Development of Upgradable Mobile Platform for Smart Applications

P.Geekiyanage[1], H.T.Jayarathne[2]
Department of Mechanical Engineering
University of Moratuwa
Moratuwa, Sri Lanka.
[1]pgeekiyanage600@gmail.com
[2]tharangaj13@gmail.com

L.A.D.I.T. Jayasinghe[3], Y.W.R. Amarasinghe[4]
Department of Mechanical Engineering
University of Moratuwa
Moratuwa, Sri Lanka.
[3]hisurujayasinghe@gmail.com
[4]ranamajp@gmail.com

Abstract-**The research related to this paper is focused on developing a low cost autonomous robotic platform for domestic use which can be upgraded to perform multiple tasks. As the first step of it, we present here a testing prototype focusing on navigation and localization. Here the construction of the prototype is discussed where an overview of the platform is given. The driving mechanism and navigation and localization methodology is explained in details with the algorithms used. Finally the paper presents the test results generated through the testing platform including the map generated by the robot.**

Keywords-Domestic robotic platform, Upgradable, Indoor navigation

I. INTRODUCTION

Robotic servants have long been a fantasy that appears in science fiction alone, leaving us only to imagine the experience of using these autonomous and intelligent machines. During the last decade this dream has been transformed into a reality by the introduction of several domestic servant robots as consumer products, most popularly cleaning robots as developed by the iRobot Corporation. However this has never actually reached the truly autonomous nature we expect, a machine which can perform multiple tasks and also mimic human beings in doing so.

In this paper we suggest the next step in mobile robot applications, an upgradable multipurpose robotic platform that has the ability to navigate itself in an indoor environment. This concept is directly connected to the concept of Smart Houses or Home Automation where the suggested product can be employed in a domestic environment.

The robotic platform with external modules to suit the task in hand can be used in a wide variety of application. One of the main areas of applications is *Telecare* services in connection with the smart house concept. *Telecare* is remotely delivering care and support in a virtual way without physically being in the location [1]. In delivering this, autonomous robots are used to perform the tasks that are otherwise done by the caretaker or the nurse in a hospital environment [2]. Safety and security concerns of a house, is another one of the major application areas of this. While the basic platform itself can perform certain safety related duties such as visual spying and fire alarming, external modules can be fitted to perform advanced tasks.

The motivation of this paper is to suggest a suitable basic robotic platform which can cater the above needs of a robotic assistant in a smart house. The platform has self-navigation and localization ability and provides means to support external upgrades. In doing so, we focus our research on developing a low cost product than the ones that are currently available expecting it to be more marketable. This is achievable through fusion of inexpensive sensors instead of using advanced technologies which in return would also demand higher computational power leading in to cost accumulation.

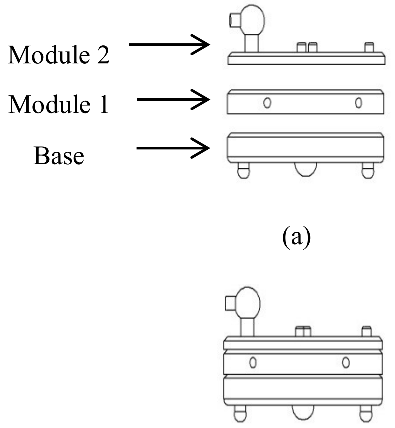
The novelty of the paper lies in the multipurpose nature which is achieved through external modules which can be mounted on to the basic platform as upgrades. These upgrades will allow the domestic user to make the robot do exactly what is needed. A comparable conceptual product for the mobile platform in concern is iRobot AVA Mobile Robotic Platform [3]. However it is clear that this conceptual product does not match the cost criteria of the product that is discussed in this paper and is solely focused on industrial applications.

This paper explains the navigation and localization of the platform and how optimization could be achieved through sensor fusion. The principal focus is on developing on the base of dead reckoning techniques and error reduction through a pre-defined coordinate system. An initial training of the robot is required to identify the map of the expected working area. Once it is trained to the environment it is designed to follow the trained path to navigate itself. This is an extension of the most basic navigation method of line/path following and Kalman filter is used to reduce the error component in the predictions between dead reckoning and inertial navigation [4]. Test results of the concept using the prototype are presented in this paper while the learning will be used in developing the final product.

II. AN OVERVIEW OF THE CONSTRUCTION OF THE PLATFORM

Fig 1 shows the components of the robotic platform.

978-1-4673-5221-5/13/$31.00 ©2013 IEEE



(a)

(b)

Fig 1: a) Exploded view b) Completed platform

In this paper, the main focus is on designing a basic platform that can be upgraded using external modules easily. Fig 2 breaks down the main components of that upgradable platform. The prototype presented in this paper is used to identify the requirements in each of these sections.

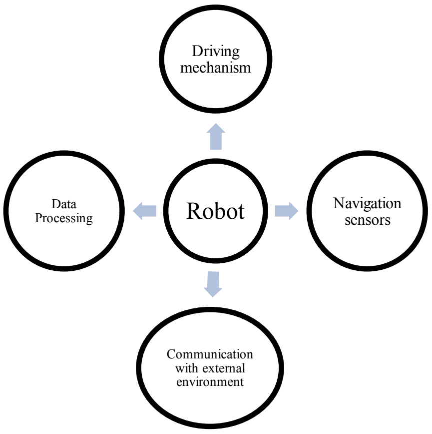


Fig 2: Components of the System

III. DEVELOPMENT OF THE PROTOTYPE

A. Driving mechanism

1) Differential drive

This mechanism is one of the simplest diving mechanisms where, two drive wheels are mounted on a common axis and each wheel can be independently driven in both directions [5].Fig 3 The simplicity of this mechanism has made it one of the most popular driving mechanisms used for indoor robotic applications such as vacuum cleaners.

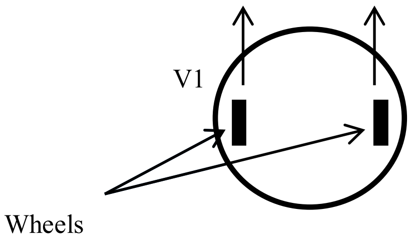


Fig 3: Differential Drive

For the driving purpose different kinds of motors can be used such as full rotating servos, stepper motors, brushed DC motor, brushless DC motors etc. As the servo motors operate in a closed loop control system; therefore it has an advantage of being able to use for differential driving mechanisms. But their cost is relatively high and consumes much higher power.

Generally the low cost solution for differential driving mechanism is brushed DC motors. But as this type of a motor is an open loop system, brushed dc motor alone is not very suitable for differential driving mechanism. For that, an encoder can be attached to the motor to get a feedback on speed. For the platform in concern, a differential driving mechanism with brushed motors and encoders is used.

In employing differential drive mechanism we used two 6v brushed DC motor as it is the cheapest option available.

In order to increase the torque to the wheels, a gear reduction unit is used in between the motors and the wheels.

2) Direction and Speed control

Direction of the wheel rotation can be simply changed by altering the current supplying direction. For that an H-Bridge circuit is used for each motor. The chip here used is L298 which is a 2A dual H-Bridge. The speed controlling is achieved by using PWM (Pulse Width Modulation) signal.

B. Navigation

Navigation is one of the key aspects in robotic applications, where the robot determines its location and makes required movements.

For the upgradable mobile platform in concern, it is required to have at least a basic navigation system. Later on as per the preference of the user; he should be able to upgrade the mobile platform using advanced vision based navigation system.

Navigation can be basically divided into two main sections as

- Outdoor navigation
- Indoor navigation

Outdoor navigation mainly focuses on navigation in areas where GPS (Global Positioning System) could be used as the navigation approach. For that, one of the key requirements is clear LOS (Line of sight) [6].

But in indoor environments GPS is not a practical approach where LOS is not always possible.

Therefore dead reckoning techniques are used in this platform.

1) *Dead reckoning*

Dead reckoning is one of the oldest navigation method used in robotics. The main concept behind this is to calculate the travel distance of the robot from a reference point. The key drawback in this method is caused by the imperfect nature of the sensors used whereby the sensor drift leads to accumulation of error over travel distance.

One of the most commonly used equipment for dead reckoning is wheel encoders. Wheel encoders can be used to get the number of rotation of the wheel. In ideal conditions, if the diameter and the number of rotations of the wheel are known, it is possible to simply calculate the travel distances using basic mathematics. But in practice, a number of errors can occur due to the uncertainty of wheel diameter and the wheel slipping [5] [7].

When using an accelerometer for dead reckoning purposes; even though wheel slipping and wheel diameter doesn't affect the accelerometers, it suffer from offset ,drift and proportionally increasing error caused by double integration of acceleration to predict the travel distance [8].

In addition to the travel distance measurements, it is also required to measure the robot's current angle. Traditional approach for this is to use a digital compass. But a digital compass is highly sensitive to the changes of the surrounding magnetic field. Thus it is not feasible to totally rely on the compass data.

Another approach is to use a Gyroscope. It can be used to measure the relative rotation angle and it is not affected by the changes to the surrounding magnetic field. But the problem with this approach is that it is not possible to detect the absolute angle.

In this paper a method is proposed whereby a combination of above equipment could be used to reduce errors occur in the navigation of the platform.

2) *Navigation approach used in the platform*

Here encoder data is used to calculate the travel distance. The encoder used here is an 8 CPR incremental type. In order to estimate the angle of the robot both the Gyroscope and the digital compass is fused (Fig 4). Here the digital compass is only used to get the initial angle. Then the gyroscope is used to calculate the relative angle.

This proves to be effective to counter the distortion created by the magnetic fields present in indoor environments due to electrical equipment such as speakers and motors.

MPU6050 IMU unit is used to take the gyroscope readings. The module is equipped with the I2C communication medium. For the Digital compass, we used HMC5883L with I2C communication medium.

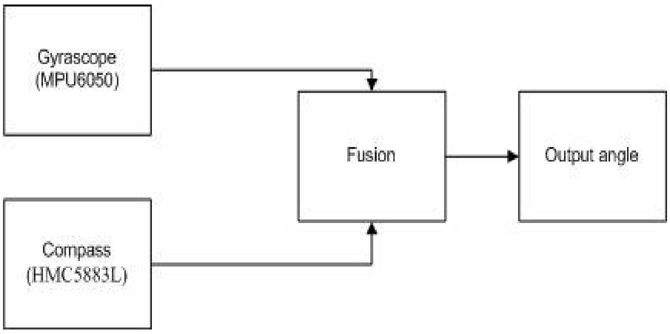


Fig4: Sensor Fusion

C. *Obstacle Avoidance and Mapping*

Some of the most popular types of sensors used in obstacle avoidance are Sonar, IR, Laser and Optical cameras [9]. Even though the Laser range sensors give good accuracy, the cost of the device is very high making it not appropriate for low cost commercial products. The optical cameras also require high amount of processing power, making is difficult to be used for a low cost applications.

The IR and Sonar are the cheapest sensors used for navigation purposes. Both of these sensors have their advantages and disadvantages. In this paper Sonar sensors are used due to its relatively high detection range (>4m) compared to the IR (1.5m) in commercially available modules.

SR04 sonar sensor was selected as the sensor could take measurements in the range of approximately 5cm to 350cm.

D. *Sensor Arrangement*

1) *Arrangement 1*

At the initial testing phase, we mounted a Sonar sensor on a stepper motor such that it can be rotated in a half circle.

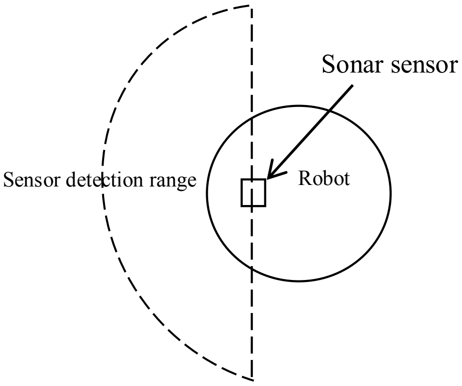


Fig 5: Rotating Sonar Sensor

The advantage of this method is that it requires only a single Sonar sensor to identify obstacles located inside the dotted area (Fig 5).

But the main drawback of this method occurs in a dynamic situation where the robot moves some distance during a scanning cycle. This makes it difficult to locate some of the obstacles. This together with the mechanical imperfections of the sensor and alignment errors in the rotating mechanism demands a better arrangement.

2) Arrangement 2

In the second arrangement, three sonar sensors were mounted 90 degrees to each other on the front side of the platform as seen in Fig 6.

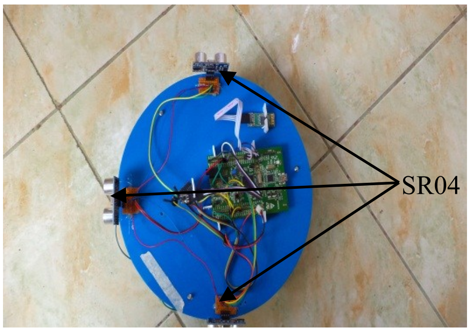


Fig 6: Fixed Sensor arrangement

The sensors mounted on the sides of the platform are used to generate the map while the sensor in the front detects the obstacles. The sensor has naturally a detection angle of 15^0

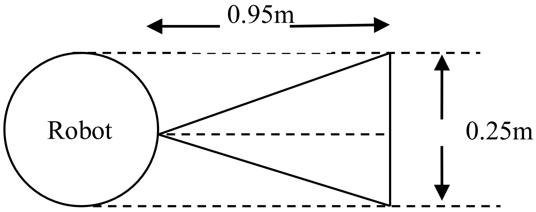


Fig 7: Front sensor coverage

Due to this property, the front mounted sensor can detect obstacles that can be collided with the robotic platform, whose diameter is 0.25m at a distance of approximately 1m as seen in Fig 7. However in contrast to this arrangement, in the previous arrangement, the detection distance is shorter. In other words, in this method the robot needs to detect the obstacle from a distance of approximately 1m.

The task of the two sensors mounted on the sides can be explained as below. This is especially used when the robot is trained to follow a certain path for the first time. While the robot is made to follow the path, the side sensors will detect the distances to obstacles on each side of the path. Hence while following the path, the robot will generate a map of the surrounding environment and keep this in memory so that when asked second time to follow the same path, it knows where exactly obstacles are. This map will be verified in subsequent travelling.

E. Communication

A communication medium with a remote computer is required to generate the map of the surrounding environment. For that we used a Bluetooth communication module with serial connectivity which can be used to communicate with an external computer running MATLAB r2013a.

F. Controlling Module

Purpose of the controlling module is to process the acquired data from the sensors and to control the actuators.

For the platform, STM32f4 Discovery board is used. This development board is equipped with ARM cortex M4 micro controller with more than 80 GPIO's .This gives a sufficient flexibility to carry out further developments without switching to another development board.

The development board was basically selected due to its high Performance to Cost ratio compared to most widely available development platforms.

G. Hardware Summary

Table 1 displays a summary of the components, used in our project.

Table 1: Component List

Feature	**Used component**
Motors	6v DC brushed
Controlling unit	STM32F4 Discovery Board from ST microelectronics
Communication	Bluetooth rs232 module
Wheels	2x 6.5cm diameter wheels +2x Ball wheels
Power supply	6800mAh Li battery pack
Accelerometer +Gyro	MPU6050
Encoders	8CPR incremental
Compass	HMC5883L
Sonar Sensor	SR04

Fig 8 shows the final testing prototype.

Fig 8: Final testing Prototype

IV. ALGORITHMS

In indoor navigation scenario, it is not always required to know the whole map of the environment. For a domestic robot in many cases there are default locations that the robot is required to navigate such as living rooms, kitchen etc. In that case the user can initially guide the robot to the required locations.

This method is applied in this paper, where an RC (Remote Controlling) unit is given to the user, which can be used to manually navigate the robot to the required location and assign the location. During this process, the robot generates its own map.

For the remote controlling purpose, Bluetooth connectivity is used.

A. *Operational Flow*

The operation of the platform can be basically summarized into following chart in Fig 9.

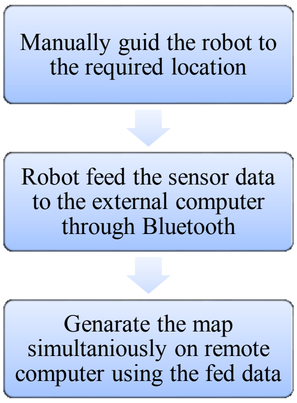


Fig 9: Operational Flow

After robot generates the map, we can point the required location in the map and the robot travels to that particular point.

B. *Algorithm in the Robotic Platform for Map Generation*

Fig 10 shows the algorithm used by the platform for map generation. Data acquired here will then be fed to MATLAB.

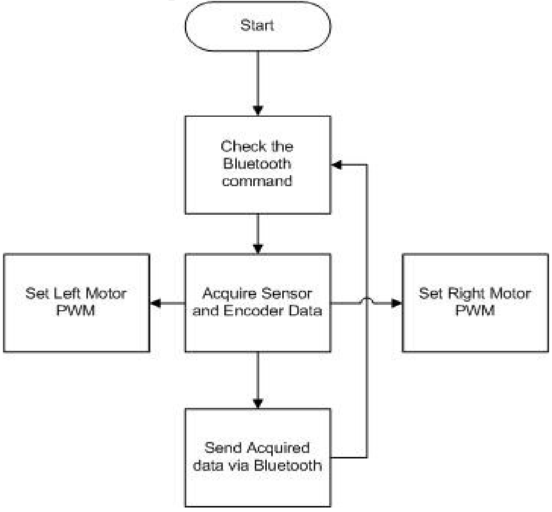


Fig10: Map Generation Algorithm

C. *Algorithm for correcting dead reckoning errors*

Main reasons for errors in dead reckoning distance measurements are wheel slipping and unevenness in the terrain. The amount of error of the measurements depends on the type of floor on which the robot moves. Therefore in order to obtain optimum results, an initial calibration can be used. However due to the wide variety of floors the robot can encounter in domestic environments it is not possible to factory calibrate it for each and every floor type. Hence a self-calibration algorithm is feasible to be used. In doing so it is assumed that the surface condition of the entire floor on which the platform operates is uniform.

The platform is equipped with a differential drive mechanism and the following steps could be used on the initial setting up of the robot to a new floor type.

1. Drive only one of the wheels such that the platform turns around the other wheel as shown in Figure 11.

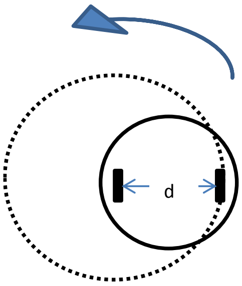


Fig 11: Platform Rotating Path

2. Turn the robot a full rotation using the compass and measure the encoder count (C).
3. The travel distance of the robot (l) can be calculated from the following formula as the radius of the traveling circle is a known parameter (Distance between two wheels d).

$$l = 2\pi d \qquad (1)$$

4. Take encoder readings by performing n number of rotations and find the corresponding travel distance.

$$l = 2\pi d \times n \qquad (2)$$

5. The data set which is arrived at is a comparison between an actual distance and an encoder measurement. This can then be interpolated to predict the dead reckoning error in all travels on that floor.

This self-calibration method will be further tested to verify its applicability in all circumstances, especially to determine its behavior over working time.

V. GRAPHICAL USER INTERFACE(GUI)

In order to control the platform wirelessly and to display the acquired data, a GUI was developed using MATLAB r2013a. The GUI is equipped with keys to control the platform manually. The bidirectional data flow enables the GUI to generate the map simultaneously while navigating as seen in Figure 12.

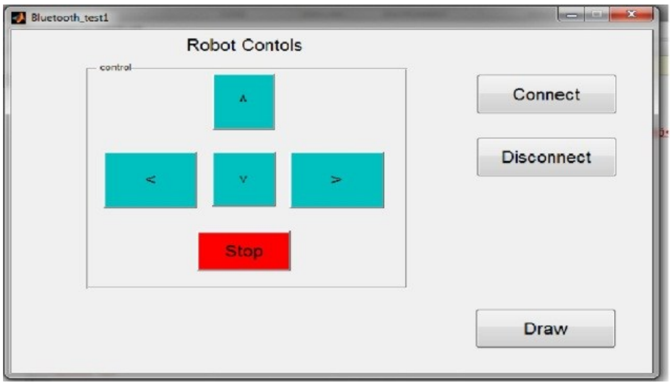


Fig12: Controlling GUI

VI. EXPERIMENTS AND RESULTS

A. *Map Generation*

After implementing the algorithm in the robotic platform, a map was generated in the testing arena using MATLAB r2013a Mapping toolbox (Fig 13). When doing this Kalman filtering is used for error filtering .The red colored area indicates the walls of the testing arena.

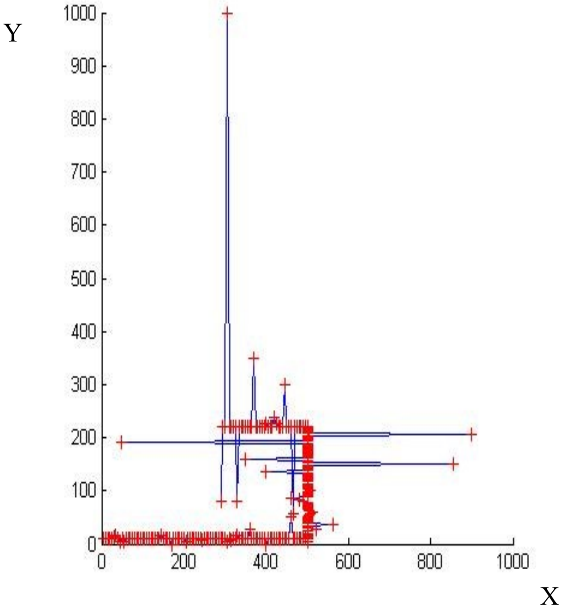


Fig13: Generated Map

B. *Dead Reckoning Distance prediction*

An experiment was performed to estimate the travel distance of the robot with the encoder readings for the given algorithm above.

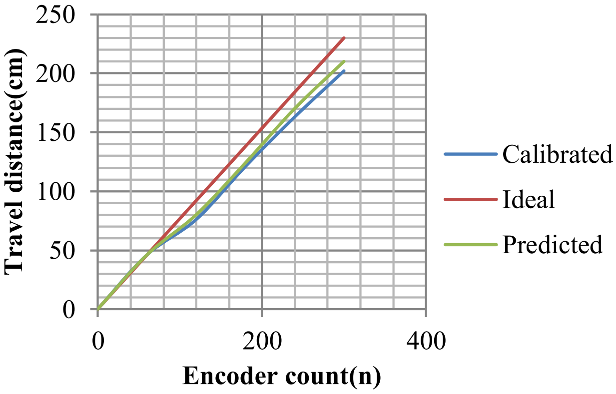


Fig14: Predicted Travel Distance

In Fig 14, the Ideal line which is indicated in color red gives the theoretical travel distance that the robot travels when there is no wheel slipping. The Calibrated line indicated in the color blue gives the travel distances measured by the self-calibrating algorithm. And the Predicted line indicated in the color green shows the predicted travel distance using the calibrated dataset.

VII. CONCLUSION

In this paper the current progress of the research done focusing on the design, development and working principles of an upgradable mobile platform which can self-navigate in a domestic environment is presented. The context discussed in the paper is the foundation of the autonomous robotic construction for which the learning from the prototype discussed here can be used. The authors intend to further develop the concept keeping the fully autonomous robotic platform as the goal.

For the initial stage a testing prototype was developed using low cost hardware. For the navigation purposes dead reckoning technique were used with wheel encoders. To improve the accuracy of the robots orientation, sensor fusion approach was used with digital compass and gyroscope.

Obstacle avoidance is achieved by using Sonar sensors. Two types of sensor arrangements were tested and finally a fixed sensor arrangement is used.

In order to generate the map of the environment, robots hardware was linked with an external computer running MATLAB r2013a where the robot could be controlled by the GUI explained in the paper. A map of the testing arena was generated in MATLAB.

For further developments, we are focusing on identifying the features of the surrounding environment to correct the dead reckoning errors.

VIII. REFERENCES

[1] P. Tang and T. Venables, "'Smart' homes and telecare for independent living," *Journal of Telemedicine and Telecare,* vol. 6, no. 8, pp. 8-14, 2000.

[2] M. Chana, D. Est`evea, C. Escribaa and E. Campoa, "A review of smart homes—Present state and future challenges," *Computer Methods and Programs in Biomedicine,* vol. 9, no. I, pp. 55-81, 2008.

[3] IRobot Corporation, "iRobot Ava™ Mobile Robotics Platform," 2013. [Online]. Available: http://www.irobot.com/us/learn/commercial/ava.aspx. [Accessed 15 October 2013].

[4] P. Coelho and U. Nunes, "Path-Following Control of Mobile Robots in Presence of Uncertainties," *IEEE Trans. Robot.* , vol. 21, no. 2, pp. 252-261, 2005.

[5] J. Borenstein and L. Feng, "Measurement and Correction of Systematic odmetry Errors in Mobile Robots," *IEEE Trans. Robot. Autom.,* vol. 12, no. 6, pp. 869-880, 10 1996.

[6] S. Panzieri, F. Pascucci and G. Ulivi, "An outdoor navigation system using GPS and inertial platform," *IEEE/ASME Trans. Mechatron.,* vol. 7, no. 2, pp. 134-142, 10 2002.

[7] D. Wang and C. B. Low, "Modeling and Analysis of Skidding and Slipping in Wheeled Mobile Robots: Control Design Perspective," *IEEE Trans. Robot.,* vol. 24, no. 3, pp. 676-687, June 2008.

[8] H. Barshan.B and Durrant-Whyte, "Inertial Navigation System for Mobile Robots," *IEEE Trans.Robot.Autom,* vol. 11, no. 3, pp. 328-342, June 1995.

[9] F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers and W. Burgard, "An evaluation of the RGB-D SLAM system," in *IEEE Trans. Robot. Autom.*, Saint Paul, MN, 2012.