Development of Autonomous Explorer Mobile Robot for a Specified Environment

Csongor Márk Horváth and Róbert Tamás Fekete PhD
Budapest University of Technology and Economics/Department of Mechatronics, Optics and Engineering Informatics, Budapest, Hungary
hcsongorm@gmail.com, frt@mogi.bme.hu

***Abstract* — The main contribution of this paper is the online user control interface on eternal network which could serve more users parallelly and the control algorithm of the robot, which is working autonomously in a specified area like a corridor of a building floor. On the website the user could monitoring the movement, the position and the status (busy, idle) of the rover and give target coordinates, where it to go should. The main purpose of research was to create an education mobile platform to involve students with different interest and make possible to develop and try out their own algorithm contained the new learnt knowledge. Another application field of the rover is to use as a runner. It could make the work of organization easier and more comfortable. In this case the employees should not leave there workplaces to deliver to others the small items like hard copies, office accessories and other objects up to the highest carry capacity, which is currently around five kilograms. We introduce the mechanical parts and electrical components of robot, the structure and working methods of the Artificial Intelligence and how would work and the visual navigation.**

I. Introduction

The main motivation to start a new project from the very basics was the huge interest for mobile robots and space activities, to get familiar with the building up an autonomous robot, to create a multifunctional rover platform, to develop mapping and tracking systems

The development of a robot is challenging, because e.g. a robot can be autonomous, semi-autonomous or remotely controlled. Different fields need various degree of autonomy; the highest requirements are in space exploration, cleaning of the floors, mowing lawns, and treatment of waste water [1], [2].

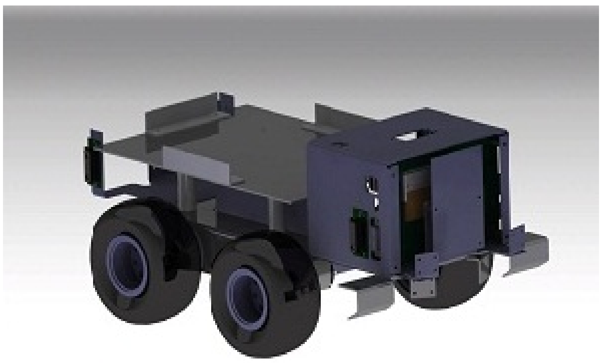

Figure 1. The early 3D CAD model of rover

The rover is working on the floor corridor of one building of University. The main task of the construction is to fulfill an order to go from one target point to another. The order gets on a user interface which is reachable on the internal network. The targets could be a room door or any other specified point. Points stored in a database and connected into a graph. Other points are also there which could be corners, ends and starts of lines next to the wall. On this graph the robot can plan individually an optimized route between the targets. During the way the robot able to avoid impacts, redesign the planned route, tracking the route without the interaction of humans.

The robot needs the constant eternal connection as wireless communication for the correct final target definition by the user. This Wi-Fi system is usually installed and covers the full working area both of the humans and the robot.

II. Parts of the Robot

A. Mechanics

The robot contains a massive and simple chassis to carry the electrical parts and the useful cargo. The base was made from 2 mm thick fine steel plate. That contains the suspensions of wheels, the recessed holder of the accumulator and some sensor fixation point. The wheel suspension has formed from separated plates to make possible to use vibration damper equipments.

Connected to the base plate is the superstructure, which was made from 1 mm thick aluminum plate for the easy handling because it contains some complex

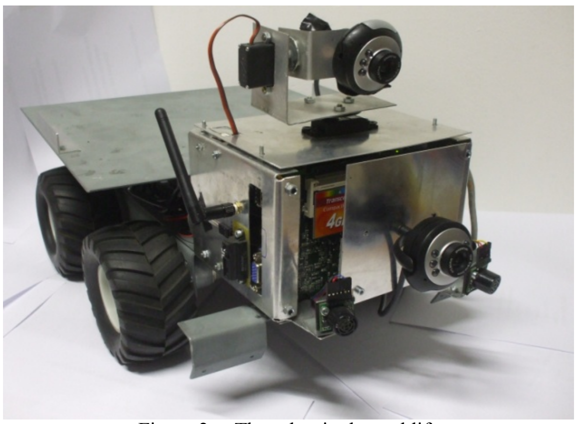

Figure 2. The robot in the real life

978-1-4577-0045-3/11/$26.00 ©2011 IEEE

structures. In the frontier part of the base plate is the box of logical boards with CPU panel and embedded microcontroller board. Sensor holders and the cameras joints are made of the same aluminum as the superstructure. The chassis has one more part, the container of the useful cargo. This has approximately the size of a copy paper and has edges to protect carried items to fall down during operation.

B. Driving system

The driving system contains 4 individually motorized wheels with quadrature motor encoder connected to the external shaft of the motor and a power electronically suiting. With individual motorization we saved the transmission between a bigger motor and the drive shafts, meanwhile giving to the robot bigger moment and maneuverability. The DC motors have premade transmission with 1:30 gear ratio and 200 revolutions per minute on output shaft, manufactured by Lynxmotion [3]. The encoders have a 100 unit disk resolution per revolution and give out two different digital signals with time offset [3].

The requisite motor voltage produced by a Sabertooth product. The 2x10 type motor driver has a microcontroller and a switch pad to apply different settings. In our case, it controlled by pulse width modulated signals for the two sides and give out +/- supply power (12DCV) up to 10 A, however only 3-4 A are necessary.

Sabertooth is the first synchronous regenerative motor driver in its class. The regenerative topology means that the batteries get recharged whenever the robot is commanded to slow down or reverse. Sabertooth also allows making very fast stops and reverses - giving to the robot a quick and nimble edge [4].

The electrical power supplied by a 12DCV and 7, 2 Ah lead accumulator. All boards have its own power connector, which contains voltage regulation and protection against too high current. In the future it planned to be replaced the lead accumulator to Lithium-ion batteries to save weight and expand effective explorer range; furthermore build up a docking base to support automatically battery recharge.

C. Sensors

The robot has sensors on its sides and front. Two Sharp GP2D12 and two Sharp GP2Y0A710YK infrared light sensors are placed to the left and right side to make able to ride parallel the wall and calculate the angle of rotation. The GP2D12 type sensor has 10 – 80 cm range, but between 60 and 80 cm comes a significant level of noise and over 80 cm the noise becomes too high, makes a very various signal, but always stay below a specified value [5]. GP2Y0A710YK sensor has the same characteristics, but in range of its own. It is calibrated between 100 and 550 cm, but become noise over 500 cm [6].

There is 20 cm lack of measurement, which could makes problem, if we seeing the sensor's data individually. But combined with each other an increased measurement distance is obtained by use the rising edge of GP2Y0A710YK sensor's characteristics as seen on the figure 3 [7].

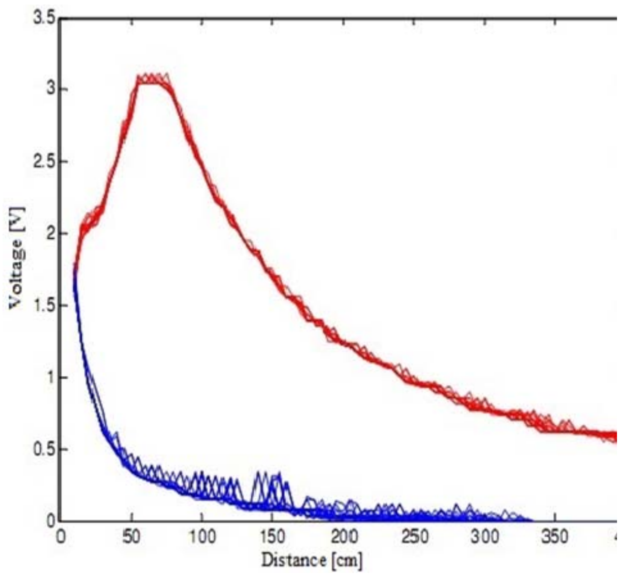

Figure 3. Sharp GDP2D12 (blue) and GP2Y0A710 (red) IR sensors output voltage characteristics dependence of distance

Two LV - MaxSonar - EZ0 placed in front of the robot. They are taking care of avoiding harmful impact by stopping the robot if anything gets too close to the front. These sonar sensors send out ultrasonic sound waves and calculating the distance of the subject from the difference between transmitting and receiving time [8].

Figure 4. The box of logical boards. In the center is the MCU, at the bottom is the CPU and the right side is the Motor driver

D. Function of the Microcontroller

For processing the sensors data and controlling motors an ATMEL ATmega128 microcontroller (MCU) was used. The MCU's Analog to Digital Converter (ADC) is getting the raw measurement results of the sensors, filtering and store for later handling. The data are sent through UART[1] to the board computer, connected with USB.

[1] Universal Asynchronous receiver/transmitter

The UART communication structured to commands like a protocol form. The commands separated into two parts, the header and the packet. The header is the first byte, which contains the length of the packet and the specification of the command. The packet length is changeable between one and four byte.

The command could be data request, set variables (references or program mode and constants) or turn on or off a function. The transmitting commands have the same structure with only one difference – the header contains the type of sent data.

The biggest part of communication is the data request. The computer program collects all of the measured data from the microcontroller like the followings: sensors' results, counted revolution, reference values of speed and other program set variables and function's constants.

E. Motor regulation

The motor regulation is also done by the MCU. The microcontroller has a reference revolution value and sending out 20 ms long PFC PWM[2] pulses with declared length between 1000 and 2000 μs to the Sabertooth motor driver. Here it means 1000 μs the full forward, 1500 μs the stop and 2000 μs the full backward as at radio controlled models.

The motor driver transforms the PWM signals to proportional voltages which make the motors working. On the motor external shafts are the encoders to give a certain value of revolution. Knowing the measured and the reference value, the microcontroller is able to change the width of pulse.

III. PROGRAMS

A. Artificial Intelligence

The system board computer is an ALIX 3D3 model with 500MHz processor, 4GB compact flash memory card and a Wi-Fi card [9]. A Linux kernel is used to support high level programming languages.

Our Algorithm collects the data of environment from the different sensors (through the microcontroller) and from the frontier camera, which can give distance values and objection realization with perspective bijection [10]. The robot navigation solved with low level detection, but combined with visual information processing could be more powerful.

The robot can be controlled remotely, or can be assigned to go to a location, known by its inner map. In both cases, the user and the rover communicate through the local Wi-Fi network, using the same java programs.

When the robot controlled manually, the user has to choose this function on the application window and then use the direction button on screen, or on the keyboard. In this mode, the signal processing is minimized to the level of collision detection.

[2] Phase and Frequency correct Pulse Width Modulation

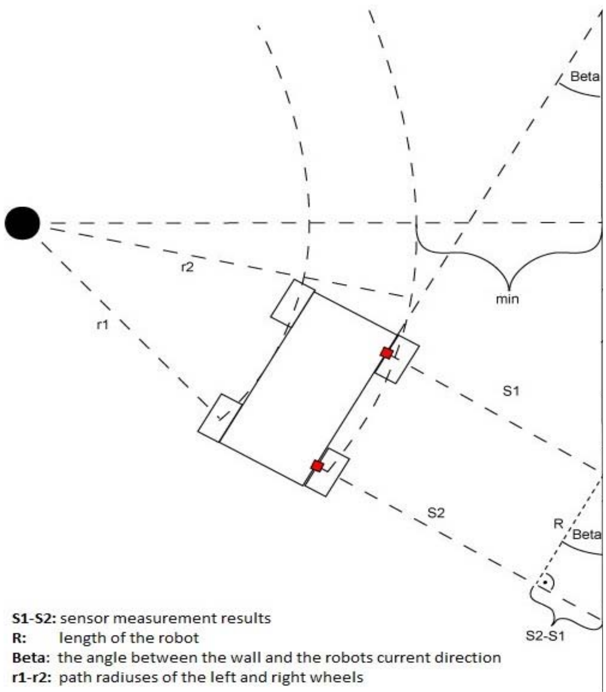

Figure 5. Minimum distance from the wall during parallel correction

If the robot sees an object too close in front of itself, the AI stops the movement, and ignores any orders, commanding it to go closer to that object.

When the user just wants the robot to go to the chosen location, regardless of the path of the movement, it can be done by selecting an end location from the robot's map. This map is the visualization of the graph that allows the construction to find the optimal rout by applying the A-star algorithm. This means that the robot has to stand in a known starting point, with a known orientation. In real life the path, given by the A-star algorithm, may be impassable because of the obstacles in the way. Obstacles are moving (e.g. humans) or static ones (e.g. furniture). The rover is not able to tell the difference, so when it has to stop; first it waits a few seconds for the obstacle to move away. If it does not happen it bypasses it.

Normally the robot should move in the middle of the corridor, so sometimes its orientation or its distance from the walls needs to be corrected (after bypassing an object, or because of the roughness of the floor). The robot tries to correct these values while moving, but to do so; it needs to know the minimal wall distance of the new path (Figure 5).

The minimal distance (S_{min}) calculated the equation (1) and shown on Figure 5, where β is the angle between the wall and the robot current direction, S_1 and S_2 distances measured by IR sensors of the side, r_1 and r_2 come from the actual speed values.

$$S_{min} = \left(\cos\beta \, \frac{(S_1 + S_2) + (r_1 + r_2) + robotlength}{2} \right) - r2 \qquad (1)$$

The algorithm starts when the value of S_2 become too low and changes only the left side speed, which effect on r_1 value. The comparison between the calculated S_{min} and

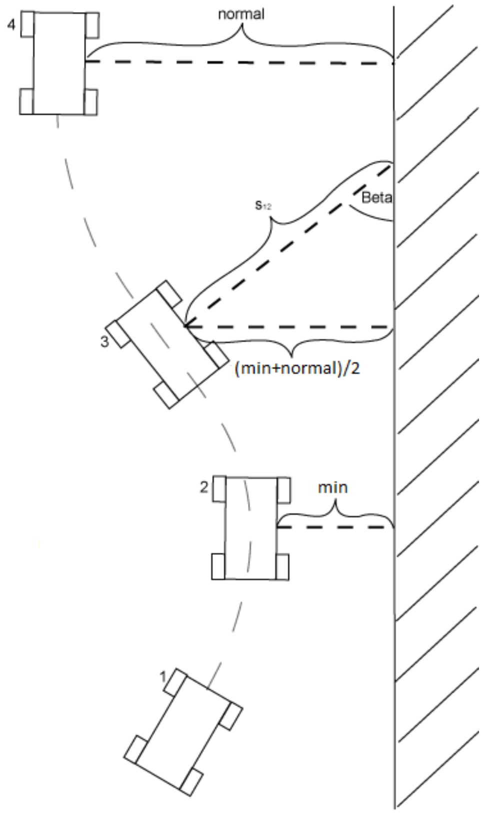

Figure 6. Parallel correction

the safe minimum distance determines the needed left speed value. If this is smaller than the 30 % of the actual speed, the robot has to stop and change orientation on the spot. A probable correction during moving showed on Figure 6.

B. *Picture processing*

The processing of camera's picture is supported by four line laser and based on 2D-2D perspective bijection. These lasers are projecting a square to the floor, where the dimensions of corners are known. The coordinate axes are the current direction of the movement and the perpendicular. The origin is geometric center of the robot.

If we know the corners of the square and we know the corners in the picture (Figure 7) parameters of bijection are can be defined solving the equation system of parameters (2) of projection [10], [11].

$$v_x^i = \frac{p_0 \cdot t_x^i + p_1 \cdot t_y^i + p_2}{p_6 \cdot t_x^i + p_7 \cdot t_y^i + 1},\ v_y^i = \frac{p_3 \cdot t_x^i + p_4 \cdot t_y^i + p_5}{p_6 \cdot t_x^i + p_7 \cdot t_y^i + 1}, i = 0,1,2,3 \quad (2)$$

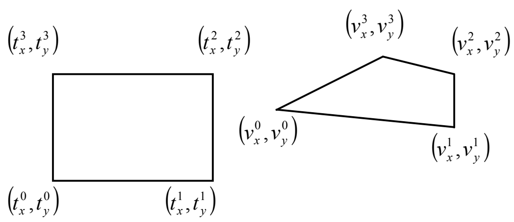

Figure 7. The object (t) and the perspective (v) image

The other principle of the picture processing is the fact that it will be used indoors. The corridor is covered by 15x15cm big tiles. It makes a recognizable pattern on the picture of the camera that easily detectable. On this grid the program can find representative points which would show the movement of the robot by the changed position between pictures. To filter unnecessary parts, the Prewitt operator is used to make noise reduction and edge detection.

This is very difficult to realize the same points on different pictures. Combine the distance measurement with the quadrature encoder's data; we could narrow the area of potential point's location. After the same point founded we could refresh the robot position with a sufficiently accurate distance measurement.

The objects are also detectable on the pictures. As the robot movement known, the size and distance of the objects can be estimated.

IV. Future Plans

The previously detailed picture processing is still in progress. We would like to find a more sophisticated visual control system which could process the picture in real time. For that a better hardware needed or a supporting computation base.

The chassis would be improved to overcome the obstacles higher than 5 cm. This limit made by the actual inflexible wheel suspension and equal to a doorstep high. This new chassis would contain one more axis and flexible suspension, power consumption monitoring system and a docking base to recharge the rover's batteries.

To reach the final goal of research the rover should be able to work outdoor. For that we need a global mapping and tracking, like GPS is necessary to get target coordinates or a map to get base points. The cameras on the board would help to explore the local environment and objects. If our attempt is successful, the rover will be able to work places where humans are in danger, like very tight and long places, contaminated or earthquake-stricken areas.

Figure 8. A view to the floor what should be processed

ACKNOWLEDGMENT

The authors would like to thank NDA (National Development Agency) for the support since this study has been carried out as part of the project TECH_08-A1/2-2008-0121 (NKTH).

REFERENCES

[1] http://en.wikipedia.org/wiki/Robot

[2] http://en.wikipedia.org/wiki/Autonomous_robot

[3] http://www.lynxmotion.com/c-11-spur-gear-motors.aspx

[4] http://www.dimensionengineering.com/datasheets/sabertooth2x10.pdf (2007)

[5] Sharp, GP2D12/GP2D15 General Purpose Type Distance Measuring Sensor Datasheet (2005)

[6] Sharp, GP2Y0A710YK Distance Measuring Sensor Datasheet (2005)

[7] http://www.maxbotix.com/documents/MB1000_Datasheet.pdf (2005)

[8] M. Kam, X. Zhu, P. Kalata, Sensor Fusion for Mobile Robot Navigation. Proceedings of the IEEE Vol. 85, Issue 1, pp. 108-119 (1997)

[9] http://pcengines.ch/alix3d3.htm

[10] D. Ujevic, L. Szirovicza M. Halász, P. Tamás, S. Petrak, K. Dolezal, Z. Domjanic, B. B. Sajatovic, M. Kisfaludy, L. M. Vas, A. Koleszár, O. Nagy Szabó, Comparison of Conventional and Computerised Human Body Measurement Methods,
Magic World of Textile 5th International Textile Clothing and Design Conference. Dubrovnik, Croatia, October 3-6, 2010, pp. 523-528, Paper 201

[11] D. K. Kim, B. T. Jang, C. J. Hwang: A Palanar Perspective Image Matching using Point Correspondesand Rectangle-to-Quadrilateral Mapping Fifth IEE Southwest Symposium on Image Analysis and Interpretation 2002