Design of the Bearcat Cub for the Intelligent Ground Vehicle Competition 2005

Ernest L. Hall, Masoud Ghaffari, Justin Gaylor, Mei Long, Steve Climer, Robert Gilliland and Rayjan Wilson

Center for Robotics Research, Department of Mechanical, Industrial and Nuclear Engineering, University of Cincinnati, Cincinnati, OH 45221-0072

ABSTRACT

The purpose of this paper is to describe the design and implementation of an Unmanned Ground Vehicle, called the Bearcat Cub, named after the UC mascot. The Bearcat Cub is a hybrid powered, three wheeled vehicle that was designed for the International Ground Vehicle Competition and will be tested in the contest in 2005. The dynamic model, control system and design of the sensory systems is described. For the autonomous challenge line following and obstacle detection are required. Line following is accomplished with a dual camera system and video tracker. Obstacle detection is accomplished with either a stereo vision system or a laser scanner. Pothole detection can also be implemented with a video frame grabber. For the navigation challenge waypoint following and obstacle detection are required. The waypoint navigation is implemented with a global positioning system. The Bearcat Cub has provided an educational test bed for not only the contest requirements but also other studies in developing artificial intelligence algorithms such as adaptive learning, creative control, automatic calibration and internet-based control. The significance of this effort is in helping engineering and computer science students understand the transition from theory to practice in automated vehicles.

1. Introduction

Robot design generally falls into two categories: fixed industrial robots and mobile robots [1, 2]. Unmanned Ground Vehicles, UGVs, are a group of mobile robots with great promising potentials for the future. Space exploration [3, 4], material handling [5], transportation, medical transport of food and patients and future combat vehicles [6] are areas that traditionally have been emphasized and the laboratory results are beginning to find application in the real world.

A power source, manipulator, control and sensory systems are the basic elements of any UGV. This paper mainly deals with the dynamic control model and sensory systems design of an Unmanned Ground Vehicle robot, the Bearcat Cub. In addition to those described in this paper, this robot has also been used to study: internet-based control [7], automatic calibration [8], and creative control [9].

The Bearcat Cub is an interactive and intelligent Unmanned Ground Vehicle designed to serve in unstructured environments such as those encountered in the Intelligent Ground Vehicle Competition (IGVC), a contest that provides real world challenges. The Bearcat Cub was designed to perform all the tasks and obey all the rules required for this contest; however, we have found that a continual improvement is necessary because of changing contest rules and technical innovations.

This paper is organized as follows. Section 2 describes the overall block diagram of the system including the sensors and mechanical system, line following, obstacle avoidance, and waypoint navigation sensors. Sections 3 and 4 outline the performance and conclusions.

Figure 1. The Bearcat Cub robot

2. Design

2.1 Mechanical System

The Bearcat Cub was designed to be an outdoor vehicle able to carry a payload of at least 20 pounds. Optimal design was attempted using proper design practices and tools during the basic design. The robot’s frame is constructed of 80/20 TM aluminum extrusions, joining plates and T-nuts. Figure 1 shows the frame assembly view of the mechanical system.

2.2 Autonomous Challenge Line Following and Obstacle Detection

The autonomous challenge requires that the Bearcat III negotiate through an outdoor obstacle course in a prescribed time while staying within the 5 mph speed limit, traversing ramps with 10 percent incline, and avoiding both physical obstacles and painted potholes on the track.

2.2.1 Vision System

The Bearcat’s vision system for the autonomous challenge comprises three cameras, two for line following and one for pothole detection. The vision system for line following uses 2 CCD cameras and an image tracking device (I-Scan^{TM}) for the front end processing of the images captured by the cameras. The I-Scan^{TM} tracker processes the image of the line. The tracker finds the centroid of the brightest or darkest region in a captured image. The three dimensional world co-ordinates are reduced to two dimensional image coordinates using transformations between the actual ground plane to the image plane. A novel four-point calibration system was designed to transform the image co-ordinates back to world co-ordinates for navigation purposes. Camera calibration is a process to determine the relationship between a given 3-D coordinate system (world coordinates) and the 2-D image plane a camera perceives (image coordinates). The objective of the vision system is to make the robot follow a line using a camera [10] . At any given instant, the Bearcat tracks only one line, either right or left. If the track is lost from one side, then the central controller, through a video switch, changes to the other camera.

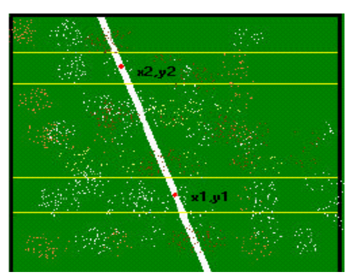

Figure 2. Two windows for two points on the line

In order to obtain accurate information about the position of the line with respect to the centroid of the robot, the distance and the angle of the line with respect to the centroid of the robot has to be known. When the robot is run in its auto-mode, two I-Scan windows are formed at the top and bottom of the image screen as shown in Figure 5. The centroids are shown as points and (x_2, y_2) in Figure 5. The angle and distance of the line to the robot are determined.

2.2.2 Obstacle Avoidance

The obstacle avoidance system is designed to detect an obstacle on the navigational course and then calls the appropriate software routine to negotiate it. Two alternative solutions one using a laser scanner and one with the sonar sensors have been used on the Bearcat for obstacle detection and avoidance. Both approaches are explained below.

2.2.2.1 Design Solution using Laser Scanner for Fine Detection

The Bearcat uses a Sick Optics^{TM} laser scanner (LMS 200 ^{TM}) for sensing obstacles in the path. The unit communicates with the central computer using a RS 232/422 serial interface card. The maximum range of the scanner is 32 meters. For the contest, a range of 8 meters with a resolution of 1^{o} has been selected. The scanner data is used to get information about the distance of the obstacle from the robot. This can be used to calculate the size of the obstacle. The scanner is mounted at a height of 8 inches above the ground to facilitate the detection of short as well as tall objects. The central controller performs the logic for obstacle avoidance as well as the integration of this system with the line following and the motion control systems. Figure 3 shows the field of view of the laser scanner.

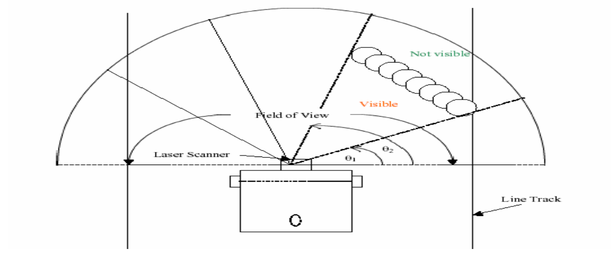

Figure 1. Field of view of the laser scanner

2.2.2.2 Design Solution using Stereo Vision and Laser Scanner

A stereo vision system and a laser scanner are used for obstacle detection. A “time of flight” approach is used to compute the distance from any obstacle. The laser transmits light waves towards the target, detects an echo, and measures the elapsed time between the start of the transmit pulse and the reception of the echo pulse. The transducer sweep is achieved by using a motor and Galil^{TM} motion control system. Adjusting the Polaroid^{TM} system parameters and synchronizing them with the motion of the motor permits measuring distance values at known angles with respect to the centroid of the vehicle. The distance value is returned through an RS232 serial port to the central controller. The central controller uses this input to drive the motion control system. The range of this system is 40 feet.

2.3 Navigational Challenge Problem and Solution

The goal of the navigational challenge is to navigate Bearcat III to a series of predefined waypoints while avoiding obstacles. For this a global positioning system (GPS) is used to get the original robot position, then tracking is used to move the robot from one point to the next, updating the new base with every pass. The laser scanning system was used to detect and avoid obstacles en route to the target waypoints. Wheel encoders on the vehicle were used to track the path navigated and make decisions about the distance to travel and angle to steer to reach a target point

2.3.1 GPS Selection

The basic criteria used in the selection of the GPS unit were:

- Embedded navigation features
- WAAS capability to improve accuracy of standard GPS signal to 3meters
- RS-232 serial port input/output to interface with robot's computer
- External antenna for accurate reception
- External power capability to ensure constant source of regulated power

Based on the above selection criteria the Garmin-76™ GPS was chosen as the unit to provide GPS navigational ability to the robot. The Garmin-76 ™ unit provides all of the above-mentioned features in addition to other features not used in the current navigation algorithm.

2.3.2 Description of Navigational Challenge Algorithm

The basic solution selected to solve the navigational challenge problem was to model the problem as a basic closed feedback control loop. This model has an input command (target waypoint destination), feedback signal (GPS unit position information), error signal, and transfer function of the output characteristics. The GPS unit uses the current position information (latitude, longitude, height and velocity information at the rate of 1 to 255 second/output) and calculates the bearing and range from the target waypoint to determine the error. Correction signals are generated to reduce the error to a certain tolerance based on the bearing angle error signal generated by the GPS unit. The correction signals consist of turn right, turn left, forward motion, or stop. These corrective commands are sent to the motion control system, which translates these commands into motor control voltages that steer and propel the robot on the course. Once the bearing angle error and target range have been reduced to the required tolerance the command is considered complete and the robot arrives at its target destination waypoint. At this point, the next target waypoint is selected and the process is repeated until all target waypoints in the database have been reached. The GPS signal is very poor inside buildings; however, the terrain is relatively flat and even. Here the data from the wheel encoders provide data for the motion control. This system is also used at the start until the velocity reaches a point that provides accurate GPS data.

2.3.3 Point to Point Navigation using Wheel Encoders

An encoder translates motion into electrical pulses, which are feed back into the Galil motion controller. The feedback is used to calculate the distance traversed. Steering is achieved by differential motion of the two wheels. The problem is modeled as a closed feedback control loop. The input command is the target waypoint destination relative to the robot position. The wheel encoder provides the feedback signal. The motion from the origin A to target B is achieved by two motions. The program calculates Angle "α" and distance "d". The robot first steers "α" units and it then traverses "d" units to reach the target.

2.3.4 Obstacle Avoidance

A single line Sick Optics™ laser scanner is used to detect and avoid obstacles. If an obstacle is detected, an obstacle avoidance routine, similar to the feedback control loop used for the GPS navigation, is used to navigate the robot around the obstacle. Once the robot avoids the obstacles, the original target waypoint is

restored and the navigational feedback control loop is resumed.
This approach has some limitations. The Bearcat has only one laser scanner. It scans a single plane at the mounted height limiting it from detecting obstacles that are either short or overhung as shown in the Figure 10. If the laser scanner is mounted at a shorter height, it will detect shorter obstacles but will lead to a different problem. When the terrain is not plane but wavy, or if there are ramps, the scanner will detect the ramp or the ground as an obstacle-giving rise to misinterpretation of the environment.

The algorithm fails to handle the case when the situation becomes dynamic. It will change its path to avoid all the obstacles it sees every time it scans the surrounding. It fails to take into account the trajectory of the moving obstacles. The alternate path taken by the robot may result in hitting the obstacle if both the robot and the obstacle move in the same direction.

The algorithm also fails when the configuration of obstacles becomes so complex such that it will not have a way to go further. In some situations, there is always an option to retrace its path and look for a different path to avoid the deadlock. The algorithm does not have any kind of memory mapping, so the robot comes to a stop when it enters a deadlock.

2.3.5 Algorithm Implementation

The physical implementation of feedback control loop of the GPS navigation consists of the Garmin 76 GPS unit, the motion control system, laser scanner, and the robot computer. Waypoint coordinates are read from the waypoints file during the initialization stage of the program and stored in an array in memory. A NMEA message is sent to the Garmin 76 GPS unit via the RS 232 port, which sets the active target waypoint in the GPS unit's memory. This is the command signal. Once set, the waypoint coordinate is used by the GPS unit to calculate bearing, track, and range to the target waypoint. The Garmin 76 GPS unit transmits ASCII data output via the RS232 port containing the bearing, track, and range to the destination waypoint. The turn angle (angle error) is related to the track angle and bearing angle by the equation: Turn Angle = Track Angle – Bearing Angle. This equation gives the turn angle in the 0 to 360 degree reference frame but this angle is transformed to zero to 180 degrees (left turn angle) or 0 to -180 degrees (right turn angle) for the robot turning subroutine. The robot turns to the commanded correction turn angle if the turn angle is greater than 6 degrees or less than -6 degrees and then moves forward until the GPS position data are updated. When the robot arrives within 5 feet of the destination waypoint, the next target waypoint is selected and this process is repeated until all targets have been reached. This process defines the discrete feedback control loop algorithm used for the robot GPS navigation course.

3. Performance

Actual performance at the contest has varied depending on the software, weather and sometimes luck. We have learned to do a Potential Failure Modes and Effects Analysis (PFMEA) and a large amount of pre-contest testing. The most predictable failure is battery discharging so a digital voltmeter is used. Other events such as hard drive crashes led to shock mounted, dual drives. Loose connections can be quickly detected with a voltmeter. A rain cover is used in wet weather. The robot was always able to compete.

4. Conclusion

Several aspects of the design and implementation of an unmanned ground vehicle were explained. A dynamic model was constructed and the robot parameters were computed to provide a basis for control and learning studies. The main sensory systems and algorithms for as line following, waypoint navigation, and pothole detection were described. The design has been proved rugged and robust in the International Ground Vehicle Competition as well as many other experiments. The UC robot team currently is using the lessons learned from Bearcat III to design a completely new robot called Bearcat Cub and is preparing to compete in the next IGVC with the new robot. Overall, this contest has been a wonderful educational experience and a great proving ground for ideas.

5. Acknowledgments

The authors would like to thank all the former and current UC robot team members listed on www.robotics.uc.edu that have contributed their time and efforts to the design and construction of the robot and leadership of the team. We also gratefully acknowledge our organizational sponsors and individual sponsors: Finally, the faculty and technical staff of the University of Cincinnati have guided us through many difficult technical problems.

6. References

[1] A. N. Goritov and A. M. Korikov, "Optimality in Robot Design and Control," *Automation and Remote Control*, vol. 62, pp. 1097-1103, 2001.

[2] J. B. Pollack, G. S. Hornby, H. Lipson, and P. Funes, "Computer Creativity in the Automatic Design of Robots," *Leonardo*, vol. 36, pp. 115-121, 2003.

[3] A. Elfes, S. S. Bueno, M. Bergerman, E. C. de Paiva, and J. G. Ramos, "Robotic Airships for Exploration of Planetary Bodies with an Atmosphere: Autonomy Challenges," *Autonomous Robots*, vol. 14, pp. 147-164, 2003.

[4] S. Hirose, H. Kuwahara, Y. Wakabayashi, and N. Yoshioka, "The mobility design concepts/characteristics and ground testing of an offset-wheel rover vehicle," *Space Technology*, vol. 17, pp. 183-193, 1997.

[5] T. Ito and S. M. Mousavi Jahan Abadi, "Agent-based material handling and inventory planning in warehouse," *Journal of Intelligent Manufacturing*, vol. 13, pp. 201-210, 2002.

[6] M. D. Letherwood and D. D. Gunter, "Ground vehicle modeling and simulation of military vehicles using high performance computing," *Parallel Computing*, vol. 27, pp. 109-140, 2001.

[7] M. Ghaffari, S. Narayanan, B. Sethuramasamyraja, and E. Hall, "Internet-based Control for the Intelligent Unmanned Ground Vehicle: Bearcat Cub," presented at SPIE Intelligent Robots and Computer Vision XXI: Algorithms, Techniques, and Active Vision, Providence, RI, 2003.

[8] B. Sethuramasamyraja, M. Ghaffari, and E. Hall, "Automatic Calibration and Neural Networks for Robot Guidance," presented at SPIE Intelligent Robots and Computer Vision XXI: Algorithms, Techniques, and Active Vision, Providence, RI, 2003.

[9] X. Liao, M. Ghaffari, S. M. A. Ali, and E. Hall, "Creative Control for Intelligent Autonomous Mobile Robots, Intelligent Engineering Systems through Artificial Neural Networks," in *Intelligent Engineering Systems through Artificial Neural Networks*, vol. 13. New York: ASME, 2003, pp. 523-528..