Computer vision for CARMEL

Marcus J. Huber, Clint Bidlack, David Kortenkamp,
Kevin Mangis, Doug Baker, Annie Wu, Terry Weymouth

Artificial Intelligence Laboratory
The University of Michigan
Ann Arbor, Michigan 48109–2110

ABSTRACT

In this paper we discuss the implementation and uses of the object recognition system used for CARMEL, the University of Michigan's winning entry in the AAAI–92 Autonomous Robot Competition. Following the rules of the competition, the robot was required to navigate within a large, unstructured environment performing exploration, and then a directed search, for objects placed throughout the arena. CARMEL was completely autonomous and performed these tasks, in part, using computer vision techniques. The tasks required of the computer vision system consisted of actively searching for objects (four inch diameter tubes marked with black and white stripe patterns), detecting them in images, uniquely identifying each object based upon its distinguishing pattern, and determining each object's position from orientation and distance estimates measured from the image. We briefly describe the design of the various computer vision algorithms that were developed to perform these tasks. Because of the accuracy and robustness of the vision system, we were able to perform absolute positioning, where the robot accurately updated its position through backward triangulation from previously located objects. The success of CARMEL stemmed largely from the use and implementation of the vision system to perform the tasks listed above. Other teams chose to approach these same tasks using different sensory systems and/or techniques. We analyze the general approaches, looking at where they excelled and failed, in terms of their actual performance and in general, perhaps giving insight into how to build autonomous robots that can successfully operate in "natural" environments.

1 INTRODUCTION

The University of Michigan's robot CARMEL (for Computer Aided Robotics for Maintenance, Emergency, and Life–support) was the winning entry in the 1992 National Conference on Artificial Intelligence's Autonomous Robotics Competition, sponsored by the American Association for Artificial Intelligence. Ten entries competed against each other in a three stage contest spread over three days. Following the rules of the competition, each robot was required to navigate within a large, unstructured environment, avoiding all obstacles placed in the environment, and performing exploration, and then a directed search, for "objects" also placed throughout the environment.

For large robots, the environment consisted of an octogonally shaped arena approximately seventy feet across. Small robots operated in an environment approximately half that size. Clusters of cardboard boxes were distributed throughout this space to act as obstacles. Also distributed in the arena were ten poles upon which to fasten the "objects" that each team was to find in the exploration and mapping phase of the competition. Each of the teams were permitted the opportunity to design their own objects so as to not handicap any team by imposing perception characteristics, such as having to use a particular sensing modality. A complete description of the competition guidelines, competitors, and results can be found in [4]. In short, Stage 2 of the competition consisted of exploring the arena looking for ten objects within a 20 minute time limit; Stage 3 consisted of performing a directed search for three of the objects found in Stage 2, in the order specified by judges at the beginning of the run.

The structure of the paper is divided primarily into two sections. The first will describe our particular implementation of the sensing systems, particularly the computer vision system, and we will explain the various design issues that we faced, and our solutions to them. The second section will analyze our approach with that of the other teams in an attempt to discover why we performed so well and the other entries not quite so well.

2 CARMEL

2.1 General description

CARMEL is based upon a commercially available Cybermotion K2A mobile robot platform. It is a cylindrical robot about a meter in diameter, standing a bit less than a meter high when equipped with a large hollow shell (for holding electronics and other equipment) on top. It has a top speed of approximately 800 mm/sec and is driven by three synchronously driven wheels. CARMEL's hexagonal top is decoupled from these wheels, so that the orientation of the top is unchanged when the robot itself turns.

While research has been conducted on CARMEL for the past five years, this research has concentrated on the development of a fast, robust, and reliable sonar–based obstacle avoidance system. CARMEL was totally lacking any other sensing or high level planning system until work began on the robot competition in January of 1992. For a more thorough description of the design and architecture of CARMEL, see [5].

2.1.1 Sensors

CARMEL's current suite of sensors are:

- Odometry – Wheel encoders maintain the robot's position and orientation. Errors in the estimation of the distance traveled accumulates relatively slowly, and have a small impact on CARMEL's uncertainty in its position. Angular errors also accumulate slowly but have a *great* affect upon the accuracy of the robot's knowledge of where it is.

- Sonar – There are 24 ultrasonic sensors evenly distributed around CARMEL's torso. These sensors have a range of approximately 2 meters, each scanning a cone about 30 degrees wide. The minimum firing cycle for the set of sonar sensors is approximately 160ms. Crosstalk and external noise were detected and filtered for greater reliability.

- Vision – A grayscale CCD camera was added to CARMEL to give it visual capabilities. While computer vision can be very time consuming and difficult, the long range sensing benefits far outweighed any disadvantages. The vision routine, in conjunction with the object tags that we designed, allowed CARMEL to see the objects between 1 and 12 meters away, in a field of view (FOV) of about 55 degrees, with high accuracy and reliability. The camera is mounted on a 50cm tower that is mounted on a computer controlled rotating table. This table is mounted on the top base of the robot; the top base does not rotate with CARMEL's base. This decouples the motion of the camera from the rotation of the robot. The camera mounted on the tower allows CARMEL to see over the tops of the obstacles, resulting in unobstructed views of the arena and of all objects located in the arena (and of any possible false objects outside of the arena as well).

2.1.2 Processing

All processing is done on board CARMEL. Three computers work cooperatively while the robot is running. These consist of: an IBM PC clone running a 33 MHz, 80486–based processor, which performs all of the the top level functions of the system such as vision processing, planning, absolute positioning, building and maintaining the sonar–based occupancy map, etc.; a motor control processor (Z80) receives motion and steering commands from the top level computer, controls the robot's wheel speed and direction, and maintains the robot's odometry information; and an IBM PC XT clone which is dedicated to the sonar ring, controlling the firing sequence and filtering sonar crosstalk and external noise from the sensor data. Having all processing on board allowed CARMEL to navigate at high speeds while smoothly avoiding obstacles. This is in stark contrast to most of the other teams (all except CARMEL and SRI's Flakey) in the competition that were sending information to external processors. These robots operated in a jerky, stop–and–go fashion waiting for sensor information to be sent offboard for processing.

2.2 Sonar subsystem

The ultrasonic system on CARMEL consisted of a ring of 24 sonar sensors evenly distributed about the robot's torso. A separate processor manages the firing sequence of the sensors and the subsequent filtering of noise and crosstalk (see [2] for a detailed description of the design of EERUF, the error eliminating rapid ultrasonic firing system designed by Borenstein and Koren). This firing sequence and filtering process allows CARMEL to rapidly fire and sample the ultrasonic sensors for fast obstacle avoidance. The implemented version on CARMEL permits a firing rate 2 to 5 times faster than that of most conventional sonar implementations.

One of the most popular approaches to obstacle avoidance is based on the principle of potential fields. However, in the course of experimentation with this method, Koren and Borenstein found that at higher speeds potential field methods will inherently cause oscillations when traveling near obstacles or in narrow passages. To overcome these problems, Borenstein and Koren developed an obstacle avoidance method called the vector field histogram (see [1] for a thorough description of VFH). The VFH method uses a two dimensional Cartesian grid, called the *histogram grid*, to represent data from ultrasonic (or other) range sensors.

The combination of EERUF and VFH is uniquely suited to high speed obstacle avoidance; it has been demonstrated to perform reliable obstacle avoidance in the most difficult obstacle courses at speeds of up to 1.0 m/sec.

2.3 Vision

The ability to accurately detect and identify objects in the world was important for earning the maximum number of points, as well as for keeping position and orientation errors within tolerable limits. Consequently, the system was designed from the onset to be reliable, accurate, and fast. Obviously, all these characteristics are desirable for any computer vision system. However, this is often not true of many implemented systems. Various object identification schemes were considered, but a vision based system had an important advantage in its potential for long range sensing. A major concern was the inherently heavy computation generally required for image processing. However,

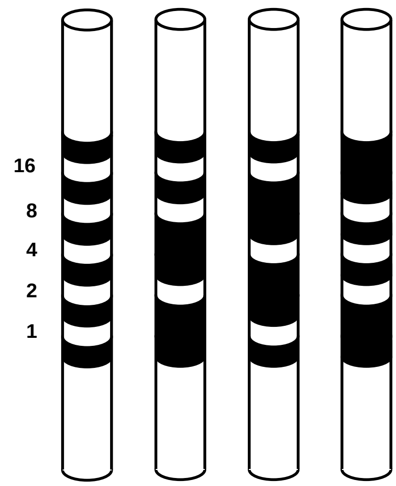

Figure 1: Example object tags showing the basic pattern (left) and the patterns for some of the objects used (bit pattern 5, 10, and 17).

by intelligently designing the object tags, the computation was greatly reduced. We succeeded because our focus was on the objective, not the method. In other words, the algorithm did not drive our development, we let the desirable characteristics, given the task at hand, drive the design and development.

2.3.1 Object design

The object tag design used for CARMEL consists of a black and white stripe pattern placed upon PVC tubing with a four inch diameter. The object tags that were used are shown in Figure 1. The basic stripe pattern is six, evenly spaced horizontal black bands of 100 mm width, with the top of the top band and the bottom of the bottom band spaced 1000 mm apart. The white gaps between the black bands correspond to the bit positions in a five bit word. A white space between two bands corresponds to an "off" bit, while filling the space with black corresponds to an "on" bit. The five bits between the six bands can then represent 32 unique objects. One of the most significant aspects of the striped PVC tags is that they are omnidirectional, appearing the same from all directions. This had a great impact upon the exploration algorithm used, as the robot did not have to approach objects from a particular direction. Rather, CARMEL had only to get within visual range of an object to perform identification.

2.3.2 Issues

The commitment to computer vision for object identification and localization introduced many issues, such as:

- Reliability – The image processing system should minimize erroneous object sightings, such as: 1) seeing objects where there are none (e.g. a spectator wearing a shirt or blouse with a stripe pattern similar to that of the poles), 2) not seeing objects when the robot should have (e.g. due to poor lighting, occlusion, etc.) 3) and misidentifying an object as another object (e.g. when the object is very far away and the stripe pattern begins to become indistinguishable).

- Determinism/Robustness – The processing time should have little variance from one run to another under various conditions. The vision algorithm should have a predictable response time regardless of whether there are zero or ten objects in an image. Also, the vision process should be robust. Unexpected or unusual conditions such as bad lighting, motion in the background due to spectators, etc. should not cause the system to respond erratically.

- Accuracy – The error associated with the calculated orientation and distance to objects should be minimized. Smaller errors in these calculations correspond to a lesser degree of uncertainty in an object's position, lifting some of the burden from the planning system that has to deal with uncertainty from other sources as well.

- Speed – Vision processing can be notoriously slow. Due to the 20 minute time limit for the second stage, and the bonus for being the fastest entry to accomplish the directed search, it was imperative that the vision process be fast.

- Range – The more information the vision system can provide to the robot, in terms of a greater number of object identities and locations due to increased range, the better off CARMEL is in the long run. This allows for more effective exploration of the environment, allowing the high level planner to better select subgoals.

2.3.3 The algorithm

Object identification and localization is performed with a single pass, gray scale, vision algorithm. About two seconds of processing time is required per image. Each column of the image is scanned for a legal sequence of object bits. Since we had only ten objects to identify, all of the object codes possible with the five bits were not used. While processing a column for potential objects, those bit patterns that are illegal are thrown out of consideration.

Similarly, any potential object whose band widths and size ratios violates the object tag design also gets thrown out of consideration. The algorithm scans a column of the image, looking for high intensity gradients, either light to dark (negative) or dark to light (positive). If a negative transition is found, the algorithm starts a potential object. Upon detecting the first positive transition after this the band width of the previous black band is calculated. Continuing, the algorithm then looks for the next negative transition. Upon detecting one, the band width of the white band is calculated, and another potential object is identified. The white band width is then compared to the black bandwidth. If the ratio of the sizes of these bands is not correct, within a certain allowable error, the algorithm rejects the object. The process continues until the bottom of the image is found or an object has five bits, indicating that it is complete. If there are still potential objects when the bottom of the image is found, all of the incomplete objects are rejected.

Once a column is complete, the objects found in the column are heuristically merged with objects found in previous columns. Objects are slowly "grown" in this fashion, until an object's edge is found, and no more columns are merged into it. The heuristics used simply look at the bit patterns to make sure that they are identical, the band widths are very similar, and the objects' tops and bottoms are in very similar positions in the image. Once the entire image is processed, another heuristic merging process is invoked that merges multiple segments of an object that happened to slip through the initial merging algorithm (an object may be initially detected as several adjacent thin objects with the same bit pattern). The heuristics used at this point check that the object's bit patterns are identical, the object segments are not too far apart in the image, and the top and bottoms are in very similar locations in the image.

The distance between the top of the top band and the bottom of the bottom band, in terms of the number of pixels in the image, is then used to estimate the actual distance from the camera to the object. The location of the object on the image plane is also used to calculate the orientation of the tube from the robot. The distance and orientation are then converted to cartesian values as an estimate of the object's position.

The general idea of the algorithm is that the number of constraints imposed by the design of the object tag will eliminate everything in the image except the objects themselves. We decided that the object tag design would be much better if we did not use a four bit code, the minimal number of bits to encode the ten objects. The five bit code allowed us to use bit patterns that did not have adjacent "on" bits, which would have created large black sections on the object, which would decrease the number of positive and negative transitions in the object.

One unexpected problem was that moving objects in the scene tend to create objects that look similar to the black–and–white band sequences that the algorithm searches for, resulting in significant added computation. The interlaced scanning of the CCD cameras electronics require 1/60th of a second between even and odd scan lines, and the boundaries of moving objects create an interlacing of bright and dark bands one pixel wide. These readings are initially flagged as potential objects that force additional heuristic processing. While the algorithm removes most of the false readings using the heuristics described above, these heuristic computations grow exponentially relative to the number of objects detected during the single pass algorithm. It happens that over one–hundred false objects can be introduced due to motion in the image, making the vision system extremely slow in such instances (sometimes running for several minutes). The main issue here is that of **determinism**, and perhaps **robustness** if the computers memory resources become exhausted.

To avoid this extra heuristic processing, a band-width constraint was added, requiring at least two pixel wide object bands before acceptance; this added computation is merely a function of the image resolution, and the processing time is deterministic. Theoretically, this decreases the effective distance of identifiable objects. In practice however, this is almost never the case because it is very rare for all of the black bands to be discernible when they are only one pixel wide on the image plane. Usually only a few object bands are noticeable at such distances. The minimal band width for effective recognition tends to be two or three pixels, with a maximum identification range of approximately 12 meters.

The absence of filtering or preprocessing of the image helped to reduce the system's **speed**, and just as important, helped to increase the system's **accuracy**. Image preprocessing tends to reduce the overall accuracy by combining data (locally) and filtering the image data. An important observation, then, is that image preprocessing is not always desirable, as in this application.

Once objects were located, their locations were recorded so that the robot could visit them. A global map, distinct from the sonar based VFH map, was created to record object locations. This map used the same coordinate system as the VFH map, however, to simplify path planning (a low level routine used to get between locations while avoiding known obstacles in the VFH map).

Although the CARMEL team experimented with preprocessing methods, the final vision system had no image preprocessing. This would have added complexity to the system and slowed down the vision process and, hence, the entire process of exploration. Color image processing methods were also explored. The red and green components of a color camera were used to segment bright orange bands from background imagery. However, the additional overhead of processing two images instead of one was inefficient, and a gray scale vision system was preferred because of reduced processing and simplicity. The algorithm had proven to be quite immune to most background noise and color vision methods deemed unnecessary. **Reliability** and **Speed** were the main issues here.

2.4 Vision system applications

To be able to find all the objects, CARMEL needed to explore the entire arena, perhaps redundantly. There was no prior information about the object locations, requiring a general and thorough exploration methodology. Due to the 20 minute time limit, CARMEL was designed to perform the task as quickly and efficiently as possible. Redundant and/or superfluous activity had to be minimized in order to accomplish Stage 2 within the time limit. Expensive computation, such as image processing, had to be optimized, and its use minimized.

The final version of the vision system could identify objects up to a 12 meter distance. Such a large visual range greatly reduced the amount of motion around the environment required to cover the area visually. From a single location near the center of the arena, it was possible to see *all ten* of the objects, although it was possible for some objects to occlude others. Therefore, CARMEL simply took a 180 degree vision sweep at the start of the run, moved across the center of the arena to a point slightly past the center, and then took a full 360 degree sweep. In the event that these two vision sweeps did not see all the objects, four additional vision locations were defined to form a square roughly 8 meters on a side, centered within the arena. At each of these locations, another 360 degree vision sweep could be performed, if needed. In actual competition, only the first two vision locations were needed.

Experimentation showed the vision algorithm to be very robust and accurate in determining an object's location, even though distance and angular orientation were estimated in a single image. Empirical results showed the distance estimates to be approximately two percent of the distance to the object, or about two centimeters for every 10 meters distance. Error in angular estimates was typically only one or two pixels, or about 0.2 degrees. These very accurate estimates stem from the ability of the algorithm to very closely identify the true edges of the objects in the image.

During times of long distance travel, CARMEL's dead reckoning system accumulated large errors. To correct for this, throughout its exploration run CARMEL determined the locations of the objects distributed in the environment, perhaps not very precisely. It could then use these object positions to determine its global absolute position. By taking a sweep of its surroundings with the camera, and determining the angular separation between the objects found, it was then possible for CARMEL to determine its position and possibly its orientation (two objects permit location determination, three or more objects permit orientation determination as well). CARMEL uses an absolute positioning algorithm based on circle intersection (see [3] for an overview of three object absolute positioning algorithms). As it turned out, the absolute positioning code was not used in the competition because some of our last minute changes were incompatible with it.

3 COMPARISON OF PERFORMANCE

Having covered CARMEL's sensing capabilities fairly thoroughly, we can now take a look at the other robot entries. There were a total of ten entries at the competition, representing both academic and commercial organizations. All of the teams were given the same rules and regulations with which to design their robots. Included in these rules was the allowance for each team to design the objects which they would be required to locate. This allowed each team to design and optimize for any sensor modality and/or algorithm that they wished. Most teams were somewhat constrained to using sensors that were already available, most relying upon some form of vision sensing. A couple entries used laser scanning techniques, while one entry used infrared beacons, and one entry used only sonar.

The competition provided a regulated environment and a common task. We can analyze the use and implementation of these designs based, in part, upon the performance exhibited in the

Entry	Finish Exploration	Finish Directed Search	Finish Overall
Michigan	1	1	1
SRI	3	4	2
Carnegie Mellon	4	3	3
Miller/Millstein	4	4	4
Georgia Tech.	2	7	5
NASA JSC	5	5	6
IBM	7	2	7
Mitre	DNF	8	8
Brown	8	DNF	9
Chicago	DNF	DNF	10

Table 1: Competition Results

competition. For a quick summary of the entries and the respective performance, see Table 1. We *will* try to account for any differences in the robot's architectures that were not sensor related but had significant impact upon the results. We will give a brief description of each entry's design, analyzing the competition's results based upon characteristics of each entry's sensing systems. As most of the entries utilized sonar and/or infrared detectors for collision avoidance, and all performed pretty well in this regard, our analysis will focus most carefully on the object identification systems used by each entry.

3.1 The University of Michigan

As already noted, CARMEL placed first in both scoring rounds, winning the overall competition. Rapid sonar firing and excellent vision performance contributed to an incredible performance in Stage 2: CARMEL was the only robot to locate all 10 objects in the 20 minute time limit, and, in fact, did so in 9.5 minutes. In Stage 3, we again came in first place, finishing the task in 3.0 minutes, well ahead of the second place finish of 3.5 minutes.

3.2 Stanford Research International

SRI's Flakey was a custom built octagonal robot fitted with 12 sonar sensors, a ring of touch sensors, and a structured light system using an infrared laser and a CCD camera. SRI was the only team other than Brown University that did not create an artificial "object" of any sort. They used a structured light system with a range of about 6' to identify the bare object tubes, upon which the other teams placed their object tags.

Flakey operated by roaming about the world, registering straight wall segments of the arena's boundaries while looking for skinny, distinct objects with its sonar system. Whenever a candidate was found, an object pole or perhaps the corner of one of the boxes, the structured light system would be used to verify the sighting. This system worked exceptionally well, giving no false readings through the course of the competition. The candidate hypothesis algorithm itself worked very well, only giving one false signal throughout the competition. The entire object detection and identification system was very well thought out and implemented. Flakey did very well in Stage 2. However, at one point Flakey got confused about its position and identified an object that it had already found as a new object. This was an artifact of the robot having lost so much accuracy on its odometry by that point that it thought the "new" object to be far enough away from other objects to be unique. They ended up finding eight objects in Stage 2 and doing fairly well in Stage 3.

3.3 Carnegie Mellon University

Carnegie Mellon University used a Hero 2000 robot named Odysseus with the standard Hero sonar sensors, one rotating at the top and one fixed at the base, and an added camera system. The sonar information was used to construct a two dimensional occupancy map of the environment. The computer vision system detected and identified large (approximately 2' in diameter and length) tubes. The detected objects were also placed into a map for later reference. The objects themselves were designed to have two visual parts: a checkerboard pattern of black and white squares to locate the object, and a stripe pattern with which to identify the object. Running on an Sun-4 workstation, the vision algorithm took approximately 6 seconds for each image. Two cameras were used, one with a wide FOV and one with a narrow FOV. The wide angle camera had an 82.0 degree FOV with a visual range of over 21 feet. The narrow angle camera had a 28.0 degree FOV, but its range was much greater, at over 69 feet.

Despite running on a fairly simple and slow robot, they did well, finding seven objects in Stage 2. They also performed the directed search well, coming in with a third place time. This may be in part attributable to their use of a computer vision system to perform long range sensing. Due to the large size of the objects they were able to detect a large number of objects at any place in the arena. Their performance was also probably aided by the fact that they performed their tasks within the small arena, which was approximately half the size of that used by the larger robots. They were hindered, however, by their use of the Hero, which could not move as quickly around obstacles due to limited mobility.

3.4 IBM

IBM had an RWI–based platform called TJ1. Its sensor systems consisted of an array of twelve short range and four longer range infrared proximity sensors, eight sonar sensors (four looking forward and four looking to the sides), a planar rotating infrared range sensor, and a low-bandwidth video camera. The infrared and sonar sensors drove obstacle avoidance behaviors in the robot, while the rotating infrared range sensor and vision system were used for object detection and identification.

The detection of potential objects was performed using the rotating range sensor. Course resolution sweeps of the environment were made while the robot was roaming around exploring the arena. Whenever a potential object pole was sensed, the robot would take a more detailed scan in the particular area of interest. If an object pole was found, its position was saved into a map and the robot moved closer to the object. The camera would then take an image which was averaged, subsampled, and a vertical band digitized before being sent via radio modem to an offboard workstation for interpretation. The object poles were tagged with a small (3" diameter) four bit black and white binary stripe pattern.

They fared horribly in Stage 2, finding only three of the objects. From observing the robot, this poor showing seemed to be due to a problematic vision system. Not only was the image acquisition performed using an analog video camera, the resulting digitized vertical strip was then transmitted via radio modem. Teams were experiencing poor reliability and noise on their radio modems throughout the competition. They also opted for a very small object tag, in part due to constraints placed on them due to the range of their sensors (4 feet or less), which also put them at a disadvantage. For Stage 3, they relied entirely on odometry for navigation, and came in with a very good time of 3.5 minutes.

3.5 Georgia Institute of Technology

The Georgia Tech team used a Denning robot base called Buzz. Buzz was equipped with 24 ultrasonic sensors distributed evenly about its base for obstacle avoidance and an active vision system for object location and identification. In addition to this, they also had an infrared beacon system to aid in object detection. The beacon system failed to work in the convention center where the competition was held, however, due to interference with the type of lighting used.

Their vision system was unique among the competitors in that it used a small, but bright, light source mounted on the robot. Their identification scheme was based upon placing three strips of retroreflective tape on the object tubes. The top and bottom strips were placed at constant heights for all of the objects, approximately six feet apart, the bottom strip placed at camera level for easy discrimination. The third strip, was placed between the other two, permitting identification based upon the ratio of the distances between the three strips. The constant height between the top and bottom tape strips permitted distance recovery from a single image. The light source's function was to cause the tape stripes to strongly reflect the light, making them stand out from the background. The object identification system had a range of about 60 feet, with a FOV of 30 degrees.

They found eight objects in Stage 2. Buzz stopped, however, because it had misclassified two of the already found objects as new objects and thought that it was done. They were slow performing Stage 3, having trouble identifying the second pole in the designated sequence. Some of the difficulties experienced during this task were caused by RF interference with the radio modems which they used to send information from the robot to an offboard workstation that processed the information.

3.6 Dave Miller and Jacob Millstein

Miller's and Millstein's entry, Scarecrow, was unconventional in that it was built entirely from electromechanical devices such as relays, servos, and microswitches. Scarecrow performed object sensing by contacting four pairs of wires on the head of the robot, held above the drive assembly on a long, thin tubular neck. Steel wool stripes were fastened around the object poles at the correct height in the binary pattern corresponding to the object number designated for that position. As Scarecrow moved around the arena, it would periodically run into an object pole. If it hit the object correctly, each strip of steel wool would close the corresponding circuit between a pair of wires on Scarecrow's head. While exploring for objects, the robot did nothing upon detecting an object other than make a flurry of noise. In the directed search, however, running into an object was checked by decoding relays. If the pattern was correct for the object, Scarecrow was searching for it would advance a solenoid that would permit the detection of the next object in the sequence.

Entirely memoryless, Scarecrow worked entirely by performing a fast random walk around the arena, hoping that chance would bring it to the objects. It performed very well, finding seven of the objects in its wandering, and finished Stage 3 quite quickly. Scarecrow's simplistic and very specialized design demonstrated that sophisticated hardware and software were not necessary for the particular tasks given, and perhaps are not necessary for many robotic tasks in general. Scarecrow's performance did suffer quite noticeably from a lack of long range sensing, missing several objects quite narrowly on its mad exploration of the arena.

3.7 NASA Johnson Space Center

JSC relied on color vision techniques for identification of the objects, and the standard ultrasonic and infrared sensors for obstacle avoidance. They also had pressure sensors to immediately stop the robot should it run into anything. Soda–Pup, their robot, was built from a Nomad 200 mobile robot from Nomadic Technologies.

The JSC team's object recognition system was based upon a color CCD camera which constantly transmitted an image in NTSC format back to an offboard workstation for processing. The image was then color segmented into eight color regions, divided into blobs, and filtered by size. They then compared the geometric relationships between these blobs to object models. Matching relationships resulted in a positive match and the identify of the object was then determined by the particular arrangement of the blobs. Range and position estimates were computed from triangulating objects from multiple views.

One possible explanation for their sixth place finish may be their computer vision system, which seems to be rather computationally expensive. This may have caused them to have done more sitting than exploring. Also problematic for them was the fact that their transmission of the images in NTSC format added a great deal of noise and distortion to the image, requiring more sophisticated low level processing.

3.8 Mitre Corp.

Uncle Bob, Mitre Corp's entry, was based upon a Denning platform, and featured 24 ultrasonic sensors about its midsection, 6 more mounted just above the floor, and a laser target reading system. The sonar system provided the sensing necessary to perform effective obstacle avoidance while the laser target reading system located and identified objects tagged with a special reflective bar code panel.

Uncle Bob had incurred heavy damage to its laser scanning system and a drive axle while it was being shipped to the competition. Although a great deal of work was put into trying to patch up the system to handle such damage, Uncle Bob was effectively out of the competition. It was unable to complete an official Stage 2 run (it was given a chance in order to create a map of objects but it did not count toward its competition score). While performing Stage 3, it was unable to find all three of the objects that it had been directed to find and came in last place of the eight entries that completed the competition.

3.9 Brown University

Brown was another robot that had problems, as much with their undebugged system as with hardware. The team was composed primarily of a group of undergraduate students, with some assistance from graduate students. Their robot, Huey, was based upon a RWI B12 platform, and relied *completely* on eight sonar and two infrared sensors for obstacle avoidance and object detection. Brown's robot was one of only two robots entered that did not artificially mark the objects. The robot interpreted the sonar readings and created a probabilistic map of interesting features in its vicinity. It also built a node map containing interesting positions as nodes linked by free paths between them. The complexity and difficulty of their approach was their downfall, however, as they were forced to withdraw Huey from the competition after not finding any of the objects in Stage 2. It is not known why they decided to rely entirely on sonars for every aspect of the competition. It is readily apparent, however, that they placed themselves at a distinct disadvantage by doing so.

3.10 University of Chicago

The University of Chicago had the worst luck of all of the teams. Their robot Chip was another RWI–based platform. They relied on eight sonar and sixteen infrared sensors, and had added a color camera on a pan/tilt head to perform object identification and recognition. They were forced out of the competition after having their original DataCube image processing board, *and* an emergency replacement board, burn out on them. Their object design was to consist of an omnidirectional color coded sign, with an algorithm using color histogramming to be used to identify them. It would have been interesting to see how their approach worked, as only one other team, NASA's Johnson Space Center, used color vision techniques, and that with limited success.

4 CONCLUSIONS

There were three issues that seemed to be the most significant for success in the competition: long range sensing, all onboard processing, and simplicity in design. Even robots with limited mobility, such as Carnegie Mellon's Odysseus performed extremely well due to its ability to detect objects from a great distance. Robots not designed for onboard processing suffered greatly due to errors and time delays in the communication of information. Finally, simple approaches also seemed to be the correct approach, epitomized by Scarecrow's spartan design. The winning entry, The University of Michigan's CARMEL, incorporated each of these three characteristics.

5 ACKNOWLEDGMENTS

The authors wish to thank the other members of the CARMEL team. Support for the CARMEL team was provided by The University of Michigan College of Engineering, The University of Michigan Rackham School of Graduate Studies, the American Association for Artificial Intelligence, ABB Robotics Inc. and ABB Graco Robotics Inc. Marcus Huber is supported in part by DARPA grant no. DAAE-07-92-C-R012. Purchase of CARMEL and support for research using it is provided by Department of Energy grant no. DE-FG0286NE37969.

REFERENCES

[1] Johann Borenstein and Yoram Koren. The Vector Field Histogram for fast obstacle-avoidance for mobile robots. *IEEE Journal of Robotics and Automation*, 7(3), 1991.

[2] Johann Borenstein and Yoram Koren. Noise rejection for ultrasonic sensors in mobile robot applications. In *The Proceedings of the IEEE Conference on Robotics and Automation*, 1992.

[3] Charles Cohen and Frank Koss. A comprehensive study of three-object triangulation. In *SPIE Mobile Robots VII*, 1992.

[4] Thomas Dean and R. Peter Bonasso. 1992 AAAI robot exhibition and competition. *AI Magazine*, Spring, 1993.

[5] David Kortenkamp, Marcus Huber, Clare Bates Congdon, Scott Huffman, Clint Bidlack, Charles Cohen, , Frank Koss, Ulrich Raschke, and Terry Weymouth. Integrating obstacle avoidance, global path planning, visual cue detection and landmark triangulation in a mobile robot. In *Proceedings of SPIE Mobile Robots VII*, 1992.