Design of an Autonomous Robotic Vehicle for Area Mapping and Remote Monitoring

M. Papoutsidakis
Dept. of Automation Engineering
PUAS, Athens, Greece

K. Kalovrektis
Dept. of Informatics
University of Thessaly
Lamia, Greece

C. Drosos
Dept. of Automation Engineering
PUAS, Athens, Greece

G. Stamoulis
Dept. of Informatics
University of Thessaly
Lamia, Greece

ABSTRACT
The increase of human needs has pushed the robotics sector to evolve even more. One area that begins to bloom in robotic systems are semi-autonomous or autonomous monitoring and guarding vehicles from a distance. The present platform is an example of such a system. It will be the analysis of its design, construction and operation. The operator driving the vehicle will have video feed and additional information about the space monitor. This system could be the basis of a complete automated platform with more sensors that will cover more needs.

Keywords
Robotic vehicles, camera, surveillance, remote controlled robot

1. INTRODUCTION
The technology in recent decades developed at a rapid pace, this results to have innovative solutions that facilitate and make human life safer. In particular, the robotics industry has a long term goal of minimizing the manual work carried out every day by people and improving any task that requires human skills such as accuracy, speed and power.

Primarily the robotics application fields which caused the biggest development are in the department of robotic arms and small robotic vehicles, which are automated, moving autonomously and transferring information via the Internet or a local network to the central control station. Only in recent years, the field of robotic automatic vehicles has seen the rapid development in the areas of the remote control and monitoring space.

The robotic vehicle sector includes user mechatronics, artificial intelligence and multi-agent (multi-agent Systems systems) to help the operation of vehicles. These are the characteristics that make the vehicles are eligible to be designated as intelligent or smart. A vehicle that uses automation for difficult processes, in particular for navigating, can be described as semi-automatic.

While those vehicles based solely on automation systems are referred to as robotic or autonomous. After the invention of the integrated circuit, the complexity of the technology of automation systems has been increased. Manufacturers and researchers then added a variety of automated functions mainly in cars and other vehicles, see [1].

In general, a car is characterized as autonomous when it is capable of recognizing the environment and moving within it, without human intervention. The autonomous vehicles detect their environment by using technologies such as radar, LIDAR (LIght Detection And Ranging), GPS, Odometry and Computer Vision. Advanced control systems read and process information from the measurements of the sensors to identify suitable navigation paths, and obstacles and the relevant markings. Autonomous cars have control systems that are able to analyze the sensor data to distinguish between different cars on the road, which is very useful for planning a route to the desired destination.

Under the term 'autonomously', the vehicle will also have the possibility of self-government (self-governance). Many historical research projects related to vehicle autonomy are actually automated (they are made to be automatic) rely heavily on artificial elements of their environment, such as magnetic tapes. The autonomous control requires good performance under significant uncertainties of the environment for long periods and the ability to offset the potential damage of the system without external intervention. As can be seen from many projects reported frequently proposed to extend the possibilities of the autonomous car through the implementation of communication networks with both directly adjacent cars (Collision Avoidance) and long (for managing congestion). Adding two more outside influences in the decision-making process, some now consider the behavior or the potential of the car as autonomous, see [2].

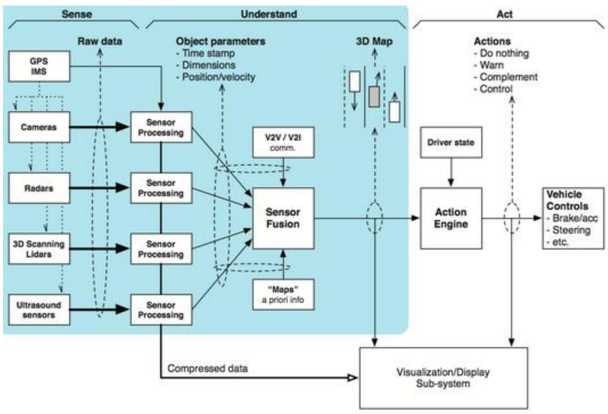

Fig 1: A typical autonomous vehicle control block diagram

The technologies used in such autonomous vehicles are as individual in a more mature stage, but due to the complex interaction between them, there is still a lot of development to be done in engineering at the field of connecting all subsystems that integrated end system can operate autonomously in complex environments, see [3].

2. VEHICLE DESCRIPTION
This paper will present the study and the construction of a small vehicle which will have sensors and controlled from a central control station and will also sent video feedback, so the operator can monitor the space. A representative picture of it can be found in figure 2.

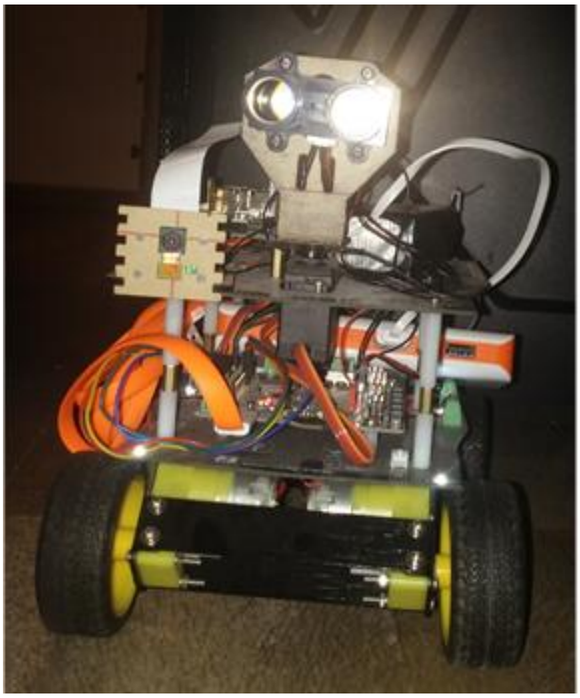

Fig 2: The robotic vehicle of the project

To achieve this objective, the following materials were used in its manufacture:

• Body of a small vehicle

• Wheels

• Engines/motors

• wheel / motor Encoders

• Sensor internal situation

o Gyroscope

o Accelerometers

o magnetometers

• Camera

• LiDAR (Light Detection And Ranging)

• Microcontroller (Romeo)

• Micro (Raspberry Pi 3)

Apart from the construction was implementing three software that will control these parts. The software is divided into three pieces and their implementation were used three different development environments:

• Software implementation operating scenarios –

Arduino IDE (C)

• Communication Software - MonoDevelop (C #)

• Central control software - Visual Studio 2015 (C #)

2.1 Sensors

The sensors are devices that convert some events or physical quantities into electrical signal in order to make the measurement. The sensory organs (robotic sensors) give the robot the ability to identify and detects that it can be more like human behavior. These instruments give the robot the ability to see, to touch and to hear, see [4]. As mentioned above the sensors presented in an autonomous vehicle may be divided into sensing internal state and external state - environment.

Inertial Sensor Units measures values of internal system variables (robot), as the engine speed, the position of the wheel, the speed of either the battery level, see [5]. This type of sensors are wheels or motors encoders (wheel encoders) and for the vehicle's direction gyroscopes, accelerometers (accelerometer), the magnetometers or a combination of all three called inertial measurement sensor (inertial measurement Unit- IMU).

The encoder is a sensor attached to an object that is rotating, like a wheel or a motor, in order to measure the rotation. By measuring the rotation of the robot, the linear displacement, velocity, acceleration or angle of the rotating sensor can be calculated, see [6].

Fig 3: A set of optical encoders used for the task

The mechanical gyro is a rotating wheel or disc which rotates about an axis having a free orientation. When rotated, the orientation of the axis is unaffected by tilt, or rotation of the support base in accordance with the conservation of angular momentum. This feature makes gyros useful in measuring or maintaining orientation. In robotic vehicles they are used in their electronic form or in the form of microchip type Microelectromechanical systems (microchip-packaged MEMS), see [7].

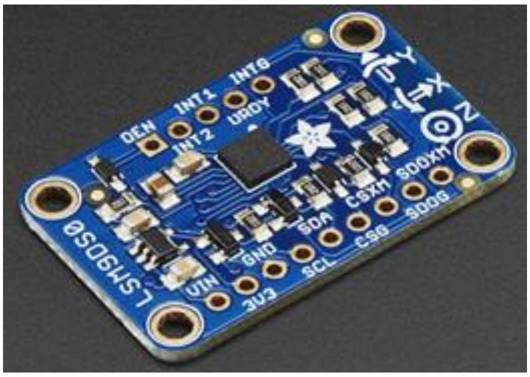

Fig 4: The Inertial Sensor Unit LSM9DS0 with 9DOF

The accelerometer (accelerometer) is a device that measures the gravitational acceleration (g-force). The acceleration of gravity is different from the acceleration, i.e. the change of speed. For example, an accelerometer at rest on the Earth will measure acceleration g = 9.81 m/s2 in the direction straight upward. Conversely, if it is in free fall towards the Earth's core with an average acceleration value 9.81 m/s2 the accelerometer will measure zero, see [8].

The magnetometers are measuring devices mainly for two general purposes, to measure the magnetization of a magnetic material, such as the ferromagnetic, or to measure the strength, and in some cases, the direction of the magnetic field at any point in space. In recent years, magnetometers get smaller to an extent that can be incorporated into integrated circuits with very low cost and find increasing use in compasses in everyday devices like mobile phones, in the form of plate computer (tablet) see [9].

External Sensors deal with the observation of the environmental features on which the robotic vehicle is moving, such as moisture or the colors of an object. The external sensors that operate with the touch of an object called contact sensors, and all the rest, as the sensory systems based on camera, called non-contact sensors, and mimic human eyes, see [10].

The camera is used in robotics as vision and image sensor. An image sensor is a sensor that detects and transmits information of an image. This is accomplished by converting the variable attenuation of the light wave (as they pass through or reflected from objects) into signals, small power bursts carrying information. The waves can be either light or other electromagnetic radiation. The image sensors used in either in analog or in digital type electronic imaging devices such as digital cameras, cameras, medical imaging equipment, night vision equipment such as thermal imaging devices, radar, sonar, and others. As technology evolves rapidly, digital image sensors tend to replace analog, see [11].

LiDAR is a stamping technology that measures the distance of a target by emitting a pulsed laser radiation in the atmosphere. The LiDAR is an acronym of the words LIght Detection and Ranging, detection and light range and was originally created as a combination of the light and radar. This technology is mainly used to create high-resolution maps, with applications in areas such as archeology, geography, geology, geomorphology, seismology, forestry and atmospheric physics. Many times the term LiDAR use to simply refer to the laser scanning or 3D scanning, with land, air and moving platforms.

Fig 5: LiDAR Lite camera

The LiDAR operating principle is quite simple · emit a beam of light on a surface and measure the time it takes to return to its source. When turn a lens on a surface that actually see is the light reflected and returned to the retina. As it is known the light travels very quickly - at about 299.792.458 m / s - so when the light is turned on it appears instantaneously. This is not the case in reality and to quickly measure required instrumentation, something that developments in computer technology has made such measurements possible. Knowing the constant speed of light, the distance traveled can easily be calculated by and from the object:

Distance=((Speed of Light*Time of flight))/2 (1)

The LiDAR pulse generator emits rapid light pulses to a surface, some up to 150,000 pulses per second. The sensor on the instrument measures the time it takes for each pulse to return back, flight time (Time of Flight ToF). Light moves with a known constant speed so the LiDAR instrument can calculate with high accuracy the distance between him and his target. Repeating this rapid sequence, as it rotates, the measuring instrument may be planning a complex map of the environment, see [12].

2.2 Microcontrollers/Processors

The microcontroller used in the present construction is Romeo BLE. The specific microcontroller is based on Arduino UNO, has additional guidebook motor (Motor Driver) and built-in Bluetooth 4.0. It is designed specifically for robotic applications. The Romeo benefit from the open source Arduino platform is supported by thousands of open source projects, and can be easily extended with Arduino Shields. Apart from DC motor can lead stepper motors and servo motors.

The Romeo BLE behaves like Arduino UNO based on ATmega328P chip. It can be programmed via the Arduino IDE environment, something which greatly facilitates the code development process. Also there are plenty of libraries that can be used to develop applications.

Fig 6: Romeo Ble Microcontroller

The objective of this platform is the movement of the space by remote control and monitoring. That is because it was not sufficient to implement only the microcontroller. The aim in this platform is to connect the robotic vehicle to the computer that will be located at the control station via Wi-Fi network. Linking the two will be for the following reasons:

- To receive data, mainly commands from the control station
- To convey instructions to microcontroller
- To send video and data to the control station

Fig 7: Raspberry pi 3 Microcomputer

The microcomputer Raspberry Pi 3 (RPi3) Model B is ideal for performing the above functions. The RPi3 is the third generation of the Raspberry Pi microcomputer.

The RPi3 has new quad 64bit ARM-type processor core clocking 1.2GHz, which can be increased (overclocking). The features and only the processor give very high computational power in order to widen the range of software which can be used and the confidence that the system is stable, no ups and downs on its performance. Also, four of our CPU cores give the ability to have multiple functions simultaneously active. The computer has 1GB RAM, which speeds up data processing. One of the features that stand out from the rest of the class is the built-in Wi-Fi. Also for the connection with other devices and is integrated Bluetooth Low Energy (BLE). Moreover, a CSI port is available for connecting the camera.

3. ALGORITHM AND SOFTWARE

In this chapter, the software parts that were created for this work will be explained. The software is divided into three pieces and their implementation used three different development environments:

Software implementation operating scenarios - Arduino IDE (C)

- Communication Software - MonoDevelop (C #)
- Central control software - Visual Studio 2015 (C #)

Also additional free applications used for sending and receiving the image from the camera:

- The Raspivid which takes the image on the platform and sends it to the network.
- The VLC which receives the data, video, network and displays them to the user.

The software in such platforms should implement all operating scenarios. In this work the main work scenario to be implemented is the transmitter of the robot, in which the user can move the platform forward, back, turn right, left. When the robot starts its forward motion, the microcontroller receives the object distance measured by the Lidar Lite sensor. If this distance is below 10 cm, then the moving platform stops. If an obstacle is at a shorter distance of 10 cm in front, then the forward movement will not be done. Apart from the movements that are send there also we receive data. As mentioned in the previous chapter, our platform is a set of sensors that provides information which are utilized by the software in operating scenarios and informing the user. In particular, the user in one of the auxiliary operating scenarios can move the distance sensor LIDAR Lite to receive distance data barriers. In this scenario, the user can move the servo motor, in which that has been installed with the sensor, from 0o to 90o and 90o to 180o to receive from the microcontroller, distance values for each 1o. In this case a double passage of the engine to get the average of sensor values by fate to reduce measurement error. Also through the encoder, the distance traveled is continuously calculated and in conjunction with the internal state sensor the coordinates of the robot in space can be provided to the user, taking as the point (0,0,0) the starting point.

The loop routine implements all possible scenarios that have been presented. The loop routine runs continuously and according to the command sent perform the appropriate operation. The loop routine is divided into two main parts, the execute motion command and the send odometry measurements

The first part of the program is executed only in the case where it has shipped something to the serial port of the microcontroller, while in contrast the mission of odometry measurements made continuously in the routine and for this reason there is a delay of 100ms to have enough time to read the man, otherwise it would be a continuous sequence of measurements which could not read the human eye. Depending on the character that has sent the corresponding operations:

'a' left turn 10o.

'd' right turn 10o.

's' linear motion toward the back 20cm.

'w' linear motion toward the front 50cm.

'r' sends odometry data, servomotor rotation of 180o and 0o record distances

'e' servomotor rotation from 0º to 90o and record distances

'q' servomotor rotation by 90o to 180o and recording distance

'x' mission odometry data, object distance and move the servomotor in the 90o.

If you receive different character send error message.

Print message 'EOA' (End Of Action) that informs the user that the command is sent successfully.

One of the innovations of this application is a standard control and correction is made in the linear motion of the robot either forward or backward. It is known that the movements performed by the robot may contain errors. For example, if you ask for a vehicle to move 10cm but it can move 11cm. If there is no control and correction even the slightest differences will accumulate and grow bigger, especially if the operation is continuous and for a long time of period. In this application the distance traveled is checked and any error can be found by the following formula:

First find if a balance of all the movements that have taken place that if the platform was asked to make 3 moves forward and two back, but the encoder has measured 100cm then we have:

Desired Distance=3*50-2*20=110

Real Distance=100

remainder=+10

If there is a difference, it means that the previous move is not performed properly, so finding the remainder means finding that the distance traveled was either above or never reached the expected step. Respectively the next step is to subtract or add the rest of the previous movement in our last step. This results each time to correct the mistake of the previous movement. At the end of each movement the corresponding variable that measures the amount of movements separately the front or back is increased. The other commands perform movements on the servomotor that has the distance sensor. These movements are a complete rotation, a rotation from 0º to 90º or 90º to 180º. In each movement calling the corresponding routine that reads values, displays and rotates the motor. In all routines the motor rotates twice in order to have the average price per degree of rotation. If an unacceptable character is send to the Arduino, the user is informed through a message send by the Arduino. The second part of the central loop function stops the vehicle and displays the user the odometry data collected.

As mentioned above, the Raspberry Pi is responsible for the communication interface between the robot and the central control station. Its main function is to transfer asynchronous data from the Arduino to anyone who has a computer connected to the RPi.

The main function of the program performs three basic functions. The first two functions, which are executed only once, they call functions that create the connection to the Arduino and a communication (Socket) start listening to the door 11000 for any IP try to connect. At the end there is a continuous loop that takes the data from the serial port that is connected to the Arduino and sends them straight in or linked users. In the event that any of the three modes display error record on file at a specific location.

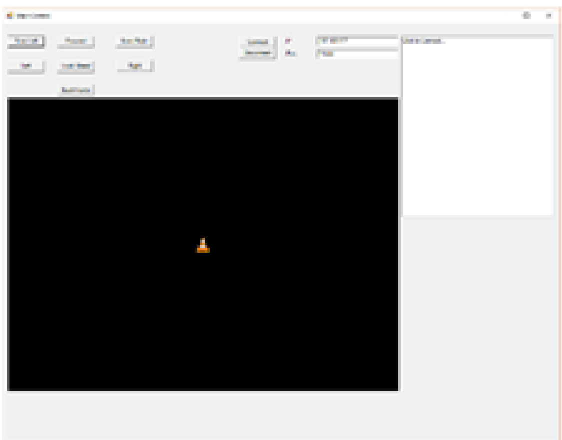

Fig 8: Main Interface (user –vehicle) Panel

This software is based in the control station for controlling the operator the robotic vehicle. The display consists of three main parts. At the top left of the screen there are different buttons, to send commands, and two text boxes that the user enters the IP and the door to make the connection. At the bottom there is the context of live image preview on the Raspberry Pi using VLC. Finally, on the right side is a multiline text box to receive data from the Arduino via the Raspberry Pi. When the user clicks the Connect button initially displays a message to inform the user that the process begins. Initially it is checked if the user has completed the address and the port. If test fails, the user is informed with a note. After the login process starts, initially through VLC and then through Socket with data completed by the user and if the connection is successful then starts the function that expects to receive data. If no error is displayed to the user a message.

After the connection the central computer receives videos via the VLC's streaming functionality and data from the Arduino via Raspberry Pi. Taking them is asynchronous and whenever data is received the supplement and displays them to the user. If an error occurs, it displays different messages that inform the user of the source of the error. In addition to receiving data as possible and send orders to the Arduino when buttons are pressed. In each case, the corresponding character is sent to the other side of communication and there the Raspberry Pi carry him to the Arduino. Finally, when the user clicks on the Disconnect button disconnects the control station with the Raspberry Pi closing the connection with him.

An example of pseudo-code that could implement a scenario semi-automatic:

Routine Traffic Record

Number of movements i

Start

Routine traffic record:

Add a motion made k [i] = [k1, ..., kn]

Add the initial position x [i] = [x1, ..., xn]

, Y [i] = [y1, ..., yn]

, θ [i] = [θ1, ..., θn]

Make the move

Add the final position x [i] = [X1, ..., Xn]

, Y [i] = [Y1, ..., Yn]

, Θ [i] = [Θ1, ..., Θn]

Add a step that eventually became

B [i] = [B1, ..., Bn]

Routine movement repetition:

Starting from the 0 to the i

Start

Make motion K [i]

If there is an error begin avoidance routine

Measures your position P [x, y, θ]

Until you get to the position X [i], Y [i], Z [i]

There is also the case with the full autonomy of the robot using the most sophisticated distance measuring sensors, and encoders internal situation which will increase the accuracy of measurements. In this case the mobile robotic platform can be left in free space to begin the mapping process, and then choose a path that will to be followed, in accordance with the criteria have been defined as optimal. In this work you calculate the position [x, y, θ] of the vehicle in relation to the space and could extend the algorithm to use and other sensors, such as a LIDAR 360 ° to calculate and distances around the vehicle. Also it could be used as a camera sensor. There could be an image analysis software that could find obstacles, corridors and recognize objects. Using the above the platform could create a map of the site, with the help of algorithms such as Kalman filters, and then with a help of an algorithm, such as Dijkstra's algorithm, to calculate the optimal route between the start – finish point.

4. CONCLUSIONS

This project was designed in order to present the design and manufacturing of a remote-controlled robotic platform that will send data and images in one or several central control stations. This process could solve many needs especially in the field of remote monitoring and telemetry in places that are difficult to access or dangerous for humans. The present system is modular and can be extended easily to new functions or new sensors in order to be able to provide more information to the user. This project could be a basis for further research on the field of robotics moving ground vehicles and its construction and creation of software based on the logic of expandability. The code tracks were driven modular construction so that one can use them as separate parts to different project. It could also be semi-automatic or fully autonomous depending on the application needs. A kind of semi-automatic mode would be to learn the robot path to be followed.

5. ACKNOWLEDGMENTS

All authors would like to express their gratitude to the Post-Graduate Program of Studies “Automation of Productions and Services” of PUAS, for the financial support to undertake this research project

6. REFERENCES

[1] Ge, Shuzhi Sam και Frank, Lewis L. Autonomous Mobile Robots: Sensing, Control, Decision Making and Applications. s.l. : CRC Press, 2006.

[2] Dokic, Jadranka, Müller, Beate και Gereon, Meyer. European Roadmap Smart Systems for Automated Driving. smart systems integration.

[3] Finn, Anthony και Scheding, Steve. Developments and Challenges for Autonomous Unmanned Vehicles. s.l. : Springer, 2010

[4] Dahiya, Ravinder S και Valle, Maurizio. Robotic Tactile Sensing. New York: Springer-Verlag, 2013.

[5] Siegwart, Roland και Nourbakhsh, Illah R. Introduction to Autonomous Mobile Robots. London, England : The MIT Press, 2004.

[6] Sensors - Robot encoder (slot,rotary,linear). Society Of Robots. Retrieved from http://www.societyofrobots.com/sensors_encoder.shtml.

[7] Spacetelescope. Gyroscope. [Ηλεκτρονικό] Retrieved from,https://www.spacetelescope.org/about/general/gyroscopes/.

[8] Doscher, James. Accelerometer Design and Applications. Analog Devices. Retrieved from

http://elpuig.xeill.net/Members/vcarceler/articulos/jugando-con-el-wiimote-y-gnu-linux/sensor971.pdf/at_download/file&prev=search.

[9] Budker, Dmitry και Jackson Kimball, Derek F. Optical Magnetometry. Cambridge : Cambridge University Press, 2013.

[10] Dudek, Gregory και Jenkin, Michael. Computational Principles of Mobile Robotics. New York : Cambridge University Press, 2010.

[11] Capturing the image CCD and CMOS sensors. Canon. Retrieved from http://cpn.canon-europe.com/content/education/infobank/capturing_the_image/cd_and_cmos_sensors.do.

[12] How does LiDAR work? Lidar UK.Retrieved from http://www.lidar-uk.com/how-lidar-works/.