Cognitive robotics: a new approach to simultaneous localisation and mapping

Mário Saleiro, J.M.F. Rodrigues and J.M.H du Buf
Vision Lab., Inst. for Systems and Robotics (ISR), University of Algarve (FCT and ISE), Portugal
{masaleiro, jrodrig, dubuf}@ualg.pt

Abstract

Most simultaneous localisation and mapping (SLAM) solutions were developed for navigation of non-cognitive robots. By using a variety of sensors, the distances to walls and other objects are determined, which are then used to generate a map of the environment and to update the robot's position. When developing a cognitive robot, such a solution is not appropriate since it requires accurate sensors and precise odometry, also lacking fundamental features of cognition such as time and memory. In this paper we present a SLAM solution in which such features are taken into account and integrated. Moreover, this method does not require precise odometry nor accurate ranging sensors.

1. Introduction

The capability of simultaneously locating a robot and mapping its environment is an essential pre-requisite to develop truly autonomous robots [3]. In order to navigate safely in unknown environments, a robot must perceive the environment with enough detail. In most cases that perception consists of data acquired by laser, infra-red or ultrasonic sensors which provide accurate distances to walls and objects. Multi-sensor perception, in combination with accurate odometry, has been shown to work well for navigation in static environments, allowing robots to perform tasks like transportation, guidance, search and rescue.

We, humans, can easily solve navigation and information storage problems because of our cognitive capabilities. In fact, we do not need to know exact distances to objects and walls, nor how many centimeters we move each time we take a step or how many degrees we rotate. We also have the capability of selecting which information must be stored and what can be forgotten or ignored. Kawamura et al. [2] presented a biologically inspired SLAM algorithm for egocentric navigation. Their approach is based on two main structures called Sensory Egosphere (SES) and Landmark Egosphere (LES), which would correspond to Short Term Memory (STM) and Long Term Memory (LTM), respectively. The SES comprises all information close to the robot at a certain time, and is always changing when the robot moves. The LES comprises information close to predefined landmarks. Navigation is performed by only estimating the direction from one landmark to another from a hand-drawn sketch of the environment. Both time and memory are directly linked to robot navigation. However, a sketch of the map is already available and detection of landmarks is achieved by detecting coloured tags which were placed in the environment.

The method we propose in this paper is also biologically inspired. It is a first attempt to create a model which mimics human capability to combine spatial localisation with time and memory, enabling it to realise cognitive tasks. Since the goal is to mimic human cognition, the method we propose is only based on vision, without precise odometry, yet allowing the robot to create a 2D map, to navigate, and to locate itself.

2. Cognitive SLAM

At the moment we use only one camera (no stereo) with pan and tilt control to acquire images of the environment. Only two tilt angles are used: pointing a bit down for images of the immediate surround with some detail, and straight ahead for images with less detail further away from the robot. We will refer to them as P1 and P2, respectively. Both P1 and P2 images are processed for object (obstacle) detection. The distance to objects is estimated by using a simple interpolation function which relates each image line to an inaccurate distance. From the information acquired using P1, a very small binary map is created and stored in the robot's STM, which is used for immediate navigation. This allows for a quick reaction in case a mobile obstacle crosses the robot's path. Data acquired from both P1 and P2 is also stored in a bigger map, the robot's LTM. However, this map is not binary; it is a buffer with object labels and their positions. In addition, each object or obstacle detected and stored in LTM is quantified by its degree of certainty. An object which has been detected only once at a certain location has a lower certainty than an object which has been detected more often at the same location. In addition, an object detected using P1 has a higher certainty than one detected using P2, since information from P1 is more detailed than that from P2.

Map building in LTM is achieved through positive reinforcement, negative reinforcement and time-vanishing negative reinforcement. Every time an object is detected at the same location in STM, it is positively reinforced until it has reached a maximum value M, which defines it as a permanent object and its value will not decline over time. Each time a known object cannot be verified in the STM map, its value is negatively reinforced, until the value becomes zero which leads to the elimination of the object. All non-permanent locations in the LTM map which are not updated within a time interval T are negatively reinforced, thereby fading in time if no positive reinforcement is received. This process of map building allows the robot to also manage its memory, by "forgetting" and erasing information which is not consistent or no longer true.

In order to apply corrections of position and orientation, the robot also processes P2 images for landmark recognition. The OpenSURF library [1] is used to perform keypoint detection and to match landmarks contained in its "known landmark library" (KLL). Each landmark may have multiple views stored in the KLL. All images in the KLL are taken with the camera at the same distance D such that the robot can estimate its relative position to the recognised object: the ratio of the diagonals of the bounding boxes which contain the SURF keypoints in both images (P2 and KLL) is proportional to the distance between robot and landmark. Nevertheless, the estimate is not very accurate. When the robot recognises a landmark, it stores the landmark's position and orientation in memory, i.e., the LTM map buffer. When the robot encounters a landmark again, it corrects its own position and orientation in the map. This way, the position error is corrected every time the robot finds a previously detected landmark. With this SLAM algorithm the robot is able to navigate from landmark to landmark, also detecting and avoiding obstacles. The robot also keeps a buffer with its last 10 positions in order to avoid areas which were previously explored while performing a task like finding a specific landmark or returning to the start location.

All processing is done on a laptop which receives images by Wi-Fi from a Surveyor SRV-1 robot. All movement instructions are given in durations, like go forward 200 ms. It is possible to estimate the distance from the duration, but it also depends on the state of the battery and the tracks' adherence to the floor. A test environment was created in a garage with six landmarks 1-6 (shoe boxes, bags, tool boxes), 0 being the start position. Green tape on the floor delimits the robot's sandbox; see Fig. 1 (left). The robot was given the task of finding the dark grey and red tool box, numbered 6, and then returning to its initial position. This was successfully achieved. Figure 1 (right) shows the plotted LTM map buffer acquired by the robot.

3. Conclusions and Further Work

We presented a new framework for a realtime SLAM cognitive robot, using only vision and without precise odometry nor accurate ranging sensors. The only *a priori* knowledge consists of images of obstacles and landmarks. Therefore, the next step is to include autonomous learning of new objects, also solving the problem of memorising their views taken at the same distance D as used for known objects. In addition, a path optimisation algorithm must be included in order to improve global navigation efficiency, such that the system mounted on a bigger robot platform can also explore real environments like corridors etc.

The long-term goal is to integrate advanced models of the processing in our visual system, the so-called what and where pathways which employ multi-scale line, edge and keypoint representations for stereo disparity, optical flow, and invariant object categorisation and recognition [4]. In particular, saliency maps for Focus-of-Attention can be used to steer attention to really important locations, such that a lot of processing of trivial locations can be avoided. The real challenge is to integrate atomic motor and vision actions into seamless visuomotor sequences, as vision is an active process steered by scene and object awareness and behaviour.

Acknowledgements This research is partly supported by the FCT (ISR/IST pluri-annual funding) through the POS Conhecimento Program including FEDER funds, and by the FCT project PTDC/EIA/73633/2006 – SmartVision: active vision for the blind.

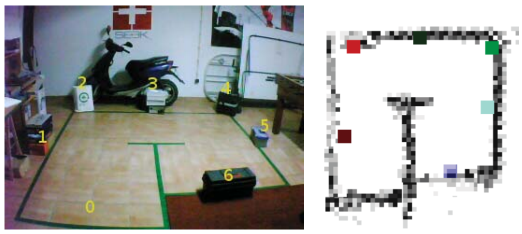

Figure 1. Sandbox with landmarks (left) and plotted LTM after task completion (right).

References

[1] C. Evans. Notes on the opensurf library. *Univ. of Bristol, 2009. www.chrisevansdev.com/computer-vision-opensurf.html*, 2009.

[2] K. Kawamura, A. Koku, D. Wilkes, R. Peters II, and A. Sekmen. Toward egocentric navigation. *Int. J. Rob. Aut.*, 17(4):135–145, 2002.

[3] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A factored solution to the simultaneous localization and mapping problem. *Proc. AAAI. Nat. Conf. Art. Int.*, pages 593–598, 2002.

[4] J. Rodrigues and J. du Buf. A cortical framework for invariant object categorization and recognition. *Cognitive Processing*, 10(3):243–261, 2009.