Temporal Coordination of Perceptual Algorithms for Mobile Robot Navigation

Ronald C. Arkin and Douglas MacKenzie

Mobile Robot Laboratory
College of Computing
Georgia Institute of Technology
Atlanta, GA 30332-0280

Abstract

A methodology for integrating multiple perceptual algorithms within a reactive robotic control system is presented. A model using finite state acceptors is developed as a means for expressing perceptual processing over space and time in the context of a particular motor behavior. This model can be utilized for a wide range of perceptual sequencing problems. The feasibility of this method is demonstrated in two separate implementations. The first is in the context of mobile robot docking where our mobile robot uses four different vision and ultrasonic algorithms to position itself relative to a docking workstation over a long-range course. The second uses vision, IR beacon, and ultrasonic algorithms to park the robot next to a desired plastic pole randomly placed within an arena.

1. Introduction

Designing robust perceptual algorithms for navigation in open world domains is a very difficult task. One strategy that can be exploited to cover a wide range of robotic operations is to design a sequence of perceptual algorithms, each associated with specific task-oriented behaviors and certain environmental circumstances. This strategy is in contrast to a reliance on one or a few algorithms that are in constant use over the entire range of robotic operation.

The channeling of perceptual processing to motor control has been one of the founding principles for our research in intelligent navigation. The use of reactive control through behavioral decomposition[3] provides a context for the choice of robotic perceptual strategies while the use of action-oriented perception[6] serves as the basis for tying together tightly focussed perceptual algorithms with specific behaviors.

We have previously demonstrated the efficacy of these principles when tested in isolation[6,11,24,27]. It is our goal, however, to accomplish missions which rely even more heavily on a wide range of perceptual capabilities. Thus it becomes important to be able to reconfigure sensor strategies (perceptual schemas) as the robot moves through the world. The question of "what to perceive and when to perceive it?" must be answered in order to provide timely robotic response in a dynamic and only partially modeled world. This article specifically addresses this problem: coordinating motor control with multiple supportive perceptual algorithms.

A brief review of recent work in selective perception and reactive robotic control is presented in the next section. The underlying philosophy and methodology of our particular approach follows. The first instantiation of this methodology is then presented in the context of a mobile robot docking with a workstation. Four different perceptual algorithms are sequenced to accomplish this task from a range of about 40 feet to a final position within several inches of the dock. Experimental results are presented using our mobile robot George. A second implementation that was developed for the Georgia Tech entry in the AAAI 1992 mobile robot competition[16] is also presented. Three perceptual algorithms are used to guide the robot to a specific plastic pole. A summary and conclusions section completes the article.

2. Related Work

Developments in the areas of reactive robotic control and selective perception serve as a background for this research.

2.1 Reactive Robotic Control

Reactive control is an approach to robotics that eliminates the use of intervening representation and reasoning during the execution of a robot's mission. A tight coupling between perception and motor action is present. This strategy provides for real-time response and is particularly well-suited for dynamic and unmodeled (or partially modeled) environments. Additionally robotic actions are typically decomposed into a collection of primitive motor behaviors, each of which is capable of functioning more or less independently.

There are now many representative examples of reactive control systems. These include the subsumption architecture[13,15], the use of virtual behaviors[28], and work by a variety of other research groups (e.g., [22,30,19]).

Our approach to reactive control is based on the use of motor schemas[3]. Multiple motor schemas (behaviors), instantiated on the basis of higher-level knowledge[4], act in a concurrent manner to yield a globally emergent behavior which strives to satisfy the robot's goals. Many different schemas have been defined for a variety of situations. The most important ones for the context of this article include the following:

- **move-to-goal**: directs the robot towards a perceived object.
- **avoid-static-obstacle**: moves robot away from detected impediment to motion.
- **noise**: random vector to produce wandering and cope with local minima and maxima.
- **avoid-past**: moves robot away from where it has recently visited.
- **probe**: directs robot towards open areas.

Each behavior independently creates a velocity vector based on an analog of the potential fields method which are summed and normalized to yield the desired velocity of the robot based upon its immediate perceptions of the world.

2.2 Selective Perception

Specialized perceptual strategies are necessary to support behavior-based robotic systems. Instead of conducting large-scale scene analysis, task-specific sensor algorithms can be constructed and run in parallel to support each individual motor behaviors' needs. This philosophy which goes under the name of action-oriented perception, task-specific perception, and selective perception all strive to channel perceptual attention and energy effectively. A recent AAAI Workshop on selective perception includes several representative approaches to this problem[8].

3. Approach

In order to best utilize limited computational resources it is important to channel perceptual processing effectively. Using a behavior-based approach for the design of robotic systems enables the focusing of perceptual processing in ways that are fundamentally different from those whose efforts are concentrated into building global representations of the world (e.g., [12,23,25]).

There are several aspects to this approach[5]:

1. Sensor fission: splitting off perceptual processing in the context of particular motor tasks and running these perceptual agents (schemas) in parallel.

2. Action-oriented sensor fusion: utilizing multiple perceptual algorithms concurrently to build up *task-dependent* representations of the world. [26]

3. Perceptual sequencing (sensor fashion): utilizing different perceptual strategies at different points in time and space to produce a uniform motor action.

The remainder of this article outlines the natural seams by which perceptual algorithms can be readily coordinated and concentrates on item (3) above.

3.1 Perceptual Sequencing

As discussed previously, by avoiding the construction of task-independent representations of the world, it is ensured that only relevant sensory processing is being performed. It must also be recognized that a single perceptual algorithm (or single aggregate of concurrent perceptual algorithms) will often not be able to satisfy the extended motor needs of a single behavior (schema). For example, it is wasteful to have long-range goal recognition algorithms in operation when the robot is close to its goal. The same holds for the value of close range positioning algorithms when the robot is far from its target. Multiple perceptual algorithms are called for, with a mechanism for sequencing them during the performance of a motor behavior.

Perceptual sequencing is a methodology by which multiple perceptual algorithms can be scheduled within the context of a single motor behavior. The goal is to have the right perceptual algorithm active at the right time. Little previous work has been performed in the area of perceptual sequencing in the context of a reactive robotic system. Most previous research has either been concerned with fusing together multiple observations over time (traditional sensor fusion) or using a single perceptual algorithm to control a specific robotic behavior (conventional reactive systems).

A framework must be developed to express the relationships between the perceptual processes. It needs to include the notion of perceptual state and a means for expressing the transitions between these states.

Trigger events are specified which represent these transitions. The nature of these triggers can be quite varied:

- They can be derived from spatial uncertainty information and the robot's expected position relative to a perceptible object[7].

- They can be based directly on elapsed time since the beginning of an event.
- They can be in response to the successful completion of a perceptual algorithm.
- They can result from failure conditions detected during perceptual processing.
- They can occur due to termination of a motor activity.

Formalizing the approach, a finite state acceptor (FSA)[1] can be devised to represent these relationships. Finite state models have been previously used to represent relationships in robotic control. Brooks uses finite state machines to provide expression for motor control in the subsumption architecture[13]. Tachi and Komoriya[29] have used an automaton map representation in the MELDOG system. FSAs have also been used for path control of vision-based mobile robot navigation[31,18]. Our use of FSAs differs decidedly from all of these methods, both in application (perceptual coordination) and expression.

A finite state acceptor M can be specified by a quadruple (Q, δ, q_0, F) with:

- Q representing the allowable states (in this case, the allowable perceptual schema),
- δ being a transition function mapping the perceptual input and the current state to another, or perhaps the same, state (capturing the trigger mechanisms described above),
- q_0 is the starting sensory configuration, and,
- F is a set of accepting states, a subset of Q, indicating completion of the sensorimotor task.

δ can be represented in a tabular form where the arcs in the FSA (e.g., Fig. 1) are invoked by sensory information that causes the trigger condition to exist.

A simple example is given for the utilization of two distinct sensory algorithms for a single motor task. The first q_0 is used for distant perception, the second q_1 for close in navigation.

The FSA for this example (Fig. 1) would be represented by the quadruple:

$$(\{q_0, q_1, q_2, q_3\}, \delta, q_0, \{q_2, q_3\})$$

where q_2 represents normal successful termination, q_3 represents abnormal termination.

Here δ, the transition function, is specified by:

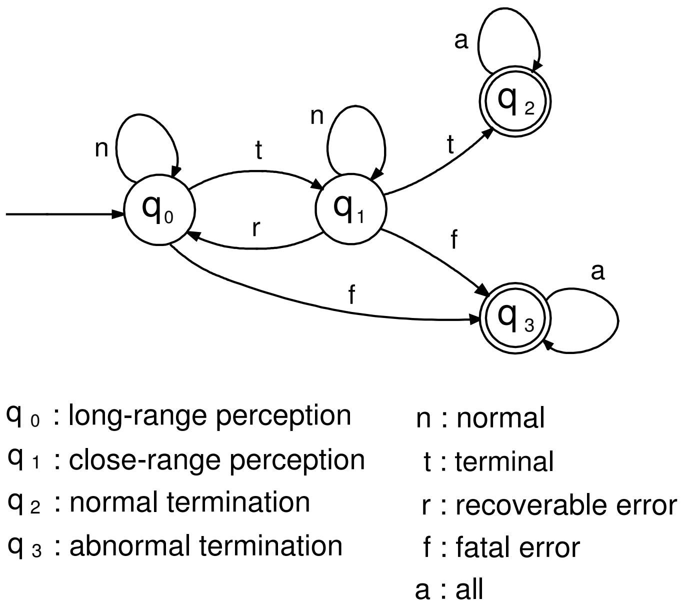


Figure 1: Simple example of a perceptual FSA

δ	normal	terminal	recoverable error	fatal error
q_0	q_0	q_1	q_1	q_3
q_1	q_1	q_2	q_0	q_3
q_2	q_2	q_2	q_2	q_2
q_3	q_3	q_3	q_3	q_3

The column on the left depicts the current state with transitions occurring to new states as determined by any of four different types of input (normal, terminal, recoverable error, and fatal error).

For this particular example, normal sensory input maintains the initial perceptual schema in its existing configuration driving the robot in accordance with the results of the algorithm. When new sensory input causes the first perceptual algorithm (q_0) to terminate normally (e.g., recognition, or some other trigger event), the second perceptual algorithm (q_1) is invoked. The second algorithm proceeds as long as valid sensory input is received. When successful goal completion occurs, it transits to the normal termination state (q_2). If an error occurs that is deemed recoverable, it reinvokes the first perceptual algorithm (q_0) to start over if possible, otherwise it denotes abnormal termination (q_3).

More comprehensive examples are presented in the experimental section of this paper.

The issue of complexity for FSAs which may have multiple transitions to new states from a single state is an important one. Opportunism can be represented in this method by having the robot have the potential to recognize one of many possible candidates for action. An example might be as a robot is moving through a building, it could be on the lookout for a recharging station, an object to retrieve, or something to clean. As schema-based control is inherently parallel [3], this presents no problem whatsoever. Multiple perceptual processes may be running in the context of a given state, each perhaps awaiting recognition of a particular event to invoke a state transition and thus carry out that particular behavioral stream (e.g. cleaning, recharging, or retrieving). If more than one trigger is satisfied at the same exact time (which is generally unlikely) either a priority based scheme could be used with one perceptual cue taking precedence over another, or a non-deterministic FSA could be used to resolve the conflict between them. Higher level processes could also dynamically affect the FSA as well, so that the perceptual plan could be modified based on other conditions (e.g., such as available energy resources enabling or disabling the perceptual strategy for finding the recharging station). This is an area left for future research.

3.2 Basis for Perceptual Sequencing

A natural decomposition arises from the existence of two distinct types of movement often found within the context of a single behavioral task: ballistic and controlled motion. There is strong physiological evidence for the existence of these two discrete types of motor control in biological systems[14,21]. Ballistic motion, a coarse high-speed operation, is typically supported by feedforward perception. Controlled motion, generally slower and more precise, usually involves intensive sensory feedback. The issue of how perceptual strategies can be effectively integrated to support ballistic and controlled motion in navigation is of primary concern in the following discussion, but these general principles can be extended to robotic manipulation tasks as well.

Within this context, at least three types of distinct perceptual activity prove useful:

- Ballistic feedforward

  Long range sensory cues that are utilized to initiate motion in a particular direction. Sensory feedback is not required to be rapid (or present at all for that matter), but a broad long-range and somewhat speculative motion (in regards to the goal being achieved) is undertaken. This type of perception has also been referred to as bootstrapping[2].

- Exteroceptive cueing

At some point during the robot's travel, a trigger must invoke the transition from ballistic to controlled motion. This trigger can be arbitrarily specified in terms of elapsed time or expected spatial proximity, or it can be be based on explicit perceptual recognition. In the case of a speculative perceptual cue being used to initiate ballistic motion, a subsequent recognition process can be used to confirm or annul the target hypothesis that initiated the ballistic motion. The results of the recognition operation can either make the switch to controlled motion in the case of a positive identification or else terminating ballistic motion in the case of inaccurate feedforward data.

- Adaptive feedback

  Once the target object has been identified, especially if located by a more costly model-based recognition technique, it is advantageous to abandon the preconceived notion of what the object looked like (i.e., the model) and track it based on its extracted appearance from instant to instant. By utilizing only the relevant salient features of the object, rapid sensory feedback can be provided for the controlled motion phase providing for precise final positioning of the robot relative to the target object.

The key to the overall process lies in designing low-cost (and usually low-confidence) perceptual algorithms that initiate ballistic motion; providing high-confidence model-based recognition algorithms to positively identify the target object/position; then transitioning to controlled motion using tracked feature data directly extracted from the sensory input stream. This process is demonstrated in two specific examples in the next two sections.

4. Workstation Docking Example

We have chosen to demonstrate these perceptual principles in a scenario within which we have often worked: docking in a manufacturing environment. The details for the docking motor schema are provided in[10]. Summarizing, there are two distinct regions: the ballistic component where the robot moves directly toward the workstation dock using relatively inaccurate and speculative feedforward perception, and the controlled component where the robot moves more cautiously using adaptive tracking techniques to achieve final positioning relative to the workstation. The transition zone is where the exteroceptive cue provides the change from ballistic to controlled motion. Within the controlled region there are two types of motion: *coercive*, where the robot moves around the dock and into the *approach* zone where the robot is channeled into the final trajectory for docking.

From a perceptual standpoint, the requirement for the long-range ballistic cue is straightforward: the generation of a hypothesis for the location of the docking workstation. Two techniques are available: one based on the assumption that workstations have a high degree of motion associated with them and the other utilizing a light source as the initial direction indicator[24]. A constrained Hough transform algorithm is used for explicit dock recognition[32] and adaptive region-based visual tracking techniques as well as ultrasonic distance measurements are used for final positioning[27].

4.1 Perceptual Algorithms

Space constraints preclude a detailed description of each of the perceptual algorithms described below. The details are available in other supporting papers which are cited where appropriate. The interested reader is referred to those articles if interested in implementing the vision algorithms. It should be remembered it is the intent of this paper to demonstrate the coordination of these perceptual strategies. Certainly other vision algorithms can and hopefully will be created to work in conjunction with, or in lieu of, the ones described below. The key point that must be remembered is not which algorithms are used, but rather how they are used in relation one to another.

4.1.1 Long-range Ballistic Cues

The long range perceptual algorithms are responsible for attracting the robot to the general area of the dock. They are not expected to be exact or robust and will fail under a variety of conditions. It is the job of a high-level planner to determine their applicability and selectively enable the algorithms that should prove useful in solving the current goal. Two such perceptual algorithms have been selected to be included in our current experimental implementation based on their simplicity and low computational requirements.

Phototropic perception

The phototropic (light seeking) perceptual algorithm[24] attracts the robot to areas of high light intensity. This bright light could be a naturally occurring workstation light or an artificial beacon placed at the intended destination. To initiate tracking, the camera searches for a bright light and then provides support for the ballistic phase of the docking schema. As the robot moves through the world, the phototropic algorithm generates an attractor field surrounding the light until it is deactivated. The algorithm functions simply by finding the location in the image with the highest intensity and frequently resampling the world as it moves toward its goal.

Activity Detection

The activity detection (motion seeking) perceptual algorithm determines areas of high motion activity. This motion would be expected to coincide with pick-and-place robots operating near the intended destination or other related workstation activity. Tracking is initiated by directing the camera towards the target. Spatial constraints[7] enable the bootstrapping of the algorithm to be straightforward.

This algorithm builds up motion intensity planes by differencing and normalizing a collection of successive video images. Two variations are feasible: the first is based on the absolute intensity difference between the images, while the second is unconcerned with contrast levels and assigns a single vote to each pixel whose difference exceeds a specific threshold. High-level planning would determine which variation is most appropriate for a given workstation. Details of the algorithm appear in[11].

4.1.2 Recognition using Spatially Constrained Hough transform

A spatially constrained Hough transform[32] is used to perform model-based recognition of the dock. This recognition phase serves as the hand-off point between ballistic motion and controlled motion. The algorithm is assisted by knowledge available from a spatial uncertainty map which expresses both positional and orientational uncertainty[7], constraining the possible locations of the dock. Feedback from distance traveled, elapsed time, or estimated vehicle position all can serve as trigger mechanisms. When triggered, the algorithm determines the most probable location of the dock within the current image. It generates a predicted location as well as a measure of certainty that can be used to evaluate the match. This algorithm serves as the exteroceptive cueing mechanism in our experimental example described below.

4.1.3 Adaptive Tracking via Fast Region Segmentation

The inverse perspective transform is applied to a target region of known size and position relative to the dock. The target region is extracted using an adaptive fast region segmentation algorithm[27] applied to the image. This region segmentation method is resilient to changing intensities across the region. The result is a measure of the distance and approach angle of the vehicle relative to the target. This algorithm provides the necessary visual feedback for the controlled portion of the docking maneuver.

4.1.4 Final Positioning with Ultrasound

When the robot gets close enough so that the target used for adaptive tracking goes out of the field of view, a final positioning algorithm using ultrasonic sensors is employed. This enables the robot to move to the final docking location where cargo can be transferred. This algorithm simply holds the vehicle heading until the remaining distance to the dock has decreased to the final goal value.

4.2 Experiments

An experimental scenario has been constructed to demonstrate the principles developed earlier in this paper. The goal is to demonstrate this methodology in which coherent control between perceptual algorithms can be obtained, and not to lay claim that these individual perceptual algorithms are necessarily the best for this particular task. We intend to develop more algorithms to supplement the existing ones described below. This study in perceptual sequencing is adequately served, however, by this battery of perceptual tools.

4.2.1 Discussion

The FSA shown in Figure 2 represents the control flow in our experimental system. The FSA would be represented by the quadruple

$$(\{q_0, q_1, q_2, q_3, q_4, q_5\}, \delta, q_0, \{q_4, q_5\})$$

where q_4 represents normal successful termination and q_5 represents abnormal termination.

Here δ is specified by:

δ	normal	terminal	recoverable error	fatal error
q_0	q_0	q_1	q_0	q_5
q_1	q_2	q_2	q_0	q_5
q_2	q_2	q_3	q_1	q_5
q_3	q_3	q_4	q_3	q_5
q_4	q_4	q_4	q_4	q_4
q_5	q_5	q_5	q_5	q_5

State q_0 indicates a long-range ballistic perceptual algorithm used to generate ballistic motion. This state represents either the phototropic or activity detection algorithms. The choice would be made by a higher level planner[4] before invoking the docking schema. In the experiment described below, the phototropic method is used

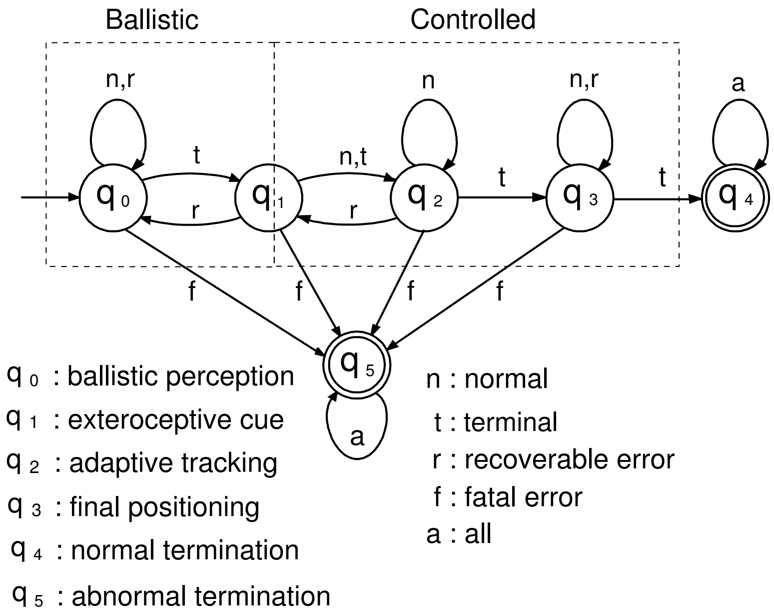


Figure 2: Experimental FSA

for state q_0. The *move-to-goal*, *avoid-static-obstacles*, and *noise* motor schemas were active in this state.

When the vehicle moves sufficiently close so that the structure of the dock itself should be discernible, a transition occurs, moving perceptual control to state q_1. This state represents the exteroceptive cueing process which separates the ballistic and controlled motor phases. In our experiment, the model-based recognition Hough transform method is used to determine the presence of the dock within the image [32]. (The Hough transform [20] is a method for recognizing arbitrary shapes within an image). If the dock is located, then control passes to state q_2, otherwise control returns to state q_0.

State q_2 represents the adaptive tracking algorithm which supports the controlled motion. This state remains active until either a proximity trigger moves control to state q_3 or a failure of the tracking algorithm returns control back to state q_1 for re-recognition.

State q_3 completes the final positioning using the ultrasonic sensors to judge distance. The *move-to-goal* motor schema was used to generate this motion. When the robot reaches the final parking position, state q_4 is activated to signal completion of the docking task. Any failure in this final phase that leaves the robot too close to the dock to allow any further movement is deemed fatal. Fatal errors (non-recoverable) in any of the perceptual strategies result in a transition to q_5, which would require

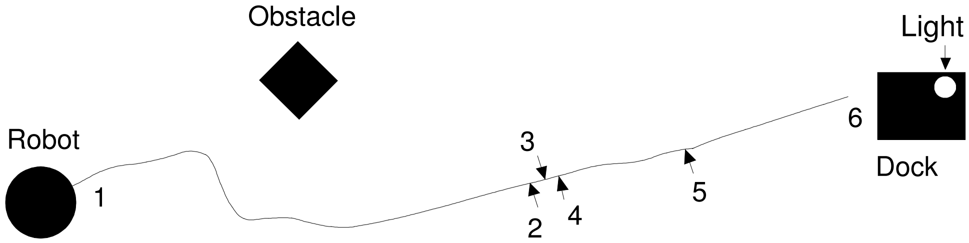
Figure 3: Experimental Set-up

invocation of the high-level planner to reconfigure the motor and perceptual strategies to cope with the unforeseen event.

4.2.2 Results

Our experimental environment is shown in Figure 3. The docking behavior is initiated with the phototropic detection algorithm being selected as state q_0 (point 1 in Figure 4). Figure 5 shows the results of the phototropic algorithm. The robot avoids obstacles as it moves towards the bright light using ultrasonic data in conjunction with the **avoid-obstacle** behavior. The robot remains in state q_0 until point 2 is reached.

State q_1 then is activated (point 2 in figure 4), based on *a priori* spatial knowledge of the world and an understanding of the motion that the robot has already undertaken [7]. The Hough transform used for model-based dock recognition [32] is invoked only when the robot is close enough to the dock to recognize it using this method (in this case approximately 15 feet plus or minus four feet). At this point, a series of five line-based models of the dock are generated from the expectations of scale and orientation as constrained by the uncertainty in the robot's position and are used within the Hough transform. All of the votes resulting from these transformations are cast within a single "compressed" Hough space in image coordinates prior to maxima location for recognition. The net result is that the model-based Hough transform procedure successfully recognizes the dock (Fig. 6).



Point	1-2	2	2-3	3	3-4	4	4-5	5-6	6
State	q_0	q_1	q_2	q_1	q_0	q_1	q_2	q_3	q_4

Figure 4: Results of Robot Run

Figure 5: Results of Phototropic Algorithm (q_0)

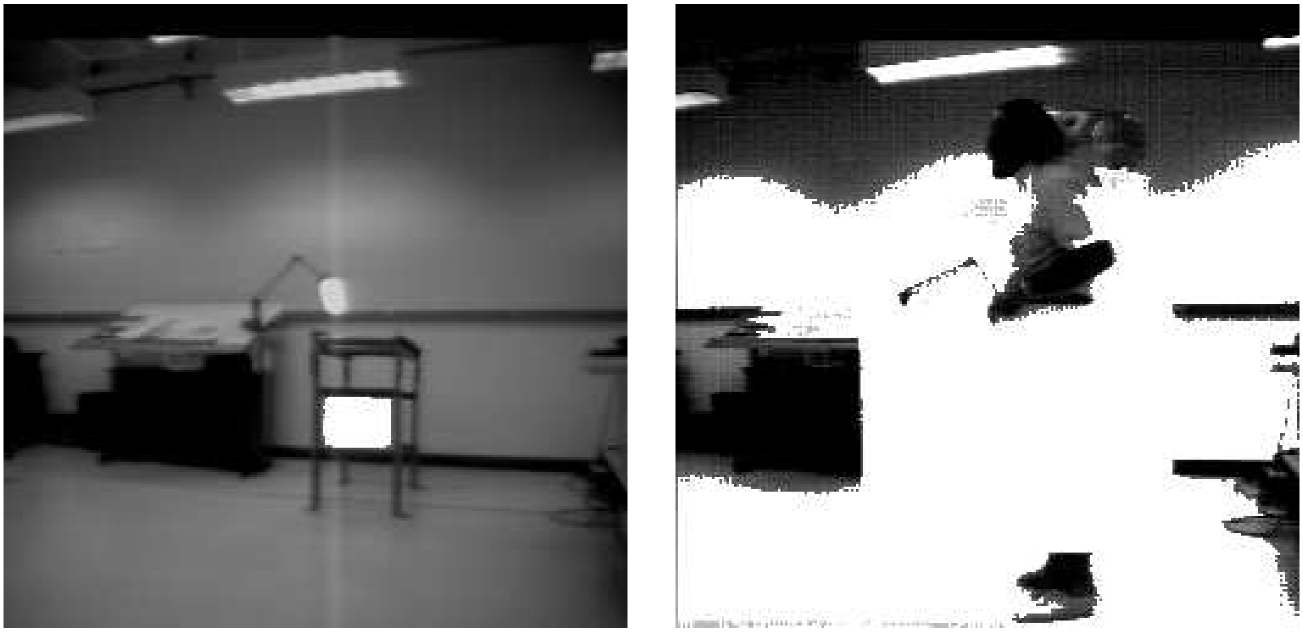
Figure 6: Successful Recognition of Dock by Hough Transform (q_1)

State q_2 is now activated and controlled motion begins using the adaptive region segmenter. The target region has been correctly located (Fig. 7a) and movement continues. In Figure 7b, the adaptive tracker has lost the target again due to occlusion by a moving object (point 3 in Fig. 4).

Control now returns to state q_1 and the Hough transform attempts to relocate the dock but the occluding object is still blocking the view. Control returns back to state q_0 (phototropic algorithm - from point 3 to 4). The robot moves a bit to obtain a new image, then reverts back to state q_1 (point 4). The occluding object has now moved out of the way, so the Hough successfully re-recognizes the dock (Fig. 8). Figure 8a shows the original dock. Lines extracted from the image (Fig. 8b) are passed to the Hough transform. In Figure 8c note that the dock was correctly located.

The system re-enters state q_2, as seen in Figure 9a, (from points 4 to 5 in Fig. 4), tracking successfully.

In Figure 9b (point 5 in Fig. 4), the vehicle is now too close to effectively use adaptive tracking any more (the trigger is based on the size of the region exceeding a predefined threshold) and control now passes to state q_3 which performs the final positioning using ultrasound (points 5 to 6 in Fig. 4). Control finally passes to state q_4 (point 6) and the vehicle stops in the final position shown in Figure 10. Other experimental runs were conducted with similar results, testing the FSA mechanism



(a) Normal region extraction (b) Region lost at point 4

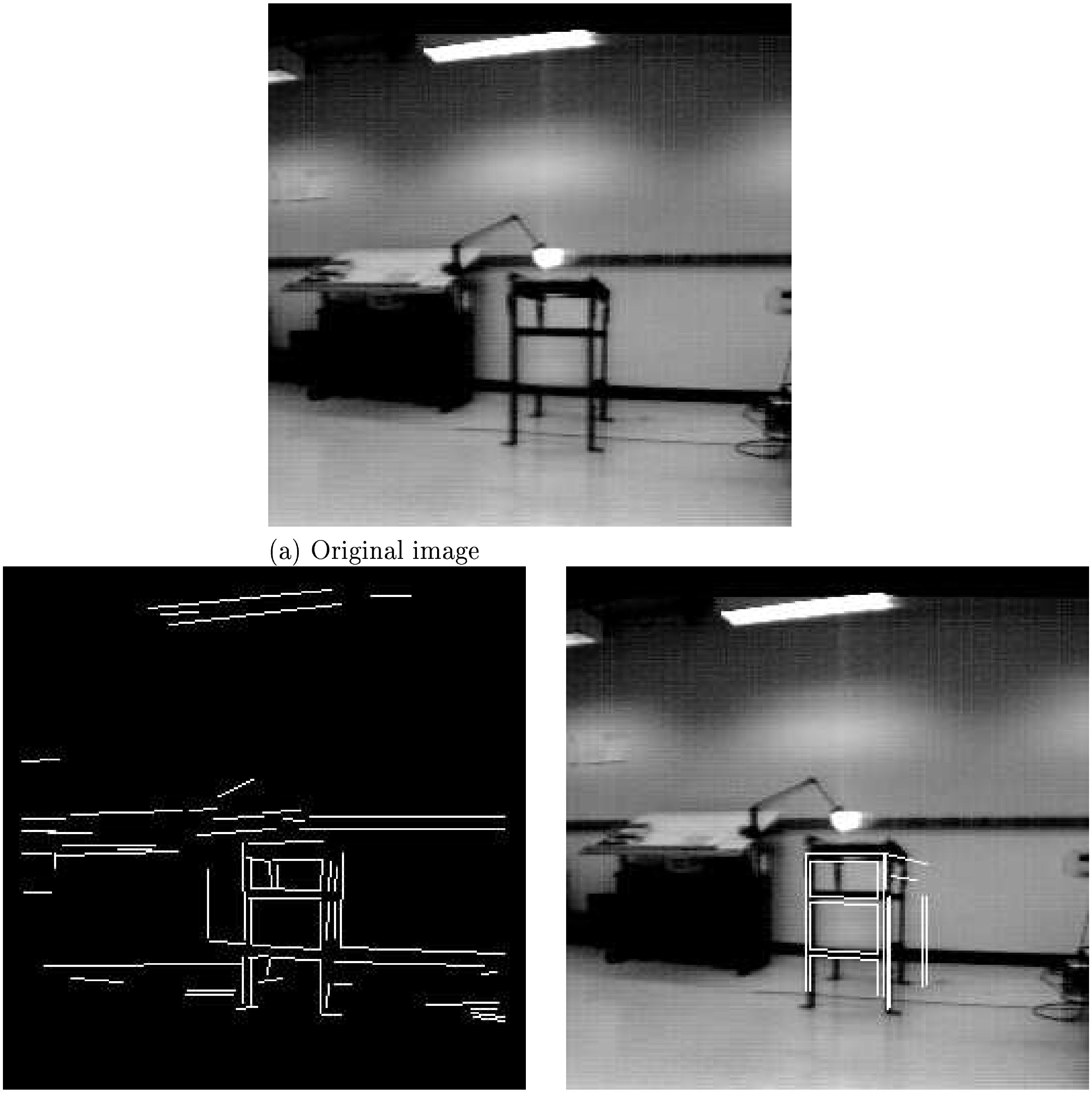
Figure 7: Adaptive tracking (q_2)

with and without various error conditions, in order to show the versatility of this method.

5. AAAI Competition Example

The Georgia Tech entry in the 1992 AAAI mobile robot competition[16] used temporal coordination of the perceptual algorithms during two phases of the contest. The competition required each robot to recognize, identify, and visit each of ten poles placed in unknown locations within an arena. Competitors were encouraged to add perceptual cues to the poles to ease the recognition requirements.

When a *move-to-pole* behavior was active, (*move-to-pole* is short for the **move-to-goal** motor schema instantiated with the **find-pole** perceptual schema), the robot located the desired pole, moved towards it and then parked next to it. This task was similar to the docking work described in the previous section. The robot utilized long range perceptual algorithms to facilitate ballistic motion and short range perception for the controlled phase of motion. Temporal coordination of the algorithms was used to activate algorithms only when they were needed and to provide robustness in the case of perceptual failures.

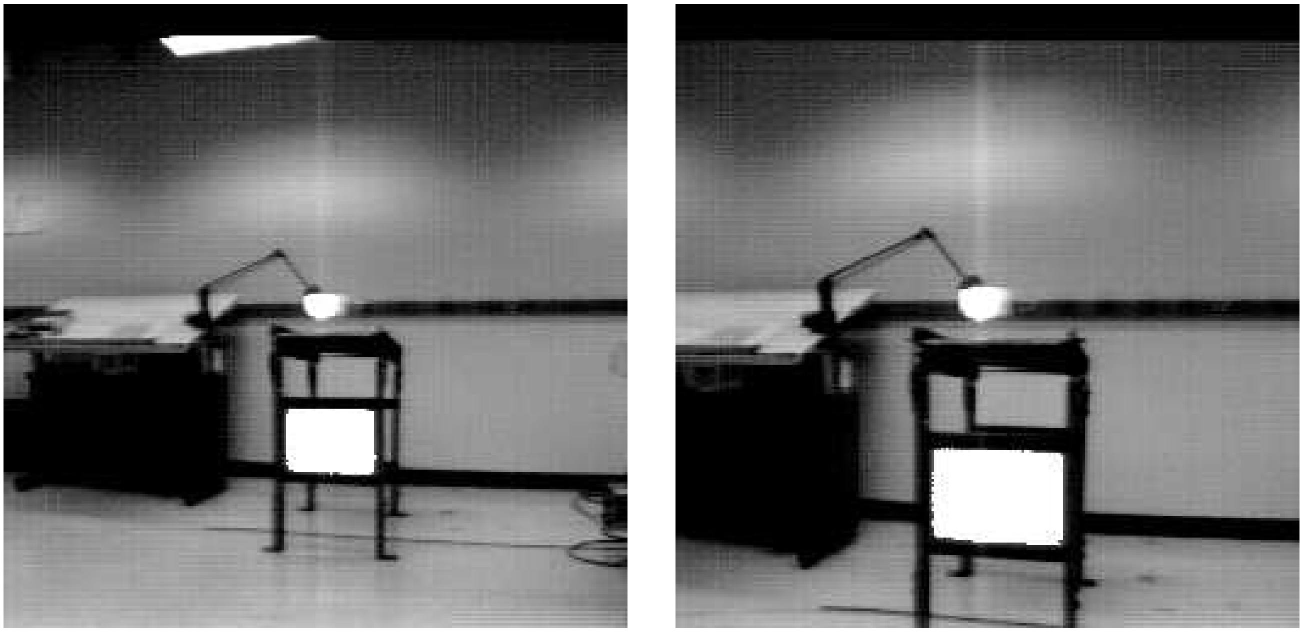


(a) Original image

(b) Extracted lines

(c) Recognized dock superimposed on image

Figure 8: Re-recognition of dock after tracking failure



(a) Normal region extraction (b) Last region segmentation at point 5

Figure 9: Re-invocation of adaptive tracking

Figure 10: Final Position of Robot at End of Run

5.1 Perceptual algorithms

The long-range perceptual algorithms are responsible for identifying the desired pole's location. The uncertainty in the predicted location is estimated to increase with distance to the pole.

5.1.1 Long-range Ballistic Cues

Two independent, long range perceptual algorithms were developed. Visual recognition of the pole provided distance, heading and the pole's code number. A second redundant system used infra-red (IR) beacons to return similar information.

Visual retro-reflective target recognition

Each pole was rigged with three rings of retro-reflective tape. A light mounted the robot illuminated the tape. The vision system determined the locations of the rings in the image. The distance between the top and bottom rings was used to recover the distance to the pole. The middle ring was placed such that it uniquely identified each pole with a code number.

IR target recognition

Each pole was fitted with a rotating IR beacon on top. A detector assembly[17] mounted on the robot reported the distance, heading, and code number when the beacon was visible.

5.1.2 Movement using position encoders

Movement to the pole was guided by the vehicle's position encoders. This information was used to create an attraction to the location for the **move-to-goal** as well as the transition to controlled motion.

5.1.3 Final Position with Ultrasound

The robot executed the controlled phase of motion, parking near the pole, using ultrasonic sensors. The robot would move towards the pole (object closest to where the pole should be) until it was within a threshold distance. It then stopped the robot and signaled success.

Figure 11: Georgia Tech Robot During AAAI Competition

5.2 Behavioral Design

Figure 11 shows the Georgia Tech robot during the competition [9]. Some destination poles are visible in the image as are cardboard boxes used for obstacles. Three retro-reflective strips and a rotating IR beacon are mounted on each pole along. The light used to illuminate the reflective strips, the video camera, and the IR detector are mounted on the top of the robot.

A schema assemblage was created that controlled the robot's high level performance. This module was responsible for searching for poles, selecting a target pole from the list of candidates, and instantiating the *move-to-pole* behavior. The operations of the assemblage were implemented as the finite state acceptor shown schematically in Figure 12. The perceptual coordination system is represented in the dotted circle surrounding the *move-to-pole* operation.

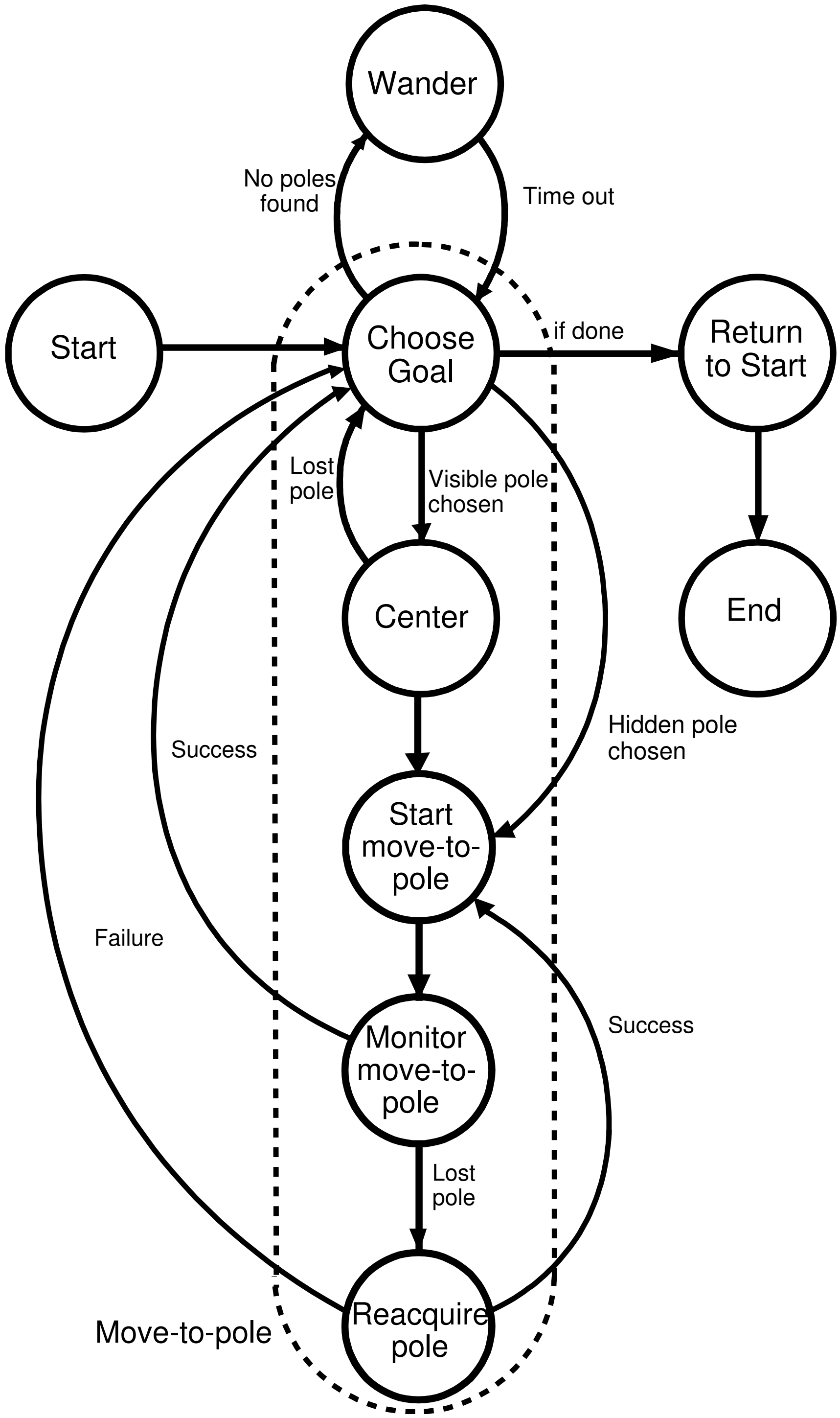


Figure 12: Schema Assemblage FSA

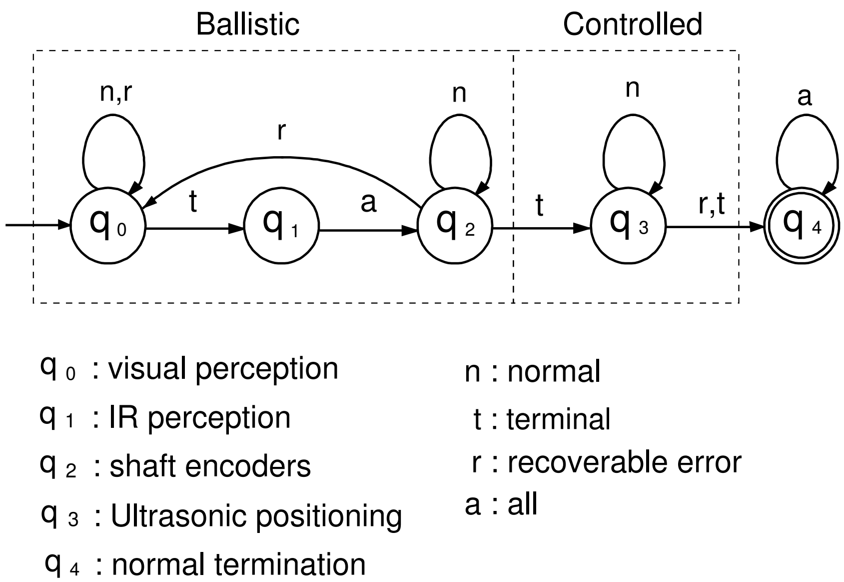


Figure 13: Move-to-pole FSA

5.3 Move-to-pole FSA

Figure 13 shows the FSA used in the temporal coordination of the perceptual algorithms for the AAAI competition. Formally, the FSA is represented by the quadruple

$$(\{q_0, q_1, q_2, q_3, q_4\}, \delta, q_0, \{q_4\})$$

where q_4 represents normal successful termination and there is no abnormal termination state. The transition function, δ, is specified by:

δ	normal	terminal	recoverable error
q_0	q_0	q_1	q_0
q_1	q_2	q_2	q_2
q_2	q_2	q_3	q_0
q_3	q_3	q_4	q_4
q_4	q_4	q_4	q_4

State q_0 represents the visual target recognition perceptual algorithm. The robot actively searched for a given pole by sweeping the camera, stopping when it found the pole. This is represented in the diagram as the recoverable error path. Each transition rotated the camera one half the field of view and took a new image. Once the pole was located, the camera was servoed to center the pole horizontally in the field of view. This increased directional accuracy without requiring calibrated camera optics. This centering process is represented by the normal transition. Once the pole was accurately located, a terminal transition was made to state q_1.

State q_1 represents the IR beacon target location perceptual algorithm. Once the robot was pointing towards a pole, the IR detection algorithm was activated. The robot would then wait long enough that the rotating beacon should have been visible. If a reading was obtained, then it was combined with the visual information determined in state q_0 to increase accuracy. Otherwise, the unmodified visual information was used. In either case, a transition was made to state q_2.

State q_2 represents movement of the robot, using the shaft encoders, to the pole. Using the knowledge of where the pole was located, the robot would move towards that location. Motor schemas active during this process included *move-to-goal*, *avoid-static-obstacle*, *probe*, *avoid-past*, and *noise*. These all combined to generate the overt *move-to-pole* behavior.

While in state q_2, shaft encoders were used to monitor the robot's progress. The motor-schema manager's monitor cycle is represented by the normal transition. If the robot did not get to the pole before a timer expired, movement was halted. The pole would then be re-acquired to confirm its location, thereby removing spatial uncertainty that had built up during movement. This process is represented in the recoverable error transition back to state q_0. When the robot arrived within a threshold distance of the pole, ballistic motion was terminated and a transition to controlled motion took place. This is shown in the terminal transition to state q_3.

State q_3 represents final positioning using ultrasonic sensors. The *move-to-goal* motor schema was active in this state. The robot moved towards the pole until it reached the desired distance. This final movement is represented by the normal transition. When it reached the desired position a terminal transition was made to state q_4. If an error occurred during this phase, such as loosing the pole, no attempt was made to recover since the robot would be too close to the pole to discern it. Instead, a transition was made to state q_4 in both the success and failure case.

State q_4 represents the final state, when the robot has reached the goal. Activation of this state re-invoked the behavior assemblage.

5.4 Experimental Results

The system proved highly robust in numerous test runs and during phase 2 of the competition. In the competition itself, it successfully found 8 of the 10 poles in less than the allowed 20 minutes, placing second in a field of ten at the end of this day.

There are some areas remaining for future work which would enhance the system even further. The *move-to-pole* FSA, as implemented, neglected to handle fatal errors.

A more complete implementation must deal with all types of failures. Also, the IR beacons had a very narrow beam width. This required rotating them at very slow rates to allow the IR detector to sample the beam as it swept by. The slow rotation required sampling many times before it could be determined that a beacon wasn't in the field of view. Rapid beacon detection would allow running the vision and IR algorithms in parallel, instead of the current serialization, increasing robustness and speed.

6. Summary and Conclusions

In order to effectively produce robust navigational behavior over a wide range of circumstances, it is essential to provide a means of coordinating perception under different circumstances. In this article a means for expressing perceptual sequencing in a robotic system has been expressed, developed, and tested. Finite state acceptors are used to represent perceptual algorithms and the temporal relationships between them. State transitions occur in response to a variety of trigger events, including normal and abnormal termination, time-outs, spatial constraints, and other related conditions. This systematic means for expressing the temporal sequencing of these algorithms in a more formal manner has utility for a wide range of robotic applications.

Mobile robot docking and the AAAI competition system were chosen as illustrative examples. For docking, four different perceptual algorithms involving vision and ultrasound have been used in the example test run, demonstrating the principles described in this article. The competition implementation used three different perceptual algorithms utilizing vision, IR beacons, and ultrasound.

This methodology is fully consistent with our action-oriented perception approach to reactive navigation. Indeed it can be fully integrated with specialized sensor fusion methods to provide a broad range of perceptual control for robotic systems.

Acknowledgments

The Mobile Robot Laboratory is supported in part by the National Science Foundation under grants #IRI-9113747 and #IRI-9100149, and the CIMS program and Materials Handling Research Center of Georgia Tech. The authors would like to thank Robin Murphy and David Vaughn for their role in developing computer vision algorithms used in the docking research. The authors also thank the members of the Georgia Tech 1992 AAAI robotics team for all their effort, the AAAI organization for supporting the team, and Denning Mobile Robotics for loaning the robot and IR beacon system used for the team's entry.

REFERENCES

1. Arbib, M.A., Kfoury, A.J., Moll, R.N., *A Basis for Theoretical Computer Science*, Springer-Verlag, N.Y., 1981.

2. Andresen, F.P., Davis, L.S., Eastman, R., and Kambhampati, S., "Visual Algorithms for Autonomous Navigation", *Proc. IEEE Int. Conference on Robotics and Automation*, St. Louis, Mo., pp. 856-861, 1985.

3. Arkin, R.C., "Motor Schema-Based Mobile Robot Navigation", *International Journal of Robotics Research*, Vol. 8, No. 4, August 1989, pp. 92-112.

4. Arkin, R.C., "Integrating Behavioral, Perceptual, and World Knowledge in Reactive Navigation", *Robotics and Autonomous Systems*, 6 (1990), pp. 105-122.

5. Arkin, R.C., "The Multiple Dimensions of Action-Oriented Perception: Fission, Fusion, Fashion", working notes *AAAI 1991 Fall Symposium on Sensory Aspects of Robotic Intelligence*, Monterey, CA, Nov. 15-17, 1991.

6. Arkin, R.C., "The Impact of Cybernetics on the Design of a Mobile Robot System: A Case Study", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 20, No. 6, Nov/Dec 1990, pp. 1245-1257.

7. Arkin, R.C., "Spatial Uncertainty Management for a Mobile Robot", *International Journal of Approximate Reasoning.* Vol. 5, No. 2, March 1991, pp. 89-121.

8. Arkin, R.C., Working Notes, *AAAI 1989 Spring Symposium on Robot Navigation*, "Towards the Unification of Navigational Planning and Reactive Control", Stanford University, CA, March 28, 1989.

9. Arkin, R.C., Balch, T., Collins, T., Henshaw, A., MacKenzie, D., Nitz, E., Rodriguez, R., and Ward, K., "Buzz: An Instantiation of a Schema-Based Reactive Robotic System", *Proc. International Conference on Intelligent Autonomous Systems: IAS-3*, Pittsburg, PA., Feb. 1993, pp. 418-427.

10. 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.

11. Arkin, R.C., Murphy, R.R., Pearson, M. and Vaughn, D., "Mobile Robot Docking Operations in a Manufacturing Environment: Progress in Visual Perceptual Strategies", *Proc. IEEE International Workshop on Intelligent Robots and Systems '89*, Tsukuba, Japan, pp. 147-154, 1989.

12. Ayache, N. and Faugeras, O., "Building, Registrating, and Fusing Noisy Visual Maps", *Proc. Int. Conf. on Computer Vision*, 1987.

13. Brooks, R., "A Robust Layered Control System for a Mobile Robot", *IEEE Jour. of Robotics and Auto.*, Vol. RA-2, No. 1, 1986 pp. 14-23.

14. Brooks,V.B., Cooke, J. D., and Thomas, J. S. "The Continuity of Movements," *Control of Posture and Locomotion*, edited by R. B. Stein, et al, Plenum Press, pp. 257, 1973.

15. Connell, J., *Minimalist Mobile Robotics: A Colony-style Architecture for an Artificial Creature*, Academic Press, 1990.

16. Dean, T.L. and Bonasso, R.P., "The 1992 AAAI Robotics Exhibition and Competition", *AI Magazine*, Spring 1993, Vol. 14 No. 1, AAAI, Menlo Park, CA, pp. 35-48.

17. Denning Mobile Robotics Inc., 21 Concord St., Wilmington MA, 01887, "Beacon Navigation System Manual", 1992.

18. Fok, K.-Y., and Kabuka, M.R., "An Automatic Navigation System for Vision Guided Vehicles using a Double Heuristic and a Finite State Machine", *IEEE Transactions on Robotics and Automation*, Vol. 7, No. 1, Feb. 1991, pp. 181-188.

19. Gat, E., "Robust, Task-Directed, Reactive Control of Autonomous Mobile Robots", *Ph.D. Dissertation*, Virginia Polytechnic Institute and State University, 1991.

20. Hough, P.V.C., "Method and Means for Recognizing Complex Patterns", U.S. Patent 3,069,654, Dec. 18, 1962.

21. Howarth, C. I., Begg, W. D. A., and Bowden, J. M. "The Relationship between Speed and Accuracy Aimed at a Target," *ACTA* Psychological Review, vol. 35, 1971, pp. 207-218.

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

23. Kluge, K. and Thorpe, C., "Explicit Models for Robot Road Following", IEEE Conference on Robotics and Automation, Scottsdale, AZ, May 1989, pp. 1148-1154.

24. MacKenzie, D. and Arkin, R.C., "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.

25. Moravec, H., "Sensor Fusion in Certainty Grids for Mobile Robots", AI magazine, Vol. 9, No. 2, 1988.

26. Murphy, R.R., "A Control Scheme for Sensor Fusion for Navigation of Autonomous Mobile Robots", Proceedings of *SPIE Sensor Fusion III*, Cambridge, MA, Nov. 1990.

27. Murphy, R.R. and Arkin, R.C. "Adaptive Tracking for a Mobile Robot", *Proc. Fifth IEEE International Symposium on Intelligent Control*, Philadelphia, PA, pp. 1044-1049, 1990.

28. Payton, D., "An Architecture for Reflexive Autonomous Vehicle Control", *IEEE Conf. on Robotics and Auto.*, p. 1838, 1986.

29. Tachi, S. and Komoriya, K., "Guide Dog Robot", *Robotics Research, Second Int. Symposium*, ed. Hanafusa and Inoue, MIT Press, pp. 333-340, 1985.

30. Slack, M.G., "Situationally Driven Local Navigation for Mobile Robots", *JPL Publication 90-17*, Jet Propulsion Laboratory, Pasadena, CA, April 1990.

31. Tsai, W., and Chen, Y., "Adaptive Navigation of Automated Vehicles by Image Analysis Techniques", *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 16, No. 5, Sept. 1986, pp. 730-740.

32. 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.