Proceedings of the 2001 IEEE/RSJ
International Conference on Intelligent Robots and Systems
Maui, Hawaii, USA, Oct. 29 – Nov. 03, 2001

Using Orthogonal Visual Servoing Errors for Classifying Terrain

Richard M. Voyles
Computer Science
University of Minnesota
Minneapolis, MN
voyles@cs.umn.edu

Amy C. Larson
Computer Science
University of Minnesota
Minneapolis, MN
larson@cs.umn.edu

K. Berk Yesin
Mechanical Engineering
University of Minnesota
Minneapolis, MN
kyesin1@me.umn.edu

Brad Nelson
Mechanical Engineering
University of Minnesota
Minneapolis, MN
nelson@me.umn.edu

ABSTRACT

A novel, centimeter-scale crawling robot has been developed to address applications in surveillance, search-and-rescue, and planetary exploration. This places constraints on size and durability that minimizes the mechanism. As a result, a dual-use design employing two arms for both manipulation and locomotion was conceived. In a complementary fashion, this paper investigates the dual-use of visual servoing error. Visual servoing can be used by a mobile robot for homing and tracking. But because ground-based mobile robots are inherently planar, the control methodology (steering) is one-dimensional. The two-dimensional nature of image-based servoing leaves additional information content to be used in other contexts. We explore this information in the context of classifying terrain conditions. An outline for gait adaptation based on this is suggested for future work.

INTRODUCTION

The Center for Distributed Robotics at the University of Minnesota conducts research in a variety of areas in embodied distributed agents. Although heterogeneity is a primary theme of the center, a common trait among its robots is small size. The subject of this paper is one of these small robots called the *TerminatorBot.* The TerminatorBot is a mesoscale mobile manipulator that uses it arms for both manipulation and locomotion [15].

This dual-use design approach was adopted to add manipulation capability to our previous *Scout* mobile robots [1]. These robots have hard form-factor constraints because they employ ballistic locomotion for initial gross positioning. The robots are intended to be launched or thrown to the vicinity of their targets for fast traversal with minimal onboard power consumption. After initial deployment, they must rely on onboard resources to position themselves and surmount obstacles.

The addition of arms for manipulation provides the capability to precisely position payloads, manipulate the environment for camouflage (pulling objects over itself), surmounting obstacles in low headroom environments, and digging into soft earth. In turn, these capabilities open up new niche application scenarios in search-and-rescue, surveillance, and space exploration.

This paper explores the use of visual servoing errors as a metric for deducing terrain conditions. The motion of tracked features in a video scene of a body-mounted camera result from the movement of the robot (equivalent to an eye-in-hand system). So the error vector of a feature from its goal location on the image plane provides two degrees of freedom. But a correctional heading for a mobile robot along a manifold consists of only one degree of freedom. Therefore, there is additional information available for analysis. We use this information to crudely classify terrain properties and eventually hope to use it to optimize parameters of the locomotion gaits.

PRIOR WORK

Mobile manipulation is an area of research that has not been extensively addressed in the robotics community. Many have placed manipulators on mobile robots (in fact, Nomadic Technologies had a commercial offering with a PUMA 560 manipulator), but they have generally been treated disjointly. Sandia, for example, has put Schilling arms on a variety of platforms for teleoperation in hazardous environments (e.g. [2]). Carriker et al integrated the path planning of low-DoF subsystems, but motion operations and design for each were treated separately [3]. Khatib has done significant work in integrating the motion control of arms and mobile bases through the Operational Space formulation [4], but has not performed visual servoing (an important goal of the TerminatorBot) nor are the mechanisms dual-use. Brachiation robots, which use arms for locomotion by swinging like a gibbon, have also received some study (e.g. [5]), but current mechanisms are incapable of manipulation.

A few robots have been considered with dual use design. SM^2 and DM^2 at Carnegie Mellon ([6] and [7], respectively) and PolyPod/PolyBot at Stanford/Xerox ([8]) are notable examples. SM^2 and DM^2 are symmetric, biologically inspired inch-worm-like robots with grippers at each end. The robots are designed to walk around the outside of the space station to perform repair and inspection tasks. PolyPod is a modular serpentine manipulator of many similar joint modules designed with both manipulation and locomotion in mind. "Platonic Beasts" [9] were developed by Pai et al with suggestions of dual-use limbs, but they were primarily studied for their robust locomotion capabilities.

0-7803-6612-3/01/$10.00©2001 IEEE

Finally, Mason et al have developed the "Mobipulator" [10] for extensive study of desktop mobile manipulation, but this robot possesses only differential drive wheels and no limbs for dextrous manipulation.

Visual servoing, on the other hand, has been extensively researched in the robotics literature for tracking and control [17][18][19][20]. Here, we use visual servoing as a tool to extract information of interest on terrain conditions.

TARGET APPLICATIONS

The Scout robot has reconfigurable payloads and both rolling and hopping modes of locomotion, so it is quite capable. Rolling is fairly power efficient and hopping enables it to overcome obstacles, which are common for a robot only 40 mm tall. Unfortunately, while the hopping is required for practical mobility, it is rather time and power inefficient due to the inefficiency of the winch mechanism. Navigational certainty is also very low for hopping. The distance and direction of travel is poorly known and orientation in the plane upon landing is completely random.

Its small size and stealth are useful for military and civilian uses. Equipped with a camera or microphone Scouts could be used in search-and-rescue operations following natural disasters (e.g. earthquakes) or terrorist actions (e.g. Oklahoma City bombing). There is also potential interest from civilian SWAT teams in hostage situations and police standoffs. These are natural military uses, as well, particularly in urban warfare environments that involve civilians. Surveillance robots of this size could be carried and deployed by warfighters, keeping the warfighters out of the line of fire and minimizing the risk of civilian casualties in the "heat of the moment."

While the Scouts' dual locomotion modes are necessary to achieve many of these missions in real environments, there are concerns they may be inadequate for particular scenarios, hence the investigation of the TerminatorBot as an alternate design. For example, mesoscale robots would be most useful in search-and-rescue operations in which the damage is too severe and constricting to send in dogs (which arguably will be superior to robots in sensing for the near future). But large amounts of rubble within extremely cramped spaces may thwart both locomotion modes of the Scouts (too much rubble to roll, too little headroom to hop - see Figure 1). A crawling robot such as TerminatorBot could fill this niche in which available headroom is, on average, just a few times the rubble size.

In surveillance tasks, it is desirable for the robot to conceal itself. The Scouts will only be able to make use of existing open spaces such as underneath furniture. A robot with manipulators could actually pull objects over itself, creating its own cover and enhancing its stealth. A miniature, telescoping pan/tilt unit has been developed to facilitate such stealthy surveillance, too [11].

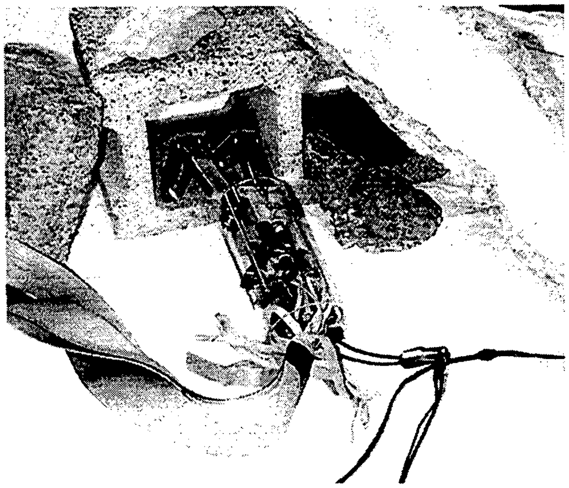

Figure 1: TerminatorBot in mock search-and-rescue scenario. (Robot was joint-level teleoperated for this photo.)

The idea of many small robots amassing a useful charge from small, insignificant explosives has been suggested by researchers in a number of scenarios. The main problem with this idea is that the efficiency of explosives is highly dependent on their placement. A bunch of mobile robots with no ability to manipulate would amass a rather inefficient bomb. Just one or two robots with the ability to locomote *and* manipulate could carefully place the charges, demanding many fewer trips to achieve a given objective.

Finally, in many of these scenarios, the ability to dig or burrow in light soils is beneficial. This could provide camouflage during surveillance, additional access during search-and-rescue, and an alternate detonation means during de-mining operations.

LOCOMOTION GAITS AND MECHANISM DESIGN

The mechanism and design rationale that led to the TerminatorBot, as well as example novel locomotion gaits, are described in [15] and [16]. In brief, the robot consists of two 3-degree-of-freedom (DoF) arms that can fully stow inside the 75mm diameter body. The tips of the arms are claw-shaped and allow the robot to drag itself along the ground using a variety of gaits.

This paper focuses on one gait: the "swimming" gait. In this gait, the robot reaches forward with both arms, lifts the body, and then pulls itself forward by pushing back on the ground below (Figures 2 and 3). The swimming gait is so named because of its similarity to two-armed swimming strokes (the butterfly, for example). This gait is used for the study of cyclic, orthogonal motion errors induced in the visual field of a body-mounted camera.

VISUAL SERVOING

A tutorial on visual servoing can be found in [21], but the basic idea is to fixate on a visual feature in the environment and servo the mechanism to hold constant the feature's position in the image plane. In the case of navigation, a distant goal point in the image would be fixated and the servoing algorithm would try to keep that feature in the center of the field of view as the robot moved toward it. This would effectively produce a "homing" behavior (given some simplifying assumptions such as no obstructions).

An important thing to note is that ground-based mobile robots are effectively planar devices. They can only move on a 2-D surface. The surface may have complex 3-dimensional shape, but, in most situations, the navigation of the robot is limited to steering corrections (right and left) as the robot moves forward.

Effectively, the information to extract for the described homing behavior is one-dimensional. It may involve a complex transformation if the 3-D structure of the surface is complex, but the output is fundamentally one-dimensional. However, the image plane provides two orthogonal degrees of freedom (three counting rotation in the plane using multiple features). Once the servoing error is orthogonalized, we investigate making use of the information orthogonal to the navigational degree-of-freedom. For our cyclic locomotion gaits, the "bounce" of the camera orthogonal to the surface provides crude information on terrain conditions when normalized with respect to the gait parameters.

Our visual servoing algorithm employs sum of squared differences (SSD) template matching to locate each feature in every video frame within a small window of the feature's last location [11]. For now, features are manually selected. The SSD feature tracker blends the best-match image patch at each cycle with the stored feature template to accommodate changes in lighting and the projection of the feature.

A miniature pop-up pan/tilt camera mechanism has been developed for image capture [11]. This allows the camera to stow inside the body of the TerminatorBot for ballistic deployment and then pop up outside the protective shell for inspection or navigation. This miniature device is shown in Figure 2. At this preliminary stage of our research, we are not able to separate the motion of the camera from the motion of the robot body (other than quasi-statically, as in [11]) so we used a fixed camera strapped to the robot as shown in Figure 3. In fact, this has important ramifications for bounce normalization as we assume a known focus of expansion to compensate for the robot moving toward the features. So we fix the camera optical axis to be parallel with the robot's forward axis of travel.

Figure 2: The miniature pan/tilt camera unit.

EXPERIMENTS

For these initial experiments we fixed the camera gaze straight ahead, manually selected targets for tracking, and commanded the robot to move forward using the swimming gait in open-loop mode. Unfortunately, the MIPS-based, onboard CPU is still under development, so the robot is constrained by a bulky tether to a desktop PC. This severely limits the range of motion of the robot and the length of trials we can perform.

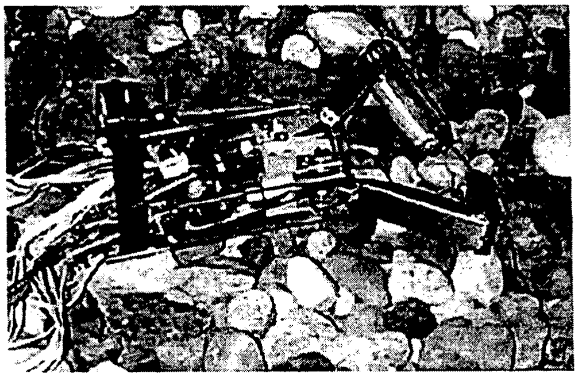

Figure 3: TerminatorBot crawling across rocks with the camera on its back

The camera has a 60-degree field of view, which introduces significant perspective distortion. But the beauty of this approach is that it only depends on quivers of rays emanating from the features and is not profoundly dependent on the actual 3-D location of the feature with respect to the camera frame. We select two features that project onto the image plane via the pinhole camera model. The projection of two points establishes a line segment from which we extract orientation and the location of the centroid (on the image plane). Again, we don't care about the actual z depth to the features as long as our assumption of constant reference angle of the ray is not violated. (With features as close as 1 meter, the assumption is violated and the impact is noticeable, but not show-stopping.)

The motion model assumes only three degrees of freedom: all three rotations of the camera. We ignore translation because we are only interested in the relative angular orientation of the camera (and therefore, the robot body) with respect to the feature rays. (The features are assumed fixed. Tracking multiple features and ensuring geometrical consistency could identify errant or moving features.)

Roll around the camera optical axis results from the cylindrical shape of the robot. If both limbs are not touching the supporting surface (which often happens during retract-and-placement phases of the gait) the robot can roll around the major axis, introducing a misalignment between the camera frame and the plane of the surface patch beneath the robot. (This surface patch is assumed locally planar for the several centimeters required for a gait cycle.) Roll-induced motion of the features is compensated by the assumption of features (rays) fixed with respect to the supporting surface patch. This is shown in Figures 4 and 5.

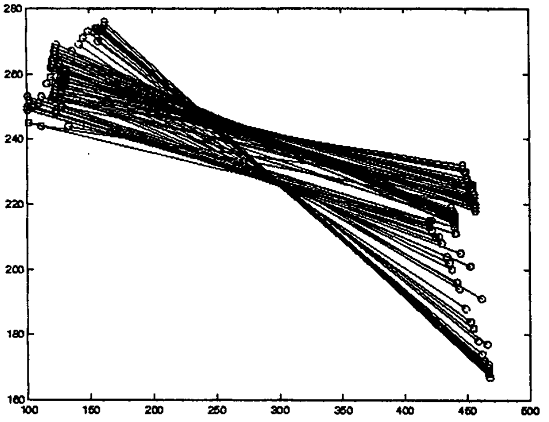

Figure 4: Uncompensated segment orientations in the image plane (many superimposed).

Since we're not yet willing to toss the TerminatorBot into a sandbox, we created four simulated surfaces to crawl upon. One was a hard surface (various pieces of industrial carpet over plywood), the second was foam rubber (two different pieces tested with approximately the same stiffness), the third was loose styrofoam packing peanuts we hoped would simulate sand or pebbles, and, finally, we created a bed of smooth river rock of various diameters (around 2 - 6 cm). We have only run a few gait cycles across each surface to-date, but representative samples of the decoupled "gait bounce" for three of the surfaces appear in Figure 6.

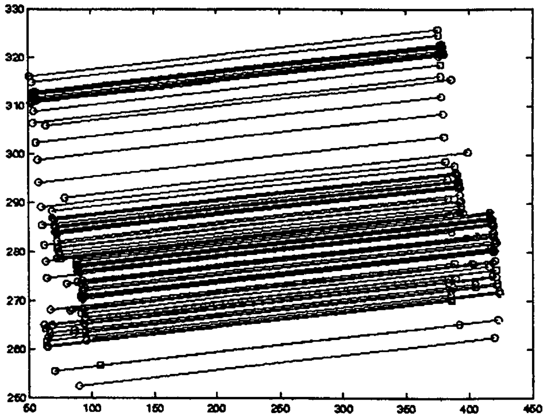

Figure 5: Roll-compensated segments corresponding to the above.

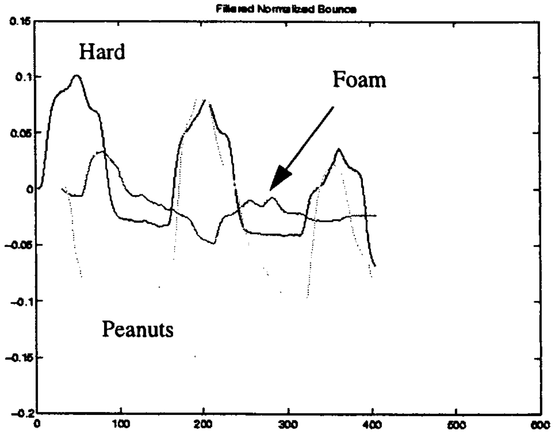

Figure 6: Decoupled gait bounce, indicating firmness of surface below.

The downward drift of the above plots is due to perspective distortion. Some features were as close as 1 meter away so the constant ray assumption was violated to a small degree. Since the features were all above the robot as it approached, it appears as if the camera is tilting increasingly downward. We crudely compensate for this by fitting a line to the data and subtracting out the slope. But for features near the robot this linear assumption is inaccurate.

The results of the peanuts seem somewhat surprising. The lift angle starts in the opposite direction. This general trend was repeatable and, upon investigation, appears to be caused by the TerminatorBot "digging itself in" as it begins to move its arms. The reason is that, without onboard CPU and batteries, the robot is front-heavy. Movement of the arms forward exacerbates the problem as well as jostling the peanuts which shift, allowing the front of the robot to sink deeper. It then jostles to tilt angles just as high as on the hard surface. This is due to the pivot point in the back sinking in as the peanuts move out from under it. The claws are also able to grasp the peanuts themselves to get reasonable traction. The conclusion is that packing peanuts are not a good simulation of sand. Furthermore, we think proprioceptive sensing of joint torques will help in the interpretation of these conditions since much of the movement appears to be due to motion of the surrounding surface.

To classify the terrain we perform an FFT (fast Fourier transform) on each gait cycle individually and compare it to representative FFT vectors for each terrain type. (Since we command the gait cycle, the beginning of each gait cycle is known.) These representative FFT vectors are just averages of several examples over each surface. After gathering 18 gait cycles and training on 10 randomly selected samples (ensuring at least 2 samples from each class), the remaining 8 samples were classified according to Table 1.

Table 1: Gait Cycle Classification

Surface	*Correctly Classified*	*Total Samples*	*Percent Correct*
hard carpet	3	3	100%
foam	1	2	50%
peanuts	1	1	100%
rocks	1	2	50%

Considering these values are on a single gait cycle (one "step"), the accuracy is pretty good. Although there are few datapoints, they were gathered over many days, so they hold some promise. Also, our goal is to use this information over several gait cycles to develop a better picture of the terrain, but our current tether prevents extensive excursions. We have completed a design revision to the initial TerminatorBot prototype and are building the second generation now with greater on-board computation.

Because we assume we know the focus of expansion of the visual scene (it is assumed to be at the center of the image plane), scale distortion can be introduced if the optical axis is not aligned with the axis of travel. However, the FFT extracts the shape of the waveform (through its frequency components) and is not affected by magnitude. We only examine the first 6 elements of the FFT result and normalize this vector to unit magnitude (zeroing high-frequency noise).

GAIT ADAPTATION

The focus of our adaptation studies is to develop intelligent agents that can detect changes in the performance of the locomotion gaits (and, eventually, manipulation "gaits") and hypothesize-and-test modifications to improve performance. The agents will learn their own metrics of performance and how to tune them.

Our implementation of gait adaptation is based on the Port-Based Adaptable Agent Architecture (PB3A) from Carnegie Mellon University [12]. A Port-Based Agent (PBA) consists of a collection of Port-Based Modules (PBM) which are essentially port automata (much like the Port-Based Objects of the Chimera real-time operating system [13]). The architecture provides a standard wrapper supporting periodic and aperiodic execution, intercommunication, parametrization, and graphical display of PBMs.

The approach we are taking to implement a gait agent is to implement a series of underlying gait primitives as PBMs. These gait primitives will include tracking a joint-space trajectory segment, tracking a world-space trajectory segment, a guarded move, etc. A gait, then, will actually be a cyclic trajectory through the primitive space and a gait agent will maintain this cyclic behavior with appropriate additional PBMs for tuning the parameters of the primitives and measuring performance.

In many ways this resembles work on central pattern generators (CPG) [14], which are rhythmic, low-motor behaviors found in biological systems. But CPGs are cyclic open-loop actuation commands. The "CPG" in our gait agents would cycle through primitive space, not actuation potential space. This allows incorporation of both open-loop and closed-loop primitives at different stages of the cycle. Eventually, we hope this will allow the smooth transition from intensive closed-loop monitoring of a gait to predominantly open-loop control of a gait as the learning progresses. This phenomenon is generally observed in biological systems as an entity becomes more "skilled."

SUMMARY

The *TerminatorBot* is a mesoscale mobile robot that employs dual-use limbs for both locomotion and manipulation to conserve space and functionality in a highly resource-constrained application. We feel it is suitable for reconnaissance, search-and-rescue, and planetary exploration applications.

We briefly described the mechanism, including a miniature pan/tilt camera mounting and an application of orthogonal decoupling of visual servoing errors. While we did not present results of visual servoing for navigation, that is the primary motivation for visual servoing on this platform. However, this paper explored a novel look at a by-product of the visual servoing that can be used to estimate terrain conditions for adaptation of the cyclic gaits. We found the use of the orthogonal tracking error could result in discriminating between different terrain samples given our preliminary data.

The ultimate goal of this research is to develop self-adaptive software agents that intelligently manage gait behavior, analyzing changes in terrain conditions and adapting gaits to accommodate. While only a rough outline was presented of preliminary work that is still underway, we hope to minimize pre-compiled strategies in favor of self-developed tuning and measurement rules developed largely by learning from human demonstration.

ACKNOWLEDGMENTS

This material is based upon work supported by the Defense Advanced Research Projects Agency, Electronics Technology Office ("Distributed Robotics" Program), ARPA Order No. G155, Program Code No. 8H20, Issued by DARPA/CMD under contract #MDA972-98-C-0008 and by the Air Force Research Laboratory under Contract Number F30602-96-2-0240. Any opinions, findings, conclusions or recommendations expressed herein are those of the authors and do not reflect the views of the Air Force Research Lab, DARPA, Carnegie Mellon University or the University of Minnesota.

REFERENCES

[1] Hougen,D.F., S. Benjaafar, J. C. Bonney, J. R. Budenske, M. Dvorak, M. Gini, H. French, D. G. Krantz, P. Y. Li, F. Malver, B. Nelson, N. Papanikolopoulos, P. E. Rybski, S. A. Stoeter, R. Voyles and K. B. Yesin, 2000, "A Miniature Robotic System for Reconnaissance and Surveillance," in *Proc. of the IEEE Intl. Conf. on Robotics and Automation*, v. 1, pp.501-507.

[2] Morse, W.D., D.R. Hayward, D.P. Jones, A. Sanchez, D.L. Shirey, 1994, "Overview of the Accident Response Mobile Manipulation System (ARMMS)," in *Proceedings of the ASCE Specialty Conference on Robotics for Challenging Environments*, Albuquerque, NM., pp. 304-310.

[3] Carriker, W.F., P.K. Khosla, B.H. Krogh, 1991, "Path Planning for Mobile Manipulators for Multiple Task Execution," in *IEEE Transactions on Robotics & Automation*, v. 7, n. 3, June, pp. 403-408.

[4] Khatib, O., 1999, "Mobile Manipulation: The Robotic Assistant," in *Robotics & Autonomous Systems*, v 26, n. 2-3, Feb. pp. 175-183.

[5] Saito, F., T. Fukuda, F. Arai, "Swing and Locomotion Control for a Two-Link Brachiation Robot," in *IEEE Control Systems Magazine*, v. 14, n. 1, Feb. pp. 5-11.

[6] Xu, Y., H.B. Brown, Jr., M. Friedman, and T. Kanade, 1994, "Control Systems of Self-Mobile Space Manipulator," in *IEEE Trans. on Control Systems Technology*, v. 2, n. 3, pp. 207-219.

[7] Xu, Y., C. Lee, and H.B. Brown, Jr., 1996, "A Separable Combination of Wheeled Rover and Arm Mechanism: (DM)2," in *Proceedings of the IEEE International Conference on Robotics and Automation*, v. 3, pp. 2383-2388.

[8] Yim, M., 1994, "New Locomotion Gaits" in *Proc. of IEEE Intl. Conf. on Robotics and Automation*, v. 3, pp. 2508-2513.

[9] Pai, D.K., R.A. Barman, and S.K. Ralph, 1994, "Platonic Beasts: A New Family of Multilimbed Robots," *Proc. of the IEEE International Conf. on Robotics and Automation*, v 2, pp. 1019-1025.

[10]Mason, M.T., D.K. Pai, D. Rus, L.R. Taylor, and M.A. Erdmann, 1999, "A Mobile Manipulator," *Proc. of the IEEE International Conf. on Robotics and Automation*, v 3, pp. 2322-2327.

[11]Yesin,K.B., B. Nelson, N. Papanikolopoulos, R. Voyles and D. Krantz, "Active Video System for a Miniature Reconnaissance Robot" in *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, v 4, pp. 3920-3925.

[12]www.cs.cmu.edu/~pb3a

[13]Stewart, D.B. and P.K. Khosla, 1996, "The Chimera Methodology: Designing Dynamically Reconfigurable and Reusable Real-Time Software Using Port-Based Objects," *International Journal of Software Engineering and Knowledge Engineering*, 6(2):249-277.

[14]Kimura, H. and Y. Fukuoka, 2000, "Adaptive Dynamic Walking of the Quadruped on Irregular Terrain - Autonomous Adaptation Using Neural System Model," *Proceedings of the IEEE International Conference on Robotics and Automation*, v1, pp. 436-443.

[15]Voyles, R.M., "TerminatorBot: A Robot with Dual-Use Arms for Manipulation and Locomotion," *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, v. 1, pp. 61-66.

[16]Voyles,R.M. "Novel Gaits for a Novel Crawling/Grasping Mechanism," in Proceedings of the International Symposium on Adaptive Motion of Animals and Machines, v 1, Aug. 2000.

[17]Papanikolopoulos, N.P., Khosla, P.K., and Kanade, T., "Visual Tracking of a Moving Traget by a Camera Mounted on a Robot: A Combination of Control and Vision.", *IEEE Transactions on Robotics and Automation*, Volume 9, No 1, February 1993, pp 14-35.

[18]Nelson, B.J., P.K. Khosla, "Force and Vision Resolvability for Assimilating Disparate Sensory Feedback." *IEEE Transactions on Robotics and Automation*, vol.12, no. 5, pp. 714-731, 1996

[19]Corke, P.I. and R.P. Paul, "Video-rate Visual Servoing for Robots," *Lecture Notes in Control and Information Science*, eds. V. Hayward and O. Khatib, pp. 429-451, 1989.

[20]Espiau, B., F. Chaumette, and P. Rives, "A New Approach to Visual Servoing in Robotics," *IEEE Transactions on Robotics and Automation*, v. 8, n. 3, pp. 313-326, 1992.

[21]Hutchinson, S., G.D. Hager, and P.I. Corke, "A Tutorial on Visual Servoing Control," IEEE Transactions on Robotics and Automation, v. 12, n. 5, pp. 651-670, 1996.