AUTONOMOUS NAVIGATION OF A 5 GRAM CRAWLING MILLIROBOT IN A COMPLEX ENVIRONMENT

ALLISON MATHIS, JARED RUSSELL, TOM MOORE,
JASON COHEN, BRIAN SATTERFIELD
Lockheed Martin Advanced Technology Laboratories
Cherry Hill, NJ, 08002, USA

NICHOLAS KOHUT, XIAO-YU FU, RONALD S. FEARING
University of California, Berkeley
Berkeley, CA, 94720, USA

We present the Medic robot: a novel 5-gram autonomous crawling and climbing millirobot designed to navigate through dark, cluttered, confined and electrified environments. Due to payload constraints, all sensing is performed by a single low-resolution camera and processing is split between an onboard unit and a base station. We present an integrated object detector, localizer, and planner that have been successfully demonstrated to enable the millirobot to determine its position within the environment, plan and navigate the most efficient path between user-specified start and goal positions, and recover from stuck conditions.

1. Introduction

Millirobotics offers a unique solution to the problem of diagnosis and repair of large electronic systems where human inspection is often costly, impractical, or error-prone. In many cases, simply gaining access to electronics installed in satellites or aircraft is enormously expensive; a resident millirobot equipped with basic circuitry diagnosis capabilities is an attractive low-cost alternative. Our research investigated whether a resident millirobot could successfully autonomously navigate through a powered electronic environment to specified components of interest. We envision that future work will develop actual diagnosis and intervention capabilities. Circuit boards are generally cluttered with surface-mount components that pose significant navigational challenges for a robot of this size (5 grams, 55mm x 35mm x 18mm). Our solution uses an onboard video sensor to perform visual landmark-based localization and to estimate egomotion using optical flow techniques.

Successful navigation at this scale poses several unique challenges. Operation on circuit boards requires a high degree of precision in order to avoid getting the vehicle stuck on surface-mount components and pins. Due to size

and weight constraints, our crawling millirobot is mechanically compliant, leading to difficult-to-predict interactions with varying surfaces and obstacle configurations. Moreover, the inherent payload constraints limit the sensors that may be carried. Consequently only a single CMOS camera was employed during the course of this research.

To address these challenges, we designed a novel four legged platform based on previously described methods of manufacturing [1] that could carry a modest payload and mount small obstacles with minimal force requirements [2]. Image-capture and settings, as well as low-level actuator control are processed onboard, but image processing and high-level planning are performed offboard via an IEEE 802.15.4 link to a PC. Our approach to detecting landmarks was adapted from a method developed by the leaders of the PASCAL Visual Object Classification challenge [3]. Due to observed variability in the steps taken by the robot and the uncertainty inherent in estimating position from visual landmarks, we use a probabilistic approach to localization using particle filters. Path planning was performed via a hybrid planner that combined A* and Field D* methods. Additionally, methods to detect and recover from "stuck" conditions were implemented.

2. Hardware Overview

The details of the construction and actuation were originally presented in [2], however the following is a high level summary to aid understanding of the overall system. The Medic robot (Figure 1) measures 55 mm long, 35 mm wide at the legs, 25 mm wide at the hips, and is 18 mm high. Its total mass is 5 grams,

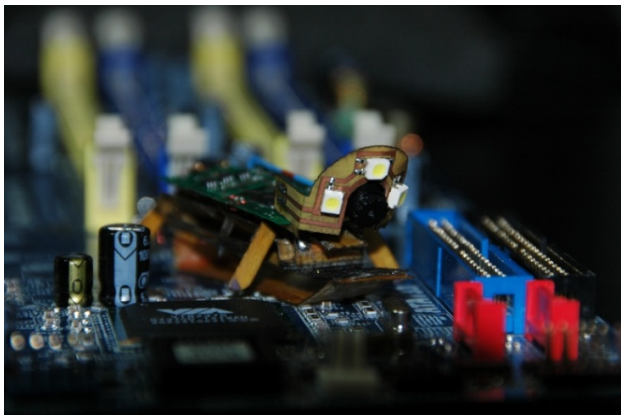

Figure 1. MEDIC robot in an electronic environment

including the onboard battery. It is constructed primarily of S2-glass [1] with integrated circuitry for mobility control. Mobility is enabled by four legs (each a member of a connected pair on either side of the chassis) that are actuated by coiled shape memory alloy (SMA) springs. Forward, reverse, and turn-in-place maneuvers are enabled by sequencing the actuation of the SMAs. It has been designed and built to traverse horizontally mounted computer motherboards and is capable of mounting obstacles up to 5.5 mm high. For further information on the physical parameters and construction method used for rapid fabrication see [2].

A consequence of the robot's lightweight design is that significant uncertainty was observed in the motion achieved by each type of step. This is due to two factors: the minimal mass of the Medic robot results in compliance in structural members when the actuators are exerting force, and the numerous unsensed obstacles (such as individual pins and traces) on PCBs result in unpredictable interactions between the robot's legs and the environment. We characterized the motion achieved by each step over a variety of surface conditions with normal distributions (Table 1).

Table 1. Step motion characterization

Step Type	Trans. Mean (mm)	Trans. Var. (mm^2)	Turn Mean (rad)	Turn Variance (rad^2)
Fwd	6.0	0.25	0	0.048
Rev	-4.0	0.25	0	0.048
Turn	2.0	0.25	+/- 0.15	0.048

3. Localization

The Medic navigation system combines a labeled overhead map of the circuit board (with only major components labeled, e.g., capacitors and slots but not individual pins or traces) with a library of visual appearance features of specific electronic component categories. Localizing the millirobot within its environment using only the 160x100 pixel onboard grayscale camera and limited a priori maps requires the fusion of multiple pieces of evidence about robot pose. A sequential importance resampling (SIR) particle filter approach was chosen, as particle filters can easily represent non-Gaussian, multi-modal probability distributions and can be used for fusion simply by applying multiple

successive “observe” functions to the distribution sequentially [4]. All processing was performed off-board via a ZigBee link to a laptop PC.

The millirobot’s component detection algorithm was derived from successful techniques used for higher-resolution classification [6, 7]. Invariant features formed the lowest level of the detector [8]. Feature clustering via a Pyramid Match Kernel with spatial pooling provided compact feature representations that were used to perform convex optimization [9]. A support vector machine was used to solve the optimization problem and find decision boundaries between component types based on hand-annotated ground truth imagery.

Detected components were integrated into the particle filter by comparing what was found with what each particle expected to see based on the annotated overhead circuit map. Ray tracing was conducted from each component to each particle, removing candidate components that were out of the particle’s field of view or occluded. An exponential reward function was used to re-weight the particles according to the amount of overlap between the set of detected and expected components and their relative locations within the camera frame. This approach minimized the effect of spurious detections, as particles that confirmed the existence of multiple components were more heavily weighted than those that confirmed only a few or a single component. False negatives were not penalized due to frequent occlusions and poor image quality, allowing the detectors to be tuned for specificity rather than sensitivity. Table 2 summarizes the performance of the component detection algorithm for each component type.

Table 2. Component detection and classification performance over a set of test imagery captured in an electronics environment (64 images, each subdivided into 25 regions).

Component Type	# of Regions	Detection Rate	Avg. False Alarms Per Image
No Component	1108	98.1%	0.76
Buzzer	3	100.0%	0
Capacitor	54	88.9%	0.03
Fan Header	10	90.0%	0
Heat Sink	63	92.1%	0.05
IDE Connector	17	52.9%	0
Inductor	14	92.9%	0
Molex Power	9	88.9%	0.02
PCI	272	94.1%	0.52
PCI-Express	58	82.8%	0

4. Planning

The density of the obstacles in the robot's environment make traditional grid-based planners ill-suited for path generation. Such planners typically discretize the world into a 2D grid whose cells represent some small region of the environment. Planners based on the ubiquitous A* algorithm [10] find optimal paths given a map and some cost function. However, when A* plans across a 2D grid, the generated paths consist of turns that are multiples of $\pi / 4$ radians. The Medic robot has highly constrained battery life, so paths that consist of superfluous turns severely limit the operational range. Furthermore, the robot typically exhibits 2 mm of translation while turning, making turn-in-place maneuvers impossible and all turns subject to more uncertainty than pure forward/backward translation.

A solution to this problem is to use a planner that allows for a continuous action space. A planner based on the Field D* [11] algorithm was implemented. Field D* generates a continuous cost field by using discrete cost maps and bilinear interpolation between nodes. The cost field can be traversed using gradient descent until the goal selection of steps available to the Medic robot necessitated using a discretized search algorithm over this continuous cost map location is reached. Our solution was to apply a hybrid planning approach wherein a Field D* cost field would be generated, and a modified version of A* descended the gradient while considering the robot's repertoire of steps.

When the Field D* cost field had been generated, A* was run in the opposing direction, i.e., toward the goal. Instead of using a 2D grid and a canonical eight-connected A* search graph, the expansion of nodes was carried out with respect to the robot's action space. From a given A* node (which represented a real-valued location), the neighboring nodes would be the predicted destination locations of each of the robot's actions. The resulting paths were optimal with respect to the cost field given the robot's mobility constraints.

An experiment was carried out in simulation in which 64 random start/goal location pairs were generated. Both a standard A* planner and our hybrid planner were used to generate paths for each pair, the first averaged a total path length of 1.52e+04 nodes compromising 70,331 steps while the second averaged 1.63e+04 nodes and only 5,066 steps. The 8-connected A* planner generated paths that were slightly shorter, but at the expense of many sharp turns within the paths, is evidenced by the order-of-magnitude difference in the number of steps required by the A* planner.

5. Detection and Recovery from "Stuck" Conditions

Even with optimal plans and accurate localization, the robot would often become stuck, typically due to one of the legs getting caught on an unsensed and unmodeled surface-mount component or pin during SMA actuation. Errors of a few millimeters in localization often made the difference between completely clearing an obstacle and "stubbing a toe," resulting in undesired vehicle behavior such as yawing around the obstacle rather than translating forward as commanded. The compliance of the legs and body of the Medic robot made accurately predicting these situations and necessitates a new approach.

In order to identify and overcome these conditions, we implemented an optical flow algorithm to compare camera frames captured before and after each step. The SURF descriptor [5] was computed on each frame, and feature points were correlated between successive frames using a RANSAC algorithm to estimate the robot's egomotion. We developed a model of expected observed motion (direction and magnitude) for each type of step, and compared the results against these thresholds to determine whether nominal movement was observed. This heuristic successfully identified situations where a leg had become ensnared with an obstacle 100% of the time during testing. Extensions to the extrication algorithms that will attempt to reason about the likely cause of entanglement through observation of motion are planned for future work.

6. Results

The Medic autonomous navigation system has been demonstrated successfully during "A-to-B" type tests on a powered, enclosed PC motherboard. Typical runs consisted of 40-60 steps (with an image taken before each), and lasted 10-20 minutes. Due to the entanglement hazards mentioned previously, many trials (approximately 2/3) ended with the robot irreparably entangled on an obstacle. However, when entanglement was completely averted (approximately 1/6 of trials) or correctly remedied via our heuristic process (approximately 1/6 of trials), the Medic robot did achieve its goal location within 1 cm of the commanded goal position.

The series of images in Figure 2 represents a typical successful trial, shown as captured by an overhead camera over a series of steps through the test environment (these are best viewed in color). In each instance, the image on the left shows the actual millirobot in the test environment, while the right is the view from the onboard camera. The light blue circle encompasses the possible locations within one standard deviation of the most likely estimate, which is represented by the yellow dot in the center. The tail coming off both the

maximum likelihood estimate (MLE) and the goal represents the orientation of the millirobot.

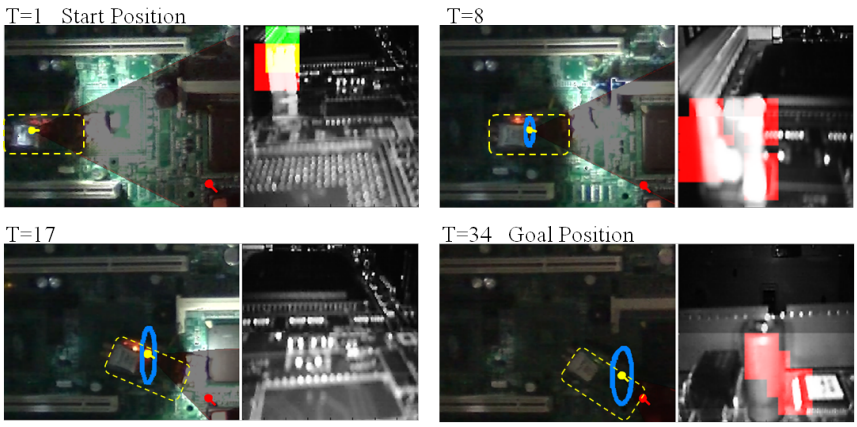

Figure 2. The robot (yellow dashes), its estimated position (yellow dot), the camera data and object detection results at four instances over the course of a run. The objects detected are shown superimposed over the images collected at that moment. The robot's FOV is the shaded triangle, the goal location is represented by the red dot. Capacitors have a red mask, power connectors have green, and PCI slots are shown in blue.

7. Conclusions and Future Work

This paper has presented the perception and autonomous navigation design for the millirobot Medic. The use of landmark-based visual navigation for millirobots in the 5-gram class is novel and useful due to the constraints for size, weight and power. The Medic robot is capable of autonomous navigation in rough terrain and of detecting and sometimes removing itself from stuck conditions. The performance of the perception system is comparable to the state-of-the-art in higher resolution object detection despite challenges in resolution, bit depth and illumination. The autonomous system reliably guided the millirobot to within 1 cm of the goal during live testing. We believe this represents the first visually guided millirobot of less than 6 grams capable of autonomous navigation through rough terrain.

The current system could be improved to increase the performance and robustness of navigation and perception. These improvements would include an improved localization method that performed simultaneous global and local state estimation, and stuck-detection reasoning that would determine how the platform was stuck and employ the correct action for extraction to increase

robustness. Replacing the current camera with a newer model would significantly increase the performance of the perception system by providing higher-resolution and improved light sensitivity. Additionally, alternate sensing modalities, such as tactile or infrared imaging, are worth investigation.

References

1. R.J. Wood, S. Avadhanula, R. Sahai, E. Steltz, and R.S. Fearing, "Microrobot design using fiber reinforced composites," Journal of Mechanical Design, vol. 130, no. 5, p. 052304, 2008. [Online]. Available: http://link.aip.org/link/?JMD/130/052304/1
2. N. Kohut, A. Hoover, K. Ma, S. Baek, and R.S. Fearing, "MEDIC: A Legged Millirobot Utilizing Novel Obstacle Traversal," IEEE Int. Conf. on Robotics and Automation, Shanghai May 2011.
3. PASCAL Challenge: http://pascallin.ecs.soton.ac.uk/challenges/ VOC/.
4. N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear-non-Gaussian Bayesian state estimation," IEEE Proceeding-F, Vol.140 No.2 April 1993.
5. H. Bay, A. Ess, T. Tuytelaars, and L. Gool, "SURF: Speeded Up Robust Features," Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346-359, 2008.
6. K. Yu, T. Zhang, and Y. Gong, "Nonlinear Learning using Local Coordinate Coding," in Neural Information Processing Systems, 2009. NIPS 2009.
7. X. Zhou, K. Yu, T. Zhang, and T. Huang, "Image Classification using Super-Vector Coding of Local Image Descriptors," European Conference on Computer Vision, 2010. ECCV 2010.
8. D.G. Lowe, SIFT flow: dense correspondence across difference scenes. Proc. ICCV 1999, Kerkyra, Greece, pp. 1150-1157.
9. K. Grauman and T. Darrell. "The Pyramid Match Kernel: Discriminative Classification with Sets of Image Features," IEEE International Conference on Computer Vision (ICCV), Beijing, China, October 2005.
10. P.E. Hart, N.J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," Systems Science and Cybernetics, IEEE Transactions on, vol. 4, no. 2, pp. 100-107, 1968.
11. Nabbe and M. Hebert, "Extending the Path-Planning Horizon," International Journal of Robotics Research, pp. 997-1024.