their toes leading to very large turns instead of forward movement.Furthermore, the robots have many degrees of freedom, which means that estimating the pose of the robot, which is necessary to measure the angle and distances in the environment, is more complex.

Obstacle Run: the goal of the demonstration is for both robots to cover a field, with three types of obstacles: wall obstacles, step obstacles, and gate obstacles. Wall obstacles are coloured in blue and represent flat walls in the environment. Step obstacles are coloured in yellow and represent small steps that a robot can step over. Red obstacles are gates that a robot can crawl underneath and stand up on the other side.

The constraints of the SLAM problem, along with the desire for efficient exploration and limited computational abilities, point to the use of multiple agents in this problem. Using more than one agent should be able to increase the accuracy of a map through multiple perceptions and the ability to reduce one another's localization error.

The presence of multiple agents should also work to counter limitations on individual robots. Assuming communication is available, the amount of information that can be obtained about the environment by multiple agents in communication with one another should have a greater impact on the SLAM problem than that of n agents operating individually, since each new landmark serves to make future work in SLAM more accurate. Another significant limitation is the battery power available on any one robot: working with a single agent would mean that any significant domain would be impossible to completely map. Other forms of individual limitation can be similarly overcome: Battery life may inhibit an agent from mapping a large environment, and some areas may be inaccessible due to a particular agent's locomotion abilities. Multiple agents, possibly heterogeneous, can increase the coverage percentage by using each agent's resources more effectively.

This paper presents a novel approach to Multi-Agent SLAM. While others (most notably [2, 4]) have developed approaches to multi-agent SLAM, we are moving beyond the limitations of these works.

1. HOMOGENEOUS HUMANOID ROBOTS

The homogeneous robots used to conduct this research are humanoid robots based on Robotis's Bioloid kit. An on-board Atmel AVR ATmega128 micro-controller and Nokia 5500 cellular telephone are interfaced by a custom made infrared data association (IrDA) board containing a Microchip MCP2150 standard protocol stack controller. The on-board micro-controller is used for communication with the servos, such as position interpolation and load checking. This is all made possible by our custom firmware running on our multi-threaded real time operating system (RTOS) FreezerOS also developed by us.

The Nokia 5500 provides a camera, communication mediums (Bluetooth and IrDA), an ARM 9 235MHz processor, and a three axis accelerometer (LIS302DL). The Nokia's processor is used for state generation, image processing, sensor data smoothing, and application programs (including the SLAM approach described here).

2. METHODOLOGY

Our SLAM approach, consists of the use of a particle filter on individual robots to allow an estimation of their current pose, a methodology for mapping, a methodology for exchanging and merging mapped information, and a method for selecting frontiers to reduce redundant exploration. Each of these are explained in the following subsections.

2.1 Particle Filter

The particle filter we employ is a variation on that used by Rekleitis [3], differing in the motion model and particle weight update method. After an action, the pose estimate of each particle is updated based on the motion model. If there was no sensor feedback, the pose estimate of each particle would suffer from this accumulation of odometry error. Our image processing returns the polar coordinates and rough distance of objects in the camera's field of view, but camera data during the humanoid robot's locomotion is extremely noisy due to motion blur. Our weight update method uses a certainty factor in the camera data and a constant decay. The particle population size is 100, which is very small, but manageable with our limited processing power. Population depletion is handled with a simple select with replacement re-sampling algorithm as used by Rekleitis [3].

2.2 Map Representation

Every agent's local map is stored as an occupancy grid with 25x25cm grid cells. A recency value $[0, 255]$ is associated with each grid cell instead of the more common posterior probability. If the recency value of a grid cell is greater than zero, a landmark exists in the corresponding grid cell.

The recency value in occupancy grid cells is updated by an increment or decrement depending on the current sensor reading. If the sensor senses an object, and the coordinates of the object relative to the best particle in the particle filter map to a grid cell with a recency value greater than zero, then the recency value is incremented; otherwise, the grid cell recency value is initialized to 128. If the sensor does not sense an object, landmarks are extended to circles with radius r, if a line segment with length l (maximum sensor range) extended from the best particle intersects a landmark circle, the recency of the corresponding grid cell is decremented.

2.3 Communication and Map Merging

A decentralized, asynchronous communication approach is used between agents via Bluetooth over the logical link control and adaptation protocol (L2CAP) layer. No agent ever waits or relies on information from other agents. An agent uses only what information is available, therefore agents can join or leave the SLAM team at any time without consequence. This also means unreliable communication links between agents are not a problem, beyond the lack of information that results when communication goes down: each agent can still operate independently. Each agent communicates its estimated pose, all landmarks in its local map, and its current target pose to other agents in messages encoded such that the size of each message is as small as possible.

Because entire maps are not exchanged, there is no merging of occupancy grids. Instead, communicated landmarks are integrated into the agent's own map individually through recency update. There are two important elements in this, understanding the local coordinates of others, and actually integrating this information.

To integrate communicated landmarks, we use the recency update method described previously, and assume agents can trust one another (in the sense that there is no duplicity in communication, and that each agent is running an approach such as this one to limit localization error). If the landmark already exists in the agent's map, the greater recency value is selected and the corresponding grid cell is updated. If the landmark does not exist in the agent's map, the corresponding grid cell is simply updated with the received recency.

3. REFERENCES

[1] J. Baltes. *HuroCup Laws of the Game*. University of Manitoba, Winnipeg, Canada, May 2010. http://www.fira.net/hurocup.

[2] W. Burgard, D. Fox, M. Moors, R. Simmons, and S. Thrun. Collaborative multi-robot exploration. In *Proceedings IEEE ICRA-00*, pages 476–481, San Francisco, CA, USA, 2000.

[3] I. Rekleitis. *Cooperative Localization and Multi-Robot Exploration*. PhD thesis, McGill University, January 2003.

[4] I. Rekleitis, G. Dudek, and E. E. Milios. Multi-robot exploration of an unknown environment, efficiently reducing the odometry error. In *Proceedings of IJCAI-97*, pages 1340–1346, Nagoya, Japan, 1997.