Journal of Cybernetics and Informatics

published by

Slovak Society for Cybernetics and Informatics

Volume 13, 2012

http://www.sski.sk/casopis/index.php (home page)

ISSN: 1336-4774

MOBILE ROBOT POSITIONING USING ODOMETRY AND ULTRASONIC SENSORS

Juraj Aziza M. Zaki , Osama Arafa, Sanaa I.Amer, and Abeer H.Bashta

Electronics Research Institute, Cairo, Egypt

Abstract

Mobile robotics is an area that deals with the control of autonomous and semiautonomous vehicles. The use of mobile robots is growing in a large number of applications especially in manufacturing, hazardous materials handling …etc In this paper, the navigation system built on a mobile robot operating in a warehouse is presented. As the mobile must be able to navigate from a known position to a desired new location and orientation avoiding any contact with fixed or moving objects while in Route, the sensory system is very important. Hybrid navigation system that combines the perception and dead reckoning is used and gives satisfactory operation .The encoder and the ultrasonic sensors used are presented in details and the navigation system designed based on their operation is illustrated.

Keywords: Mobile robot, navigation control, odometry, ultrasonic sensors

1 INTRODUCTION AND PRELIMINARIES

Navigation means the ability to wonder in the environment without colliding with obstacles, the ability to determine one's own position, and the ability to reach certain goal locations. The robot also can construct an internal representation to its environment on the form of a map. This map can be used in planning paths towards its goals locations. So, navigation system may imply the following components: robot positioning system, path planning and map building.

In order to achieve this objective, the robot needs to be equipped with sensors suitable to localize the robot throughout the path it has to follow. These sensors may give overlapping or complementary information and may also sometimes be redundant [1,2]. These are four popular positioning systems:

1. Odometry (dead reckoning) – based navigation which uses encoders to measure the rotation of the wheels and the steering orientation. The vehicle can estimate its relative position using these measures.
2. Active beacons- based navigation system where three or more actively transmitted beacons are located at known positions in the environment, and a receiver on the vehicle. The absolute position of the vehicle is computed from the measurements of the distances or angles with respect to the beacons.
3. Landmark-based navigation system which is similar to active beacons system but instead of active beacons a natural or artificial landmarks are defined.
4. Map-based navigation system: In this system, the vehicle contains a map to the environment. The vehicle estimates its position by matching the acquired information from its sensors with the environment map.

Mobile robots generally carry dead reckoning sensors such as wheel encoders and inertial sensors, also landmark and obstacle detecting and map making sensors such as time of flight (TOF) ultrasonic sensors. Sensors measurements in this case are to be fused to estimate the robot's position.

In this paper, the navigation system built on a mobile robot operating in a warehouse is presented. Hybrid navigation system that combines the perception and dead reckoning was found to be complementary and gives a satisfactory operation of the mobile robot. The position estimate provided by dead reckoning is corrected by matching the perception against a stored map. Landmark-based navigation depends mainly on the agent's perception to its environment. If the environment contains confusing information or few perceptually distinguishable landmarks, the performance of these systems decline. The perceptual aliasing problem can be solved by including the odometry data to discriminate between the similar places.

2 ODOMETRY and ODOMETRY ERRORS

Odometry is the most widely used navigation method for mobile robot positioning. It is well known that odometry provides good short term accuracy, is inexpensive and allows very high sampling rates. However, the fundamental idea of odometry is the integration of incremental motion information over time, which leads inevitably to the accumulation of errors. Particularly, the accumulation of orientation errors will cause large position errors which increase proportionally with the distance travelled by the robot. Odometry is used in almost all mobile robots, for various reasons: Odometry data can be fused with absolute position measurements to provide better and more reliable position estimation [3,4];- Odometry can be used in between absolute position updates with landmarks; -Many mapping and landmark matching algorithms assume that the robot can maintain its position well enough to allow the robot to look for landmarks in a limited area.

A. Systematic and Non-systematic Odometry Errors

The correct functioning of a mobile robot requires no faults (fault means that the robot is functioning outside its specification limits, being unable to accomplish normally its tasks). An error in orientation during the robot movement may lead to deviations (the amplitude of the deviations depends on the gravity of the error). If these deviations are causing the incapacity of the system to realize its task, then the error became a fault.

Odometry is based on simple equations that are easily implemented and that utilize data from inexpensive incremental wheel encoders. However, odometry is also based on the assumption that wheel revolutions can be translated into linear displacement relative to the floor. This assumption is of limited validity. One extreme example is wheel slippage: - if one wheel was to slip on, say an oil spill, then the associated encoder would register wheel revolutions even though these revolutions would not correspond to a linear displacement of the wheel. There are also, several other subtle reasons for inaccuracies in the translation of wheel encoder readings into linear motion. All of these error sources fit into one of two categories: systematic errors and non-systematic errors.

a) Systematic Errors:
- Unequal wheel diameters.
- Average of actual wheel diameters differs from nominal wheel diameter.
- Actual wheelbase differs from nominal wheelbase.
- Misalighnment of wheels.
- Finite encoder resolution.
- Finite encoder sampling rate.

b) Non-Systematic Errors:
- Travel over uneven floors.
- Travel over unexpected objects on the floor.

- Wheel-slippage due to: slippery floors, over acceleration fast turning (skidding), external forces (interaction with external bodies), internal forces (castor wheels) and non-point wheel contact with the floor.

Systematic errors are particularly grave because they accumulate constantly. On most smooth indoor surfaces, systematic errors contribute much more to odometry errors than non-systematic errors. However, on rough surfaces with significant irregularities, non-systematic errors are dominant. The problem with non-systematic errors is that they may appear unexpectedly (for ex., when the robot traverses an unexpected object on the ground, and they can cause large position errors.

The majority of researches are focusing on the systematic odometry errors using offline techniques based on calibrations. Also, it has to be taken into account that the mobile robot is moving in dynamic environments, where the trajectory is never the same [5].

To correct the errors in positioning resulting from the odometry system and for safe navigation and obstacle avoidance, the robot needs to be equipped with sensors suitable to localize the robot throughout the path it has to follow [6]. Because ultrasonic sensors can provide good range information based on the time of flight (TOF) principle for rather low expense, they have been widely used in mobile robot applications [7-9].

3 ULTRASONIC SENSORS

Ultrasonic transducers are preferably used to obtain three- dimensional information of the environment. Time-of-flight (TOF) ranging systems measure the round-of-trip time required for a pulse of emitted energy to travel to a reflecting object, then echo back to a receiver. Ultrasonic is typically employed. They have many advantages:- Measure and detect distance to moving objects; -Impervious to target materials, surface and color; -Solid- state units have virtually unlimited , maintenance –free lifespan and are not affected by dust, dirt or high–moisture environments.

But some problems appear in sonar response. Ultrasonic sensors suffer from unreliable sonar responses from the environment. For sonar – based mobile robot in confined space, special attention should be paid to these problems .The space is normally a closed environment.

As our concern is the navigation in confined spaces using multi sonars, we must understand why there are such unreliable readings in ultrasonic sensor responses.Two major problems are discussed in the following [10]:

A. Angular uncertainty

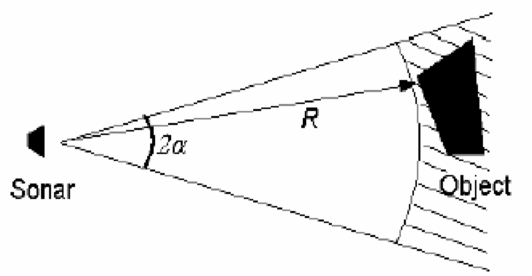

Fig .1 Angular error of an ultrasonic sensor α is the half opening angle of sonar cone, R is a sonar response

The angular uncertainty means the uncertainty in the angle information of a sonar response from a detected object. Fig.1 conveys the idea. When an ultrasonic sensor gets a range response of R meters, the response simply represents a cone within which the object may be present .There is no way to pin – point exactly where the position of the object is . As shown in Fig.1, the opening angle of the ultrasonic sensor is 2α and the object can be anywhere in the shaded region for the response R.

B. Specular reflection

Specular reflection refers to the sonar response that is not reflected back directly from the target object .In specular reflection , the ultrasound is reflected away from the reflecting surface , which results in longer range reporting or missing the detection of object all together [11,12] .

The specular reflection is due to different relative positions of the ultrasonic transceiver and the reflecting surfaces. Fig.2 shows sonar responses in two different situations. In Fig.2a, the sensor transceiver axis is perpendicular to the reflection surface, so most of the sound energy is reflected directly back to the ultrasonic sensor.

However, in Fig.2b, because the sonar transceiver is not perpendicular to the surface, much energy is reflected away. The amount of reflected sound energy depends strongly on the surface structure of the obstacle and the incidence angle [13].

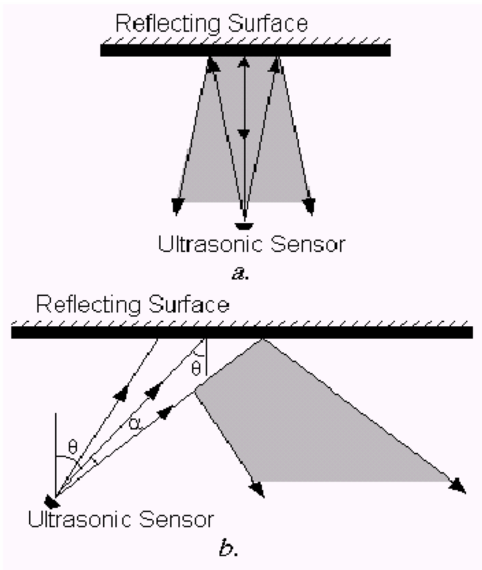

Fig.2 Specular reflections

4 ROBOT DESCRIPTION

The mechanical design for the robot plays a critical role in the success of the robot facility. The requirements of the mechanical facility of the robot is the major question before beginning the mechanical design. The following specifications should be met:-

1. Moving forward and backward without rotation.
2. Moving asides (right and left) without rotation.
3. Have the facility to rotate in a complete circle.
4. The robot will use a battery assembly with total volt 48V.
5. The control unit and battery charger should be on the robot itself.

The driving system of the robot is composed of 4 wheels each of them equipped with a separate electric motor. A front and rear steering system were added to give flexibility in the motion planning for smooth navigation. The mobile robot configuration is shown in Fig.3.

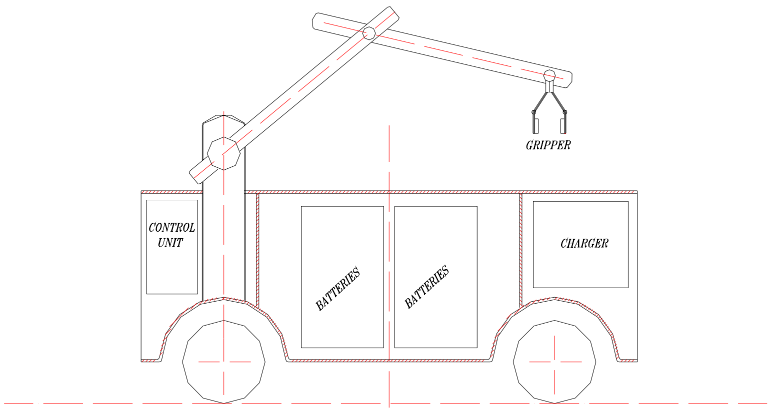

Fig.3 Mobile Robot Configuration

A. Robot Positioning

Methods for robot positioning can be roughly categorized into two groups: relative and absolute position measurements. Because of the lack of a single good method, developers of mobile robots usually combine two methods, one from each category.

In this work, the relative method used is odometry. This method uses encoders to measure wheel rotation and/or steering orientation. Odometry is totally self-contained and it is always capable of providing the vehicle with an estimate of its position, but the position error grows without bound unless an independent reference is used periodically to reduce the error. Natural landmark recognition was used as an absolute positioning measurement system to correct periodically the right position of the robot.

B. Encoder, Decoder and Motion Controller Used

Optical incremental encoders are a mean for capturing speed and travelled distance on a motor. Incremental encoders output square pulses as they rotate. Counting the pulses tells the application how many revolutions, or fractions of, the motor has turned. Rotation velocity can be determined from the time interval between pulses, or by the number of pulses within a given time period. Because they are digital devices, incremental encoders will measure distance and speed with perfect accuracy. Quadrature encoders have dual channels, A and B, which are electrically phased 90o apart. Thus, direction of rotation can be determined by monitoring the phase relationship between the two channels. In addition, with a dual-channel encoder, a four times multiplication of resolution can be achieved by counting the rising and falling edges of each channel (A&B).

In this work, the HEDS5540, 3 channels high performance optical incremental encoders, shown in Fig. 4, are used. This IC consists of multiple sets of photo detectors and the signal processing necessary to produce the digital waveforms. The digital output of channel A is in quadrature with that of channel B (90 degrees out of phase). The encoder standard resolution is 1024 counts per revolution.

The general purpose motion control IC, HCTL-1100, is employed. It frees the host processor for other tasks by performing all the time intensive functions of digital motion control. The HCTL-1100 provides position and velocity control for DC, DC brushless and stepper motors.

It receives its input commands from a host processor and position feedback from an incremental encoder with quadrature output.

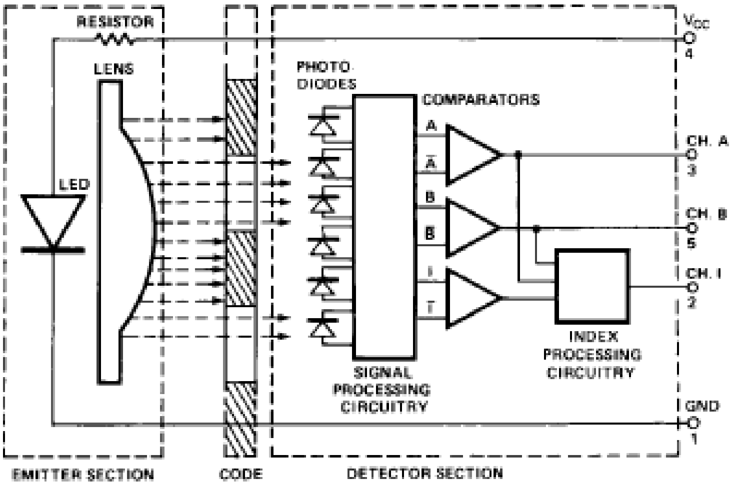

Fig.4 Block Diagram of Encoder

C. ULTRASONIC SENSOR

Sensor model is the mathematical description of the sensory data obtained from the physical sensing units. Ultrasonic sensors are used in this work to build a map of the environment. The map contains information of the boundaries of the environment and obstacles inside which are to be used in the navigation. The ultrasonic sensor provides range information based on the time-of-flight (TOF) principle as given in equation (1),

$$d = v\,t \tag{1}$$

where d is the round-trip distance, v is the speed of propagation of the pulse and t is the elapsed time.

C.1 WHY USE ULTRASONIC SENSORS?

According to [8,14], ultrasonic TOF ranging system is today the most commonly used technique employed on indoor mobile robotics systems, primarily due to the following reasons,

Low cost

Ultrasonic sensors are widely available at very low prices. In this aspect, the ultrasonic sensor has great advantage over the laser scanner and other sensors such as the camera.

Easy maintenance

For practical use, maintenance is an important issue. Ultrasonic sensors are compact in design, light in weight and very reliable. It is also easy to interface these sensors with other subsystems of the robot.

High range detection accuracy

The range detection from the ultrasonic sensor is very accurate.

C.2 SRF04 Ranger

The most commonly used sonar device for mobile robots is the well known Polaroid ultrasonic ranging system.

After studying some types of ultrasonic sensors, we selected the [Devantech SRF04 Compact High Performance Ultrasonic Ranger] for use in this work. The description and specifications of the SRF04 are as follows :

Description: This is a fantastic ultrasonic ranger that has an approximate range of 3cm to 3m. This ranger has a logic line used to trigger a pulse and the echo is returned on a second line. Minimal power requirements and a compact, self contained design make this one of the most popular detectors.

This block anodized aluminum housing can hold one SRF04 range finder. All necessary hardware is included. Fig.5 represents the SRF04 ultrasonic sensor.

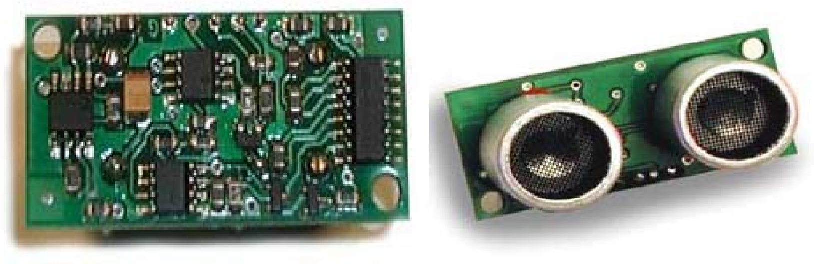

Fig.5 Devantech SRF04 Ranger

C.3 SOFTWARE DEVELOPMENT

ATMEL AVR microcontrollers is easily programmed using the ATMEL integrated development environment AVR studio 4, where assembly programs can be edited, simulated, debugged and downloaded to the microcontroller.

In our case to make our programs easy-made, efficient, portable and readable, we decided to program in C. We've used IAR imbedded workbench for ATMEL which is an integrated development environment (IDE) provided by IAR corporation for C-programming of ATMEL microcontrollers. By IDE is meant the full software development cycle including source code editing, debugging, compiling and linking supported in one user friendly operator interface.

Time Base Generation

It is clear that a microcontroller always depends on some sort of precise clock (or oscillator) for its normal operation. The oscillator used in our application is an 8 MHz crystal. By using one of the internal time/counter modules of the microcontroller itself, this 8 MHz clock can be stepped down to any desired value, the timer/counter module is fully controllable by the software regarding its start/stop. This feature is particularly useful in generating the time base only when desired and switching it off to reduce HF noise when the time measurement task is completed.

C.4 URFS H/W BUILDING BLOCKS

As shown schematically in Fig.6, the main building block beside the Atmel AVR microcontroller is a programmable TTL counter chip 8254 which has 3 independent counters, the gate of each counter is controlled by a dual input OR gate, a counter is enabled as long as its gate is in high logic level.

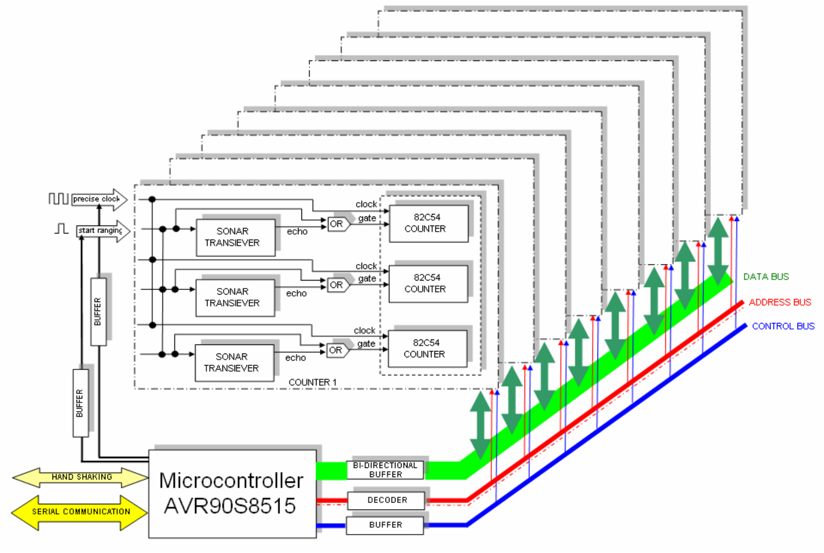

Fig 6. URF board H/W building block

C.4 Ultrasonic Sensors Positioning

The main sensory system used to detect obstacles is the ultrasonic sensors. From the specifications of the SRF04 sensor and to minimize the dead zones during the navigation of the robot, the distribution of the sensors on the robot body was designed as shown in Fig.7.

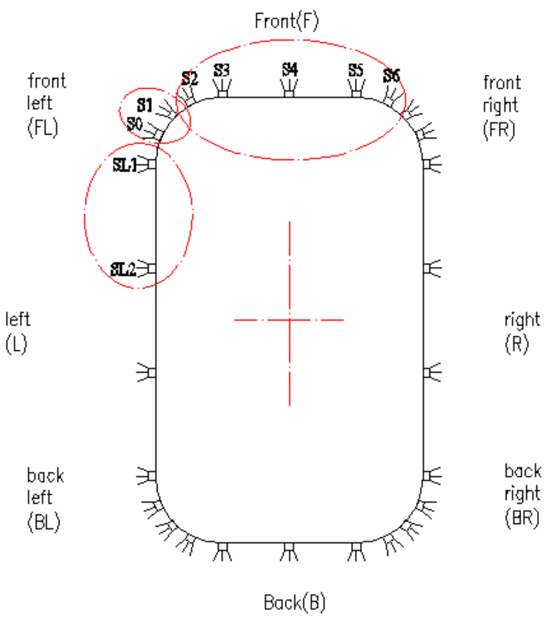

Fig.7 Sensors positioning around the robot

Although dead reckoning navigation is easy to implement, it suffers from the drift problem which is serious in some navigation tasks.

So navigation to be reliable shouldn't depend on one mechanism only. Hybrid navigation system that combines the perception and dead reckoning is better. Information from different sources can be fused to help the robot to take the decision of what is the next step. The position estimate provided by dead reckoning is corrected by matching the perception against a stored world map. Landmark-based navigation depends mainly on the agent's perception to its environment. If the environment contains confusing information or few perceptually distinguishable landmarks, the performance of these systems decline. The perceptual aliasing problem can be solved by including the odometry data to discriminate between the similar places.

5 PROPOSED ROBOT NAVIGATION

A. AUTONOMOUS CONTROLLER APPROACHES

There are two main control approaches used in designing robot controller which are:

Traditional approach

Behavior-based approach.

A.1 Traditional Approach

Traditional approach [15,16] structuring a robot's control into functional modules: perception, planning, learning etc., and constraining as much as possible the environment where the robot will operate. Creating a model of the environment and the robot preprocess sensor information into abstracted internal representations that are acted on by a central planner, then instantiated the results to become actions that can be executed by the robot to reach a specific goal. This can be represented in horizontal control architecture as shown in Fig.8.

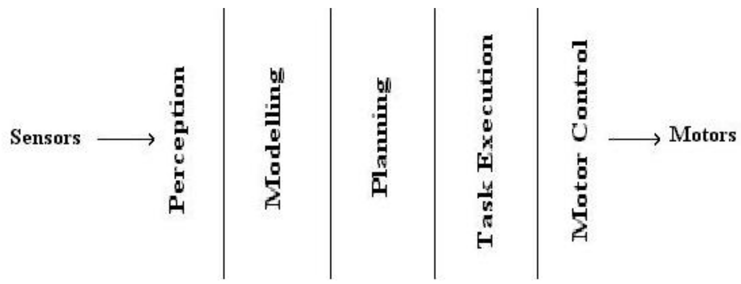

Fig.8 Horizontal control architecture of the robot

Control systems using this architecture solve their task in several steps. First, the sensor's input is used to modify the internal representation of the environment. Second, based on the internal representation planning is made. This results in a series of actions for the robot to take to reach a specified goal. Third, this series of actions is used to control the motors of the robot. This completes the cycle of the control system and it is restarted to achieve new goals.

General planning approach has several problems. Maintaining the model is in many cases difficult because of sensor limitation or imperfection. The plans produced by the planner often don't give the effects in the real world that is anticipated.

A.2 Behavior-Based Approach

In Behavior-based approach, instead of decomposing the task based on the functionality, the decomposition is done based on task-achieving modules, are called behaviors on top of each other as shown in Fig.9, this is called vertical control architecture.

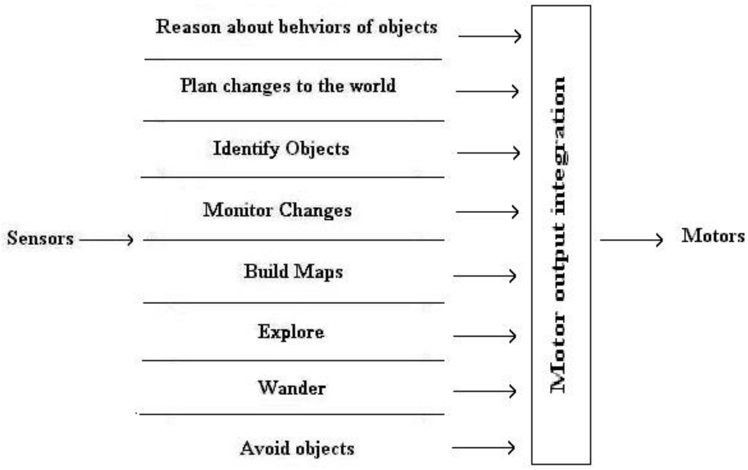

Fig.9 Vertical control architecture of the robot

Each behavior calculates a mapping from sensor inputs - the sensor inputs relevant for the task of that behavior are used - to motor outputs. The suggested motor outputs from the behavior with highest priority are used to control the robot's motors, or summed to generate one motors' output. These architectures are called behavior-based control approaches and represent methodologies for endowing robots with a collection of intelligent behaviors. Behavior-based approaches are an extension of reactive architecture, their computation is not limited to lookup table and executing simple functional mappings

Behavior-based systems are typically designed so that the effects of the behaviors interact in the environment rather than internally through the system. We used this controller architecture in designing our controller.

B. ROBOT'S MOTION TRAJECTORY

The vehicle will move inside and outside the warehouse as shown in the following block diagram Fig.10.

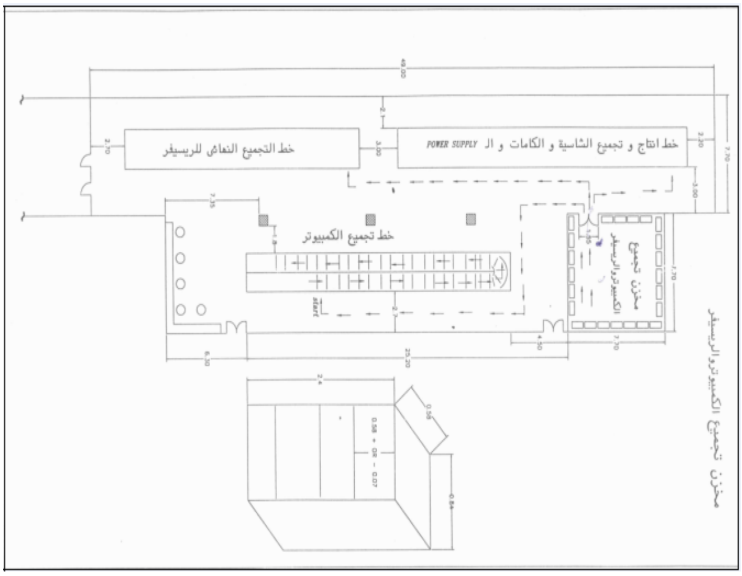

Fig.10 Motion Trajectory and Assembly Lines

Navigation of the vehicle is divided into three main parts:

1. Inside warehouse
2. Outside warehouse

3. Maneuvering at the warehouse door.

Robot can figure out from the odometer system if it is inside or outside warehouse. There are several rule bases according to sensors readings and odometer that make the robot switch from one controller to the other.

C. ROBOT'S MOTION TYPES

The robot vehicle is designed to perform only two distinct kinds of motion in the warehouse:

Straight-line motion, where both motors are running at the same speed and in the same direction,

Rotation about the vehicle's center-point, where both motors are running at the same speed but in opposite directions.

This approach is advantageous for several reasons:

1. Wheel slippage is minimized because of the simultaneous action or rest of both wheels and because of the "on-the-spot" rotation action for turns.

2. A relatively simple control system may be used, since in either case the only task of the controller is to maintain equal angular velocities,

3. The vehicle path is always predictable, unlike other motion strategies which smooth sharp corners by an unpredictably curved path. A predictable path is advantageous when global path planning, to avoid obstacles, is employed.

4. The vehicle always travels through the shortest possible distance (straight-line or "on-the spot" rotation.

D. THE PROPOSED CONTROLLER

At designing a motion controller for an autonomous mobile robot there is a main problem that must be handled which is the obstacle avoidance problem.

The robot senses the obstacles using its sonar sensors.

D.1 Controller inside the Warehouse

In this location of the warehouse, it is required from the robot to go to certain shelf to pick up some components and then move towards the door as a first step to drive these components to the required assembly lines. So, this task can be interpreted as, it is required from the robot to move from initial (x,y) position to certain target (x,y) position avoiding collision with any obstacles in his way, like human, furniture ..etc the robot can perform this behavior by one of two strategies:

1. The angle between the robot's head and the line connecting between the center of the robot and the target point is calculated as shown in Fig.11 then the robot rotates as shown in equation (2). Assume the vehicle has to travel from a known present location (xo , yo, θo) to a new location (xf , yf , θf), the following procedure is performed to determine a trajectory. First, the distance L and the slope θ of the straight line connecting the present and final locations are calculated:

$$\phi = \arctan \frac{y_{f} - y_{o}}{x_{f} - x_{o}} \tag{2}$$

$$L = \sqrt{(x_{f} - x_{o}) - (y_{f} - y_{o})} \tag{3}$$

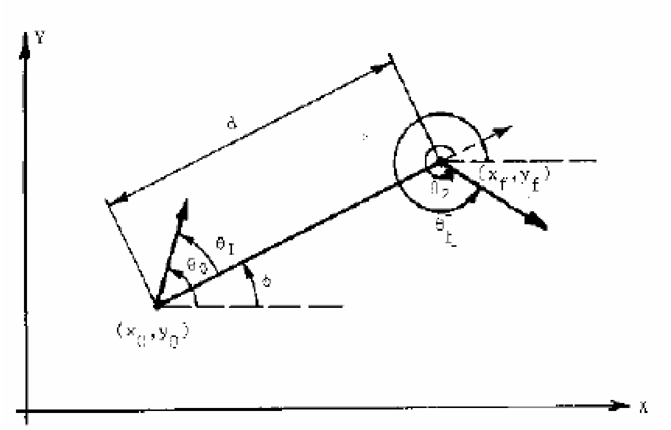

Fig.11 Vehicle traveling from initial position to a final position.

2. As the mission of the robot in the inner warehouse is to pick up components, so the target point will be always next to one of the walls. So we suggest the robot will move parallel to the walls until arriving to the target point to pick up the required components.

In our implementation we used the second strategy. It is more convenient than the first one as the robot may need to pickup components from different faraway shelves.

The readings of the grouped sensors are assigned one of two labels which are: far and dangerous. Each group has different interpretation to the meaning of far and dangerous. For example for the front sensors group the reading of the group ranging from 1m to 3m is far while the reading from 3cm to 1m is dangerous. The expert system rule base is presented in Table I.

The difference between the readings of the two sides sensors are used to help the robot to align to the left wall. We want the robot to keep certain distance to the wall which is in our work 50 cm. the selection of this value to help the robot to turn smoothly at the corners without getting stuck with the walls. While the other grouped sensors allow the robot to detect the obstacles in his way and avoid by rotating around.

Table I. Expert System's Rule Base

#	Δ L	FR	FR-LT	Left Wheel	Right Wheel
1	0	Far	x	Forward	Forward
2	+ve	Far	x	Back	Forward
3	-ve	Far	x	Forward	Backward
4	X	Dang	x	Forward	Backward
5	X	Dang	dang	Forward	Backward

D.2 Outside the warehouse

Over the outer-warehouse, the robot will move in one of three predetermined paths to reach one of the three assembly lines. These paths are saved in the robot. Each path is saved as a consequent number of (x, y) points as shown in Fig.10. In order that the robot correctly follow a certain path, it has to trace its points in sequence

6 CONCLUSIONS

In this paper, the navigation of a mobile robot in a warehouse was presented .The robot has to navigate inside the warehouse to pick the components needed for the industrial operation. Over the outer path, the robot will move in one of three predetermined paths to reach one of the three assembly lines in operation. These paths were saved in the robot memory. The sensor system used is composed of wheel encoders and ultrasonic sensors to correct the position of the robot resulting from odometry system and to avoid any obstacles during navigation.

The navigation system was described and the sensory system description, positioning and operation was presented.The control of the navigation system was also detailed.

REFERENCES

[1] M. Kam, X. Zhu, P. Kata, "Sensor Fusion for Mobile Robot Navigation", Proceedings of the IEEE, Vol. 85, No.1, January 1997, pp 108 -119.

[2] I. Ashokaraj, A. Tsousdos, P. Silsonand B. White, "Sensor Based Robot Localization and Navigation: Using Interval Analysis and Extended Kalman Filter" Proc of the MISC´99 Workshop on Applications of Internal Analysis to systems And control, Criona, Spain , 1999, pp158-165.

[3] F. Chenavier and J. Crowley, " Position Estimation for a Mobile Robot using Vision and Odometry ", Proc. of IEEE Inter. Conf. on Robotics and Automation , ICRA´92, Nice, France, May 12-14, pp2588-2593.

[4] J.M.Evans, " HelpMate: An Autonomous Mobile Robot Courier For Hospitals " Proc. of the Int. Conference on Intelligent Robots and Systems, IROS´94, Munich, Germany, Sept 12-16,pp1695-1700.

[5] J. Borenstein and L. Feng, "Measurement and Correction of Systematic Odometry Errors in Mobile Robots", IEEE Trans on Robotics and Automation, Vol 12 , No. 6, December 1996, pp 869-880.

[6] A. Korodi, T. L. Dragomir, "Correcting Odometry Errors for Mobile Robots Using Image Processing", Proc. Of the International Conference of Engineers and Computer Scientists 2010, Vol. II, IMECS 2010, March17-19, 2010, Hong Kong.

[7] A. Elfes, "Sonar–based Real- world Mapping and Navigation" IEEE Journal of Robotics and Automation, Vol. RA-3, IV'3, 1987, pp249-265.

[8] J. J. Leonard and H. F. Durrant – Whyte, "Directed Sonar Sensing for Mobile Robot Navigation" Kluwer Academic Publishers, 1992.

[9] S. Borthwick and H. Durrant–Whyte, "Dynamic Localization of Autonomous Guided Vehicle", Proc. 1994 IEEE Int. Conf on Multi sensor Fusion, Las Vegas , NY, 1994, pp92-97.

[10] Z. Yi, "Multi-Ultrasonic Sensor Fusion for Mobile Robots in Confined Spaces", M.Sc thesis, School of Electrical & Electronic Engineering, Nanyang Technological University, 2001.

[11] M. Drumheller, "Mobile Robot Localization Using Sonar", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-9, No.2, pp.325-332, 1987.

[12] J. H. Lim and D. W. Cho, "Specular Reflection Probability in the Certainty Grid Representation", Transactions of the ASME, Vol.116, pp.512-520, 1994.

[13] R. Kuc and B. Barshan, "A Spatial Sampling Criterion for Sonar Obstacle Detection", IEEE Transaction on Pattern Analysis and Machine Intelligence, PAMI-12(7), pp.686-690, 1989.

[14] U. Raschke and J. Borenstein, "A Comparison of Grid Type Map-building Techniques by Index of Performance", Proc. of IEEE International Conference on Robotics and Automation, Vol.3, pp.1828-1832, 1990.

[15] R Arkin, "Behavior-Based robots", MIT Press, 1998.
[16] S. Russell, P. Norvig "Artificial Intelligence: A Modern Approach", Prentice-Hall International, Inc., 1995.