MITIGATING SPATIAL INTERFERENCE IN A SCALABLE ROBOT RECYCLING SYSTEM

ANDREW VARDY
AUGUST 2015

2014-15 HARRIS CENTRE - MMSB WASTE MANAGEMENT APPLIED RESEARCH FUND

THE HARRIS CENTRE
Memorial University

Contents

1 Acknowledgements

We would like to acknowledge the Harris Centre and the Multi Materials Stewardship Board (MMSB) for their financial support of this project.

2 Executive Summary

The initial aim of this project was to address the issue of spatial interference between robots in a robotic recycling system. The main potential benefit of the proposed robotic recycling system is scalability. The underlying concept is that a swarm of robots process an incoming stream of materials, sorting them into homogeneous clusters of material which can then be quickly bagged and removed. When installed for a large centre, the number of robots would be correspondingly large. However, when installed for a smaller centre—such as a remote community in Newfoundland & Labrador—the number of robots, and therefore the cost of the system would be much lower. The robots themselves would constitute the system, with the additional minimal requirements of an unstructured floor space in which to operate and some input from users to help classify the input materials. A previous Harris Centre / MMSB project to explore this system made some headway, but difficulties were encountered in developing an appropriate set of robots to support further experiments. While the aim of this project was to address the issue of spatial interference, it was found that much more work was required to develop appropriate robots that could transport proxy materials (coloured pucks), classify them, navigate, and exhibit sufficient endurance for meaningful experiments. Therefore, the focus of this project switched to the development of a robot platform with these desired characteristics. It is important to note that the robots under discussion are intended for laboratory experiments using coloured pucks as proxies for real-world recyclables. A transition to robots capable of dealing with real-world conditions is far outside the project's scope.

The main outcome is a robot platform called the BuPiGo which will facilitate our own experiments as well as others who are interested in swarm robotics and other distributed robotic approaches. The BuPiGo fills a key gap in terms of the robots available to researchers. It is not intended to be a product, but an open platform that is easily extensible and makes use of widely available low-cost computing technologies such as the Raspberry Pi and Arduino platforms. It also incorporates an omnidirectional camera system to allow visual navigation between points of interest (e.g. the input point of a recycling facility and the designated output points for sorted materials). We are hopeful that this platform will now allow us to move beyond the development of experimental hardware to develop a complete model of a recycling facility using the concepts described above. This model is a first step towards a real-world scalable recycling facility that would allow remote communities in Newfoundland & Labrador to implement local recycling centres that would minimize transport costs and demonstrate commitment to innovation and sustainability.

3 Introduction

This document comprises the final report for the project "Mitigating Spatial Interference in a Scalable Robotic Recycling System" which was awarded $6,100 in funding as part of the Harris Centre / MMSB Waste Management Applied Research Fund. The application for this funding was submitted in November, 2013 and awarded in February, 2014. The project began on April 4, 2014 with a planned duration of one year. A request for extension was granted so that the project would conclude on August 31, 2015.

The purpose of this project is to continue development of a robotic system for sorting recyclable materials. The long-term vision is to deploy a swarm of robots to sort the recyclable materials produced by remote communities with little access to large-scale recycling facilities. The system will consist of a set of mobile robots with minimal additional infrastructure. The crucial feature that distinguishes this approach is scalability. Materials recovery facilities, such as the one located at Robin Hood Bay in St. John's, may be efficient but they are not economically viable for communities below a certain size. The robotic recycling facility envisioned here could be scaled for a particular community simply by modifying the number of deployed robots.

The fundamental task for these robots is to travel to a source location, grasp an item, classify it, and transport it to the appropriate destination. These activities span a range of active research topics in robotics and computer vision and it would be impossible to study them all in depth. In particular, the problems of grasping and classifying arbitrarily shaped recyclable objects are considered out of scope.

The initial project proposal mentioned two particular areas of focus: mitigating spatial interference between robots and the development of robotic hardware to support this work. The plan was to focus on the first issue, however problems with the robots that we had planned on using prompting a shift to the development of new robots. Therefore, this report will focus on robot hardware development.

4 Development of the BuPiGo Platform

4.1 Introduction

Swarm robotics concerns the design of multi-robot systems which demonstrate a desired collective behaviour in a distributed, decentralized manner. The social insects (e.g. ants and bees) provide the key inspiration that effective collective behaviour can be achieved without hierarchical organization and without access to global information. There exists a common assumption that these insects are individually quite simplistic, and that the capabilities of the units in a robot swarm should be similarly restricted [11]. Indeed, the predominant trend is to utilize coarse, range-limited sensors, such as infrared distance sensors and contact switches, and to insist on a purely reactive control architecture. Yet social insects such as honeybees employ a broad range of sensors, make heavy use of vision, and also employ long-term representations of sensory data [5]. We are interested in pursuing research in swarm robotics that utilizes vision. Most of our past work has focussed on object clustering and sorting [13, 14] but the visual sense is profoundly useful in other well-studied swarm robotic tasks such as aggregation [4], chain formation [12], self-assembly [7], and many others. The purpose of this paper is to articulate the need for an open, extensible robot platform to support swarm robotic research using vision and to propose one such platform.

It is important to highlight the need for a physical platform with associated software and simulation tools. Research in swarm robotics, like any sub-discipline of robotics, ultimately relies on hardware realization for validation. Simulation is an invaluable tool but subtle characteristics of the simulator can mask the deficiencies of a robot control strategy. In Brooks' influential proposal to re-shape research in AI and robotics, he suggested the following approach: "At each step we should build complete intelligent systems that we let loose in the real world with real sensing and real action. Anything less provides a candidate with which we can delude ourselves" [2]. To follow this advice we must test our ideas on swarms of real robots[1]. This presents a significant logistical challenge for most researchers. Leaving aside the robust and resilient robots that we hope to build one day, today's "real" robots are expensive, fragile, and much more cumbersome to work with than simulated robots. For these reasons, many papers in the field of swarm robotics present their results in simulation. In their review of approximately 60 publications Brambilla et al found that "slightly more than half of these publications presented results obtained only through simulations or models" [1]. Simulations are made more credible when a particular robot platform is modelled which is the approach we attempt here in presenting a robot platform and an associated model for simulation.

Robot platforms designed explicitly for swarm robotics include the Kilobot [10], the S-Bot [6], and most recently the GRITSBot [8]. The GRITSbot paper [8] provides a good review of other robots, many of which are not considered reasonable candidates because their web sites do not appear to be actively maintained. The Kilobot and GRITSbot are of similar size (≈3 cm) and cost ($50) but differ in their methods of locomotion and odometry. Both platforms can sense the range and coarse bearing of other robots or obstacles using a ring of IR distance sensors.

4.2 Characteristics

The following are desireable characteristics for visually-guided swarm robots:

Openness All design documents should be made freely available

Extensibility The design should support the addition of new features

Cost The overall cost must be significantly lower than commercially available platforms while still satisfying other criteria (the base e-puck model starts at $ 850 USD)

Size The size should be minimized while still satisfying the other criteria; The robot should be small enough to carry in one hand

Endurance At least 3 hours (to allow 2 full experimental runs in an 8-hour work day)

Puck Gripper To support experiments in object clustering, sorting, and the use of pucks for stigmergic communication

Odometry Wheel encoders or other sensors to support relative localization

Cameras **Omnidirectional** To support awareness of the range and bearing of other robots, visual navigation, and potential visual communication

Forward-facing To support detection of the position and type of pucks in front of the robot

The most general characteristics are listed first. Openness and extensibility are the most important traits. An open platform is much more likely to be adopted by a wide and supportive

[1]One could argue that to follow this advice fully, we must actually deploy robots in completely unstructured environments. Like many other researchers our approach falls short of this particular ideal and focuses instead on engineered environments.

user community. Such a community will naturally have disparate needs and will want to extend the platform in various ways. The design documents for the BuPiGo platform are available at `www.cs.mun.ca/~bots/bupigo`. Extensibility is supported by utilizing components that lend themselves to myriad uses. In particular, we use the Raspberry Pi 2 computer for high-level computing and the MinSegMega (an Arduino variant) for low-level computing. These products provide various easily-accessible and well-documented pins and connectors that open up a broad spectrum of possibilities in terms of sensors, actuators, and communication devices.

For swarm robotics, it is almost always advantageous to minimize cost and size. Robots such as the Kilobot and GRITSbot prioritize low cost and size. However, the inclusion of cameras and computational power sufficient to quickly process images tends to increase both—particularly, when we also want robots with battery capacity sufficient to operate for 3 hours at a time. One strategy for minimizing cost and size is to develop a custom printed circuit board (PCB) bringing all electronic components together in a tightly-packed configuration. However, not all labs are equipped to fabricate and populate PCBs. Also, custom PCBs require more care and expertise to debug and extend functionality.

4.3 Hardware

4.3.1 Components

The BuPiGo is composed of the following physical components:

Bubblescope and Raspberry Pi Camera Omnidirectional vision

Pixy Camera Forward-facing color segmented puck data

Raspberry Pi 2 High-level computing and connectivity (e.g. Wi-Fi)

MinSegMega (Arduino) Low-level computing

Servo and Gate Active puck retention

Batteries (2) Separate power for Raspberry Pi 2 and MinSegMega

Lego NXT Motors and Wheels (2) Locomotion

Ball Caster Third ground contact

3D Printed Base Support structure with C-shaped cavity for puck

Acrylic Platforms Support structure

The high-level computing functions tasked to the Raspberry Pi 2 include the execution of the overall robot control strategy and processing data from high-bandwidth sensors such as the Raspberry Pi camera. The low-level computing functions of the MinSegMega include interaction with the motors, odometry calculation, and forwarding the colour-segmented blobs extracted from the Pixy camera.

The extensible nature of the BuPiGo platforms implies that all of the above components could be replaced, with suitable modifications to the rest of the system. For example, the Raspberry Pi 2 computer could be replaced with a similar single-board computer (e.g. Minnowboard or Beaglebone

Black). The MinSegMega could be replaced by another Arduino variant. Some of our choices for these particular components were driven primarily by our specifications, yet others were driven by availability. For example, the Raspberry Pi 2 has the capability to boot from a micro-SD card. This is quite useful as it allows the software configuration to be repeated on another robot by simply copying the card's image from one robot to another. The Lego NXT motors were chosen due to availability. They could easily be replaced with other DC motors with built-in encoders.

Figure 1 shows two views of the BuPiGo model created in Sketchup. The only custom-built components are the 3D printed base and the two acrylic platforms. The files to create these components are available at `www.cs.mun.ca/~bots/bupigo`. We found that a professional-level 3D printer (Dimension 1200es) provided much more useful and consistent results than a more inexpensive desktop machine (Makerbot Replicator 2X).

4.4 Simulation

This section will discuss the computer model create for the BuPiGo robot. The model has been implemented in the robotic simulator V-REP. Both a static model, for visualization purposes, and a dynamic model, for interacting with the virtual environment, were created. In addition a ROS node [9] has been written which can be used to control a BuPiGo model in a V-REP simulation.

The Virtual Robot Experimentation Platform (or V-REP) [3] is a leading robot simulator developed by Coppelia Robotics. It allows for the development of 3d robot models which can be controlled individually through embedded scripts, ROS nodes, or a variety of other mechanisms. Once create a model can be placed in and interact with its virtual environment.

The static model of the robot is used purely for visualization purposes. Figure 2a shows an isometric view of the BuPiGo static model. While simulations are run the static model is visible to the user, and the underlying dynamic model is hidden. The dynamic model is used by V-REP when determining how the BuPiGo model interacts with the virtual world (ex. collisions, distance measurements, etc). Figure 2b shows an isometric view of the dynamic model. As the image in the figure shows the dynamic model looks nothing like the static model and is made up entire of "pure shapes" (rectangular prisms, cylinders, spheres, etc.). Pure shapes are used because they are more stable and faster during simulations. The dynamic model is broken up into 4 major components; main body, gripper, servo and puck gate, and finally the motors and wheels. The main body of the BuPiGo is represented by a single rectangular prism which extends from the rear of the chassis up to the base of the gripper attached to the front. Since the body of the BuPiGo is simple, and the only interaction it will have with the environment is through collisions this rough approximation is sufficient. The gripper located at the front of the robot is used to capture pucks for object clustering and sorting. It is modelled by three rectangular prisms, two along the sides and a thinner broader one on top. The two side components ensure any puck which enters the gripper remains in place as the robot rotates. The third solid prevents the puck from being dislodged during collisions. The BuPiGo includes a servo motor which actuates a mechanism for maintaining a puck within its gripper. This ensures a puck will remain in the gripper while the robot backs up. This gate mechanism will interact with captured pucks through collisions, and so it is included in the dynamic model of the robot. The drive wheels are represented in the dynamic model by two cylinders. The wheels interact with the floor of the virtual environment through collisions, therefore they were included in the dynamic model. Lastly, a small sphere was used to represent the rear caster wheel of the BuPiGo. Similar to the drive wheels the caster was required in the dynamic simulation for collision calculations with the floor.

A V-REP/ROS bridge has been created for controlling the BuPiGo model. A ROS node is started by an embedded Lua script attached to the BuPiGo model. Once the script has been called the model passes "handles" to the ROS node for each of the model's motors (left wheel, right wheel, and servo for puck gate), and sensors (gripper camera, omnidirectional camera). Before the node can begin controlling the robot a number of topics are setup. The node instructs the V-REP model to publish sensor readings to an appropriately named topic. After confirmation that the topics have been created the node subscribes to them. Next the node creates three (left- and right-wheel, and servo) topics to control the motors. The model is then requested to subscribe to these topics. After confirmation that the model has subscribed to the required topics the node is ready to assume control of the model. In order to easily accommodate a large number of BuPiGo models in a single simulation (i.e a swarm simulation) a unique identifier is randomly generated by each ROS node and appended to all topic names. This ensures each copy of the BuPiGo model subscribes to the correct motor commands and publishes its sensor data to the appropriate topic.

5 Conclusion

This report serves as the final required deliverable for the Harris Centre / MMSB sponsored project "Mitigating Spatial Interference in a Scalable Robotic Recycling System". The main body of work entailed by this project has been the development of an open and extensible robot platform that will facilitate experiments on robotic recycling. The next logical step is to return to the originally planned focus of mitigating spatial interference between robots. Much work remains before the practical implementation of these ideas to recycling in NL will be seen. However, we now have a set of robots that can be used to move this research closer to that goal.

6 References

[1] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo. Swarm robotics: A review from the swarm engineering perspective. *Swarm Intelligence*, 7(1):1–41, 2013.

[2] R. Brooks. Intelligence without representation. *Artificial Intelligence*, 47:139–159, 1991.

[3] M. Freese, S. Singh, F. Ozaki, and N. Matsuhira. Virtual robot experimentation platform v-rep: a versatile 3d robot simulator. In *Simulation, Modeling, and Programming for Autonomous Robots*, pages 51–62. Springer, 2010.

[4] M. Gauci, , J. Chen, W. Li, T. J. Dodd, and R. Groß. Self-organised aggregation without computation. *The International Journal of Robotics Research*, 33(9):1145–1161, 2014.

[5] R. Menzel and M. Giurfa. Dimensions of cognition in an insect, the honeybee. *Behavioral and Cognitive Neuroscience Reviews*, 5(1):24–40, 2006.

[6] F. Mondada, A. Guignard, M. Bonani, D. Bär, M. Lauria, and D. Floreano. Swarm-bot: From concept to implementation. In *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on*, volume 2, pages 1626–1631. IEEE, 2003.

[7] R. O'Grady, A. L. Christensen, and M. Dorigo. Swarmorph: multirobot morphogenesis using directional self-assembly. *Robotics, IEEE Transactions on*, 25(3):738–743, 2009.

[8] D. Pickem, M. Lee, and M. Egerstedt. The gritsbot in its natural habitat-a multi-robot testbed. In *IEEE ICRA*. (submitted).

[9] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng. ROS: An open-source robot operating system. In *ICRA workshop on open source software*, 2009.

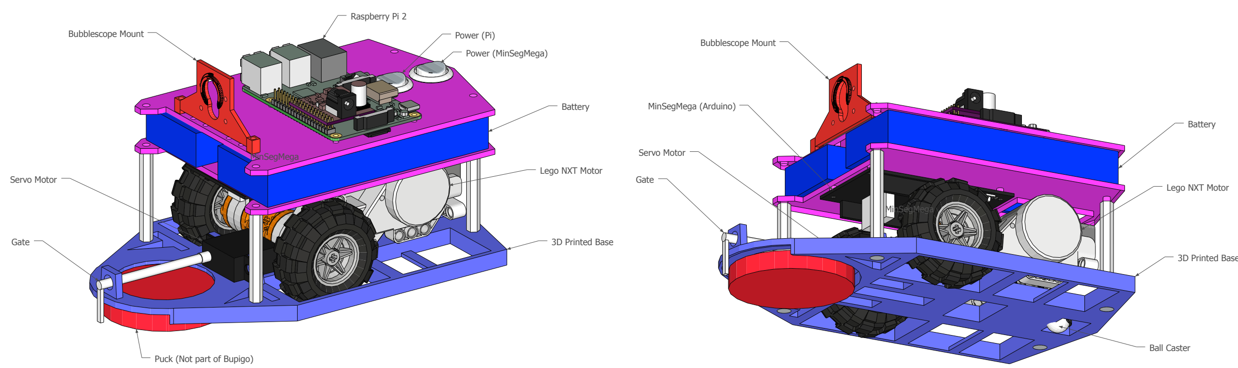
[10] M. Rubenstein, C. Ahler, and R. Nagpal. Kilobot: A low cost scalable robot system for collective behaviors. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 3293–3298. IEEE, 2012.

[11] A. J. Sharkey. Swarm robotics and minimalism. *Connection Science*, 19(3):245–260, 2007.

[12] V. Sperati, V. Trianni, and S. Nolfi. Self-organised path formation in a swarm of robots. *Swarm Intelligence*, 5:97–119, 2011.

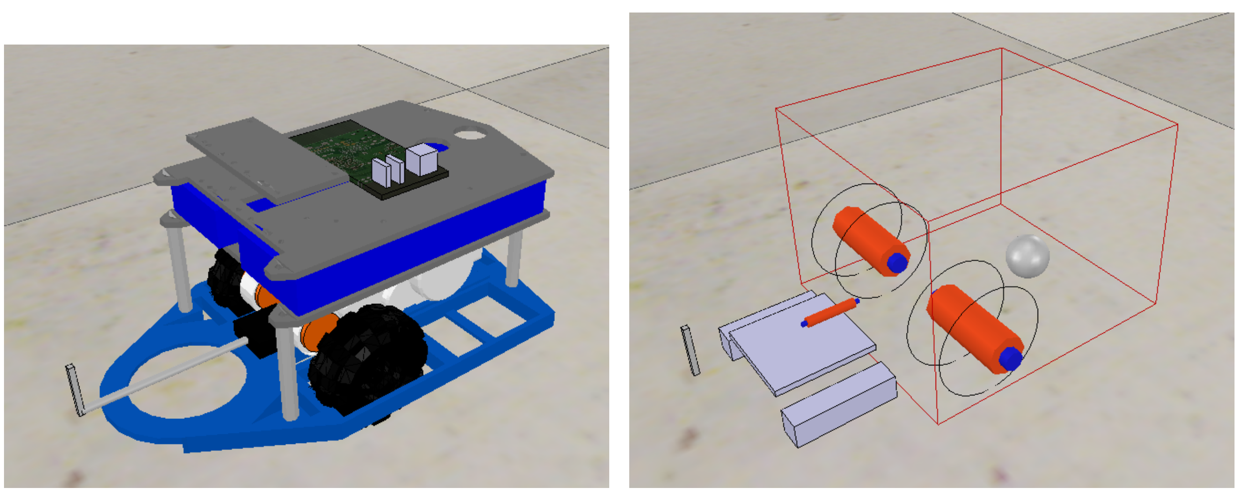
[13] A. Vardy. Accelerated patch sorting by a robotic swarm. In *Canadian Conference on Computer and Robot Vision*, pages 314–321. IEEE Xplore, 2012.

[14] A. Vardy, G. Vorobyev, and W. Banzhaf. Cache consensus: Rapid object sorting by a robotic swarm. *Swarm Intelligence*, 8(1):61–87, 2014.



(a) (b)

Figure 1: Views of the 3D model of the BuPiGo in Sketchup from above (a) and below (b). Note that the 3D model does not include either the Pixy camera or the Bubblescope mirror/Raspberry Pi camera assembly. Also, the colours of some components are chosen for visual distinction and are not reflected in the physical robot.



(a) Static Model (b) Dynamic Model

Figure 2: Views of the BuPiGo model created in V-REP. Note that this model does not include either the Pixy camera or the Bubblescope mirror/Raspberry Pi camera assembly.