Behavior-Based Mobile Manipulation for Drum Sampling

Douglas C. MacKenzie   Ronald C. Arkin
Mobile Robot Laboratory
College of Computing
Georgia Institute of Technology
Atlanta, Georgia 30332-0280 U.S.A.

Abstract

This paper describes an implementation of a behavior-based mobile manipulator capable of autonomously transferring a sample from one drum to a second in unstructured environments. A major contribution of the project was the coherent integration of the arm and base as a cohesive unit, and not just a mobile base with an arm attached. The support for smooth simultaneous operation of all joints on the vehicle facilitated biologically plausible motions, such as arm preshaping. The behavior-based controller used a pseudo-force model, where behaviors add forces and torques to joints and limbs resulting in coordinated motion. The vehicle Jacobian is used to convert the pseudo-forces into joint torques and a pseudo-damping model converts the joint torques into joint velocities. This process allows rapid control of the manipulator without the use of inverse kinematics. A drum sampling task is presented where the vehicle demonstrates how a sample of material could be moved from one drum to another, illustrating the efficacy of the solution.

1 Introduction

This paper describes construction of a behavior-based mobile manipulator using a motor schema-based approach [5]. A primary goal of the project was development of an integrated mobile manipulator, not just an arm mounted on a mobile base. This goal grew out of the desire to support biologically plausible motions [2], such as arm preshaping, and grasping while moving. Preshaping refers to the process of forming the arm into the configuration necessary to acquire a target as the robot begins to close on the target. This preshaping process is important to minimize the time required at the target and to support grasping while the robot is moving. To construct such a controller, a generalization of our motor schema-based approach using pseudo-forces and a pseudo-joint damping model to cause and control motion was developed [10]. This control paradigm permits rapid computation of the desired joint velocities without the use of inverse kinematics.

In this paper, a drum sampling task is presented where the vehicle demonstrates how a sample of material could be moved from one drum to a second. This task integrates our work in temporal sequencing[3] with our research in visual tracking[19] and visual servoing[17]. Images taken from a robot waist camera are processed using a constrained, line-based Hough transform[19, 14] to recognize and localize the drums in natural settings using a stored model. The perceptual process provides the direction and distance to the drum, which is used to servo the vehicle towards the drum. When the robot arrives at the drum, a wrist mounted camera is used to discern the bung hole using thresholding of the dark, open bung hole against the light colored drum cover. An ellipse fitting algorithm provides a certainty metric and discards false matches.

2 The mobile manipulator

The mobile manipulator is constructed from a Denning MRV-2 mobile robot and a CRS A251 industrial robot arm. The MRV-2 uses a three wheel synchronized steering system to allow motion without turning the vehicle body. The arm controller is mounted in a cart pulled behind the robot base. An off-board Sun workstation functions as the mobile manipulator controller, providing integrated control of the arm and base as a unified mobile manipulator. Treating the two disparate components as an integrated mechanism is central to this research. Proprioceptive sensors include position encoders on the wheels and joints, and a force/torque sensor on the wrist. A ring of 24 ultrasonic sensors surrounding the base provides drum position and obstacle detection information. Cameras are mounted on the robot's wrist and on a waist pan-tilt mount to accommodate active visual processes.

Motion for the mobile manipulator is generated using a behavior-based methodology. The reactive controller was documented in [8, 10] and is only overviewed here. The basic premise is to guide the motion of the arm and base by responding to artificial forces generated by concurrent active behaviors. Conventional two dimensional reactive methodologies (e.g., subsumption-based architectures [9], REX [15], RAPs [12], schema-based systems [5], and many others [1, 13, 18]) have been extended to three dimensions. Related work by Connell in mobile manipulation [11] differs in that in his approach the arm and base are treated separately from a behavioral perspective, i.e, the arm inhibits the base's motion during reaching and grasping. Our methodology cannot only replicate this strategy but more importantly permits the coherent and concurrent motion of the arm and base simultaneously for tasks such as preshaping.

To induce movement of the robot towards a goal attractor (e.g., a drum), the *move_to_goal* behavior adds a force to the end-effector, pulling it towards the goal. Other active behaviors add forces and torques of varying direction and strength to the vehicle joints. The vehicle Jacobian is then used to convert the pseudo-force into the corresponding joint torques (computed at static equilibrium). A pseudo-damping model is then used to convert the joint torques into the induced joint velocities. Qualitative changes in the mobile manipulator's behavior are generated by controlling the damping values. For example, making the damping values appropriate functions of the distance to the goal can produce rapid motion of the base towards the goal while keeping the arm relatively rigid when the robot is distant (i.e., the base joints are loose and the arm joints are stiff). As the vehicle approaches the goal, the base slows (stiffens) and preshaping occurs as the arm extends while reaching for the target object (as a consequence of the dampening being reduced on the arm itself).

3 Perceptual Support for Drum Sampling

Two independent perceptual processes are used to discern the drum and the open bung hole. The first utilizes the waist camera and the second a wrist-mounted camera looking down on the drum.

3.1 Drum localization

The waist camera, mounted on a Directed Perception pan-tilt mechanism, is used to locate the drum and servo the vehicle towards it. The *move_to_goal* motor behavior is coupled with the *detect_drum* perceptual module to support movement towards the drum. The *detect_drum* algorithm operates as follows. First, the shaft-encoders are read to compute a coarse expected location for the drum (i.e., there is some *a priori* knowledge of the drum's general whereabouts). The waist camera is then oriented towards this expected location using the pan/tilt and an image is captured.

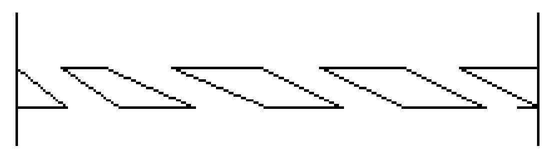


Figure 1: Model used to match the drums

A region-based line finder[16] is then invoked to extract lines from the image which are oriented collinear (within a tolerance) with the set of line orientations occuring within a stored model (shown in Figure 1). This model is based upon the patterned hazard marking tape applied to the upper region of the drum. A line-based Hough transform is then used to compare the model to the extracted image lines. The matching algorithm is shown in Figure 2.

```
scale_model(estimated_distance);
FOR seg = each_line_in_image
  FOR mline = each_line_in_model
    IF angle_between(seg,mline) < THRESHOLD
      weight = pixels_in(seg) / pixels_in(model)
      draw_model_in_hspace(weight);
    ENDIF
  END
END
location = pick_largest_hspace_bucket()
```

Figure 2: Line-based Hough transform

Once the location of the model in the image is extracted, an algorithm searches horizontally along the model's estimated location in the image to find the lines extracted along the vertical edges of the drum. The location of these edges is then used to improve the accuracy of the drum location as well as predict the distance to the drum via projection.

3.2 Bung hole localization

The *detect_bung_hole* perceptual module consists of several steps. First, an image is taken using the wrist camera. Using a realistic expectation of the open bung hole appearing dark against the lightly colored drum cover, images are processed using the multi-pass algorithm shown in Figure 3. During each pass, the darkest pixel (remaining unprocessed) within the processing window is suggested as a possible location for the bung hole. A blob-growing algorithm then processes the pixels surrounding the suggested location. Using the grown blob, a relaxation-based ellipse fitting algorithm determines if the blob is a likely match for the bung hole, and if so, returns the center of the hole.

```
FOR loops = 1 to max_tries
  loc = darkest_pixel(image)
  blob = grow_blob(image,loc)
  IF close_to_expected_size(blob.area)
     ellipse = fit_ellipse(blob)
     IF ellipse.good
        return ellipse
     ENDIF
  ENDIF
END
return failure
```

Figure 3: The `detect_bung_hole` algorithm

The `fit_ellipse` algorithm shown in Figure 4 uses a relaxation process to shape and size an ellipse to the blob. The axes of the fitted ellipse are oriented along the horizontal and vertical axes of the image. The ellipse is described by the equation

$$\frac{(x-x_0)^2}{a^2} + \frac{(y-y_0)^2}{b^2} = 1$$

where x_0, y_0 is the center (x,y in Figure 4) and a, b are the major and minor radii (horizontal and vertical in this usage). Blob pixels outside of the ellipse exert a force on the edge of the ellipse attempting to slide it towards them (away from the center). Non-blob pixels inside the ellipse exert a similar force on the ellipse in the opposite direction (towards the center). The relaxation process iterates, allowing balanced forces on opposite sides of the ellipse to deform it while non-balanced forces cause it to slide. Once the process

```
x,y,a,b = initial_estimates
FOR loops = 1 to max_relaxation_loops
  left,right,top,bot = 0
  // relax vertical (b,y parms)
  FOR col = left_in_ellipse TO right_in_ellipse
    add to top distance the highest blob pixel
     in the col is above(+) or below(-) ellipse
    add to bot distance the lowest blob pixel
     in the col is below(+) or above(-) ellipse
  END
  IF(top and bot agree on expand or contract)
     increment or decrement b based on direction
  ENDIF
  IF(top and bot differ in magnitude)
     increment or decrement y to balance them
  ENDIF
  // relax horizontal (a,x parms)
  FOR row = bot_in_ellipse TO top_in_ellipse
    add to left distance the leftmost blob pixel
     in the row is left(+) or right(-) of ellipse
    add to right distance the rightmost blob pixel
     in the row is right(+) or left(-) of ellipse
  END
  IF(left and right agree on expand or contract)
     increment or decrement a based on direction
  ENDIF
  IF(left and right differ in magnitude)
     increment or decrement x to balance them
  ENDIF
END
RETURN a,b,x,y
```

Figure 4: The `fit_ellipse` algorithm

stabilizes, the center of the ellipse is returned as the probable location of the center of the drum's bung hole.

4 Drum sampling behavioral configuration

The utility of the mobile manipulator was shown by implementing a drum sampling task where the vehicle demonstrates how a sample from one drum could be transferred to a second container. Two environmental modifications were made to simplify portions of the experiment. First, the top 1/3 of the two 55 gallon drums was cut off, allowing the robot to physically reach the top. Second, as mentioned earlier, striped standard hazard marking tape was wrapped around the two drums to facilitate recognition against the visually cluttered backgrounds occuring in the lab.

The experiment begins with the vehicle initialized with an approximate knowledge of the locations of the drums (drum 1: forward 2 meters, drum 2: left 2 meters) to constrain the visual search.

Using the pan-tilt mount, the vehicle directs the waist camera towards the drum's expected location. It then attempts to recognize the drum using a line-based constrained Hough transform, based on the model of the hash marks on the drum. If no match is found, then a panning visual search for the drum is initiated. When a match is discovered by the Hough transform, a second algorithm searches for the edges of the drum within the set of extracted lines. Finding the true edges of the drum allows determination of the actual distance to the drum by transforming the width to distance. It also permits correcting for the offset error induced by apparent motion of the hash marks as the drum rotates relative to the robot.

The vehicle then moves towards the recognized drum, preshaping the arm to position it above the open bung hole. When the bung hole becomes visible using the wrist camera, a downward docking motion moves the wrist-mounted probe (a soda straw) about 6 inches into the drum to emulate transferring samples. The vehicle then retracts the arm and backs away from the drum. The process is then repeated for the second drum to show how additional samples could be collected. Finally, the vehicle halts after finishing with the second drum.

The behavioral configuration which implements this drum sampling task uses two levels of temporal sequenced coordination[3] to control execution of the various stages of the mission. Figure 5 shows the top-level coordination operator which causes the manipulator to demonstrate how a sample of material could be moved from one drum to a second.

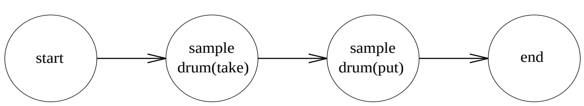


Figure 5: FSA to move samples

Figure 6 shows the details of the **sample_drum** assemblage. This finite state acceptor (FSA) moves the vehicle towards a selected drum, positions the arm over the open bung hole, lowers the probe into the drum, transfers a sample, removes the probe, and backs away. For these experiments, the **transfer_sample** state was just a short pause since no actual material transfer took place. This assemblage is active in both the **sample_drum(take)** and **sample_drum(put)** states in the *move samples* assemblage. The behaviors active in each state of the **sample_drum** FSA are listed in Table 1 and are documented in [5, 4, 7, 10]. They are only overviewed here.

- **set-goal**: causes the vehicle to conduct a visual search to initially locate an object of interest (*i.e.*, the drum or bung hole).
- **move-to-goal**: generates a constant magnitude three dimensional attractive force on the end-effector, pulling it towards the goal position.
- **avoid-obstacle**: generates forces and torques on the vehicle, repulsing it from obstacles. A cylindrical repulsive field is constructed around each link (the robot base is considered one link) and a spherical repulsive field is constructed around each joint.
- **move-ahead**: Draws the end-effector in a particular 3D direction at a fixed magnitude.
- **docking**: generates forces and torques on the mobile manipulator forcing the end-effector to approach in a particular orientation.

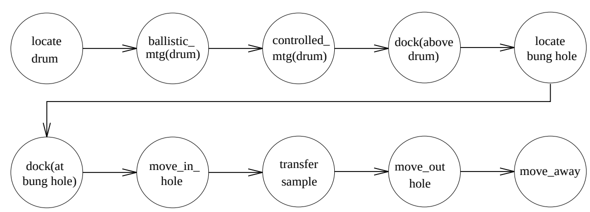


Figure 6: FSA to probe drum

5 Experimental run

Multiple runs of the mobile manipulator executing the drum sampling task configuration were completed in the Georgia Tech Mobile Robot lab. The mobile manipulator incorporates a route planner[6] for utilizing *a priori* map information in partially modeled environments. However, for these experiments, the planner was not used. Instead, the robot was provided with rough estimates of the locations of the drums relative to its starting location. This information was used to constrain the initial visual search for the drums, to reduce startup time. The accuracy of the estimates was approximately ± 1 meter.

Figure 7(a) shows the mobile manipulator in the starting configuration. Figure 7(b) shows the waist camera view with the extracted lines and the drum

Table 1: Behaviors active in `sample_drum` FSA states

State	Exit Trigger	Active Behaviors
`locate_drum`	found drum	set_goal(waist_camera)
`ballistic_mtg(drum)`	near drum	avoid_obstacle(ultrasonics) move_to_goal(detect_drum(shaft_encoders,waist_camera))
`controlled_mtg(drum)`	at drum	avoid_obstacle(ultrasonics) move_to_goal(detect_drum(shaft_encoders,waist_camera))
`dock(above_drum)`	over drum	dock(detect_drum(shaft_encoders))
`locate_bung_hole`	saw bung hole	set_goal(wrist_camera)
`dock(at_hole)`	lost bung hole	dock(detect_bung_hole(shaft_encoders,wrist_camera))
`move_in_hole`	within drum	move_ahead
`transfer_sample`	timeout	no active behaviors
`move_out_hole`	above drum	move_ahead(shaft_encoders)
`move_away`	away from drum	move_to_goal(shaft_encoders) avoid_obstacle(ultrasonics)

model (sized to match the calculated distance) overlaid on the image. The drum is tracked as the vehicle moves to correct for locational uncertainty and to handle dynamic environments where the drum is permitted to be moved. Once the vehicle gets within four feet of the drum, the robot stops visually tracking the drum since the edges of the drum may extend beyond the field of view of the camera. Figure 7(c) shows the image taken using the waist camera at this transition point.

Next, the arm is activated causing it to leave the safe travel position it has been in and to extend straight up. Starting from this raised configuration allows the docking behavior to position the wrist camera to achieve the best possible view of the bung hole. Using shaft-encoders and dead-reckoning the mobile manipulator moves the end-effector above the drum, positioning the wrist camera vertically, to allow viewing of the bung hole. Figure 7(d) is a photo of the vehicle during the docking phase.

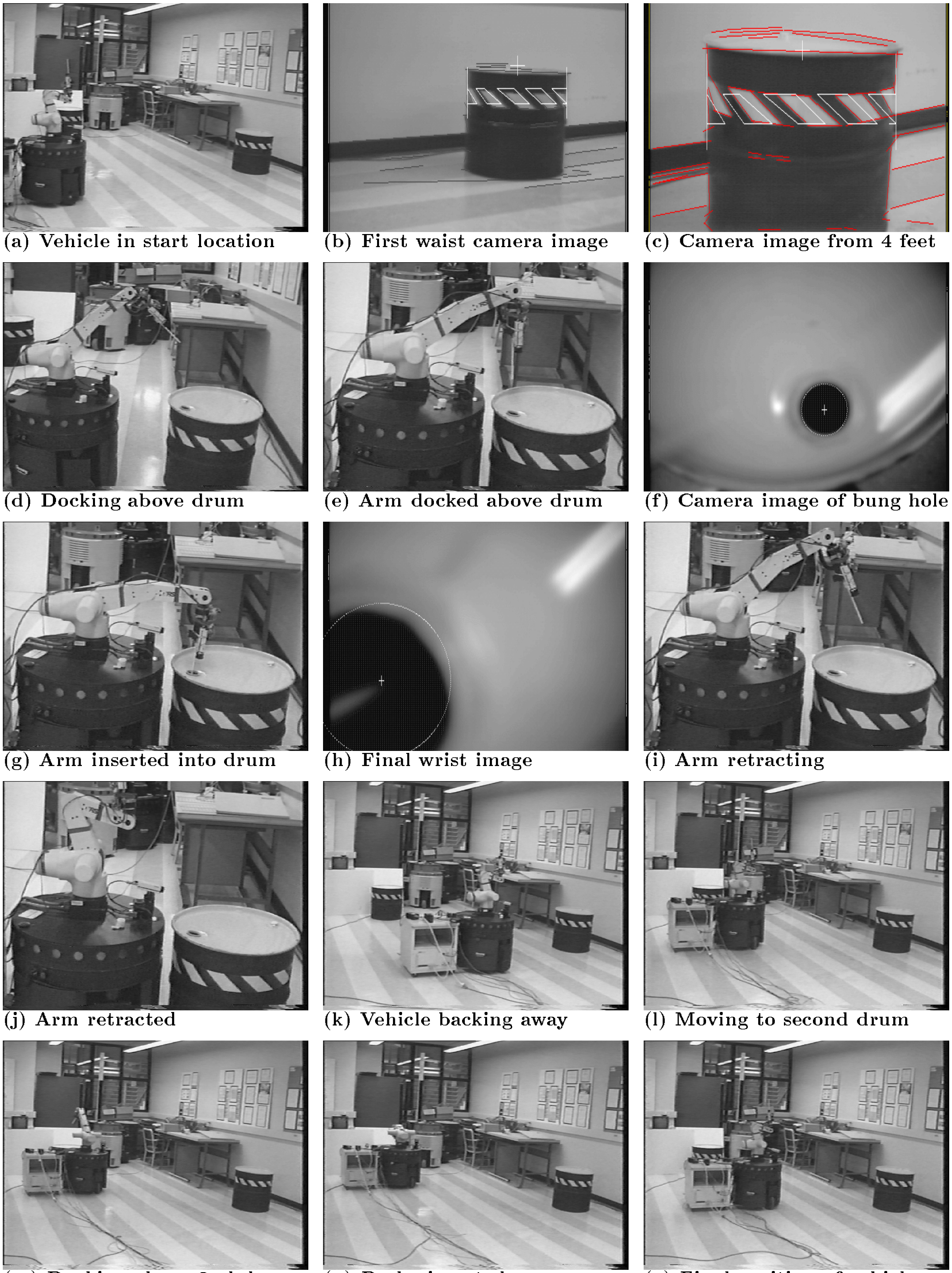
The vehicle now uses the wrist camera to track the open bung hole. A low-cost vision algorithm is used which thresholds the image looking for the dark bung hole. The ellipse fitting algorithm then verifies that the tracker has in fact located the hole and estimates the distance to the bung hole based on its perceived diameter in the image. This corrects for uncertainties in the robot's position and handles drums of varying height. Figure 7(e) is a photo of the arm docked above the drum and (f) shows the wrist camera image taken in this position with the extracted bung hole highlighted and the projected center of the hole marked with a cross.

The vehicle continues tracking the hole as it lowers the sampling probe into the drum, visually servoing to correct for motion errors. Figure 7(g) shows the arm in the fully inserted position, and (h) shows the corresponding wrist camera image. Notice that although the probe visibly protrudes into the image (from the lower left corner towards the center of the hole) and the entire hole is not visible, the ellipse is still positioned reasonably accurate. The hole is servoed to the lower left corner of the image because the probe is centered in the wrist and the camera is offset by about 3 inches in both X and Y from the centerline of the arm.

The vehicle then retracts the probe and backs away from the drum to get a good view of the second drum. Figure 7(i) shows the arm retracting, (j) shows the arm fully retracted, and (k) shows the robot backing away from the first drum. It then repeats this entire process to deliver the sample to the second container as shown in Figure 7, images (l)-(n). Finally, it retracts the manipulator into the safe traveling configuration and backs away from the delivery drum before halting. Figure 7(o) shows the vehicle in the final, halted state.

6 Summary

This paper has described construction of a behavior-based mobile manipulator using a motor schema-based approach. The vehicle operates as an integrated system, utilizing preshaping while moving and supporting grasping while moving. A new schema-based controller was developed for this project which uses application of pseudo-forces to induce mo-



(a) Vehicle in start location (b) First waist camera image (c) Camera image from 4 feet

(d) Docking above drum (e) Arm docked above drum (f) Camera image of bung hole

(g) Arm inserted into drum (h) Final wrist image (i) Arm retracting

(j) Arm retracted (k) Vehicle backing away (l) Moving to second drum

(m) Docking above 2nd drum (n) Probe inserted (o) Final position of vehicle

Figure 7: Photos of sample drum test run

tion. An attractive behavior such as *move_to_goal* induces a force on the end-effector pulling it towards the goal. Repulsive behaviors such as *avoid_obstacles* induce repulsive forces on joints when obstacles approach nearer than a safe distance. Other behaviors add forces and torques to joints as required to induce their desired motions. The integrated vehicle Jacobian matrix (at static equilibrium) is used to convert these forces into the set of corresponding joint torques that would be induced. A pseudo-damping model is then used to convert the joint torques into joint velocities. By controlling the damping values at each joint using functions of distance to the goal, varying performance characteristics are generated. A drum sampling task was demonstrated to show the feasibility of this type of control in real-world problems.

Acknowledgments

This research was supported by the National Science Foundation under grants IRI-9100149 and IRI-9113747 and the Material Handling Research Center of the Georgia Institute of Technology. The authors would like to acknowledge the efforts of Keith Ward and Jonathan Cameron in developing hardware and software in support of this project.

References

[1] T. Anderson and M. Donath, "Animal Behavior as a paradigm for developing robot autonomy", in *Designing Autonomous Agents*, (P. Maes, ed.), MIT Press, Cambridge, MA, pp. 145, 1990.

[2] M. Arbib, T. Iberall, and D. Lyons, "Coordinated Control Programs for Movements of the Hand", *Exp. Brain Res. Suppl.*, Vol. 10, pp. 111-129, 1985.

[3] Arkin, R.C. and MacKenzie, D., "Temporal Coordination of Perceptual Algorithms for Mobile Robot Navigation", *IEEE Transactions on Robotics and Automation*, Vol 10, No. 3, June 1994, Pg 276-286.

[4] R.C. Arkin, "Behavior-based Robot Navigation in Extended Domains", *J. of Adaptive Behavior*, Vol. 1, No. 2., 1992.

[5] R.C. Arkin, "Motor Schema Based Mobile Robot Navigation", *International J. of Robotics Research*, Vol. 8, No. 4, pp. 99, 1989.

[6] R.C. Arkin and D. MacKenzie, "Planning to Behave: A Hybrid Deliberative/Reactive Control Architecture for Mobile Manipulation", *1994 International Symposium on Robotics and Manufacturing*, Maui, HI, August 1994, pp. 5-12.

[7] Arkin, R.C. and Murphy, R.R., "Autonomous Navigation in a Manufacturing Environment", *IEEE Transactions on Robotics and Automation*, Vol. 6, No. 4, August 1990, pp. 445-454.

[8] Arkin, R.C., editor, "Integrated Control for Mobile Manipulation for Intelligent Materials Handling", *Intelligent Material Handling Using Mobile Robots—Collected Papers from the Mobile Robot Laboratory,* Georgia Tech Material Handling Research Center Report OP-92-19, 1992.

[9] R. Brooks, "A Robust Layered Control System for a Mobile Robot", *IEEE Jour. of Robotics and Auto.*, Vol. 2, No. 1, pp. 14, 1986.

[10] Cameron, J., MacKenzie, D., Ward, K., Arkin, R.C., Book, W., "Reactive Control for Mobile Manipulation", *Proc. 1993 IEEE International Conference on Robotics and Automation*, Atlanta, GA, May 1993, Vol. 3, pp. 228-235.

[11] J.H. Connell, "A Behavior-Based Arm Controller," *IEEE Trans. on Robotics and Auto.,* Vol. 5, No. 6, Dec. 1989, pp. 784-791.

[12] R.J. Firby, "Adaptive Execution in Complex Dynamic Worlds", Ph.D. Dissertation, Yale University, YALEU/CSD/RR #672, January, 1989.

[13] E. Gat, "Robust, Task-Directed, Reactive Control of Autonomous Mobile Robots", *Ph.D. Dissertation*, Virginia Polytechnic Inst. and State Univ., 1991.

[14] P.V.C. Hough, "Method and means for recognizing complex patterns", U.S. Patent 3,069,654; 1962.

[15] L. Kaelbling and S. Rosenschein, "Action and planning in embedded agents", in *Designing Autonomous Agents*, pp. 35, (P. Maes, ed.), MIT Press, Cambridge, MA, 1990.

[16] P. Kahn and L. Kitchen, "A fast line finder for vision-guided robot navigation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 11, Nov. 1990, pp. 1098-1102.

[17] D. MacKenzie and R.C. Arkin, "Perceptual Support for Ballistic Motion in Docking for a Mobile Robot", *Proc. SPIE Conference on Mobile Robots VI*, Nov. 1991, Boston, MA, pp. 22-33.

[18] D. Payton, "An Architecture for Reflexive Autonomous Vehicle Control", *IEEE Conf. on Robotics and Auto.*, pp. 1838, 1986.

[19] Vaughn, D.L. and Arkin, R.C., "Workstation Recognition using a Constrained Edge-Based Hough Transform for Mobile Robot Navigation", *SPIE Conference on Sensor Fusion III: Three Dimensional Perception and Recognition*, Boston, MA, pp. 503-514, 1990.