Transactions of the VŠB – Technical University of Ostrava, Mechanical Series
No.1, 2016, vol. LXII
article No. 2011

Radovan SIVÝ*, Daniela PERDUKOVÁ**

VERIFICATION OF SLAM METHODS ON ROS PLATFORM

VERIFIKÁCIA METÓD SLAM NA BÁZE PALTFORMY ROS

Abstract

The paper deals with the problem of experimental vehicle concept for 3D SLAM method based on data from Kinect device. One part of the paper describes the design, implementation and description of the physical model of the experimental vehicle. The second describes an odometry representation with a mathematical model of the Ackerman steering as well. Next describes the ROS platform, which was determined to solve complex robotic tasks. Last part of the paper contains to the evaluation of the results of 3D mapping and localization in unknown space.

Abstrakt

Príspevok popisuje problém konceptu experimentálneho vozidla pre metódu 3D SLAM založenú na spracovaní dát zo senzora Kinect. Časť príspevku popisuje návrh, implementáciu a popis fyzikálneho modelu experimentálneho vozidla. V príspevku je taktiež popis matematického modelu vozidla založeného na Ackermanovom riadení, ktorý sa využiva pre odometriu. Príspevok sa venuje popisu platformy ROS, ktorá sa využíva na riešenie zložitých úloh v robotike. Posledná časť je venovaná vyhodnoteniu výsledkov 3D mapovania a lokalizácia v neznámom prostredí.

Keywords

SLAM, Kinect, Mapping, Localization, ROS, DSP

1 INTRODUCTION

The current trends of modern robotics tend to the development of small unmanned autonomous robots capable of performing tasks set in unfamiliar surroundings without full external operator intervention. Paper solves a highly topical issue in the field of modern robotic systems with a focus on the development of a mobile robot using operating systems based on the latest technologies for a more efficient, faster and more reliable to fulfill its goals in an unknown environment. The concept of the robotic system in the context of the mobile robots includes the knowledge of methods for locating the position of robotic systems, the methods for mapping the unknown space [1], [2], [3]. Important is the area of knowledge of sensor systems to create maps of the unknown environment based on camera systems (machine vision). Machine vision is currently the world's most modern trend to substitute expensive sensors operating on the principle of laser or ultrasonic distance measurements. Therefore, this paper describes concept and structure of the

* Ing., Department of Electrical Engineering and Mechatronics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, Košice, tel. (+421)55 602 2283, e-mail radovan.sivy@tuke.sk

** prof. Ing. PhD., Department of Electrical Engineering and Mechatronics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, Košice, tel. (+421)55 602 2696, e-mail daniela.perdukova@tuke.sk

experimental vehicle for processing Kinect data and verification of SLAM methods and algorithms in a real environment. The use of this sensor is currently a worldwide trend in the development of robotic systems [4],[5].

2 EXPERIMENTAL VEHICLE

Many papers verified SLAM methods only in simulations [6], [7] therefore; the main goal of this paper was to build the experimental vehicle for research purposes. The experimental vehicle consists of physical model and mathematical model. A mathematical model was developed for verification of SLAM methods and also for physical model operation in the environment.

2.1 Physical model of experimental vehicle

The design of a physical model of experimental vehicle is based on low-cost sensor system in combination with the chassis used in cars. The aim was also to create a ground vehicle that will be able to undertake a process of simultaneous localization and mapping (SLAM)based only on the processing of the vehicle speed and rotation. The prototype of experimental model consists of the RC car chassis Traxxas Slash 1:10 TQ RTR (Fig.1). This RC car is equipped with DC motor Titan 12T 550, transmission Magnum 272 with hardened steel gears and gear ratio of 2.72 to 1. We choose the RC car model due to its robust design and compact size of the chassis. A physical model of vehicle chassis is based on the Ackermann steering. We choose the Kinect sensor as the main sensor for the experimental vehicle. The experimental vehicle consists of the two main parts: motion and control. The motion part contains elements for the implementation and management of elementary movements of the experimental vehicle. On the other hand, control part contains the elements of the sensor system and provides calculation, data processing and communication.

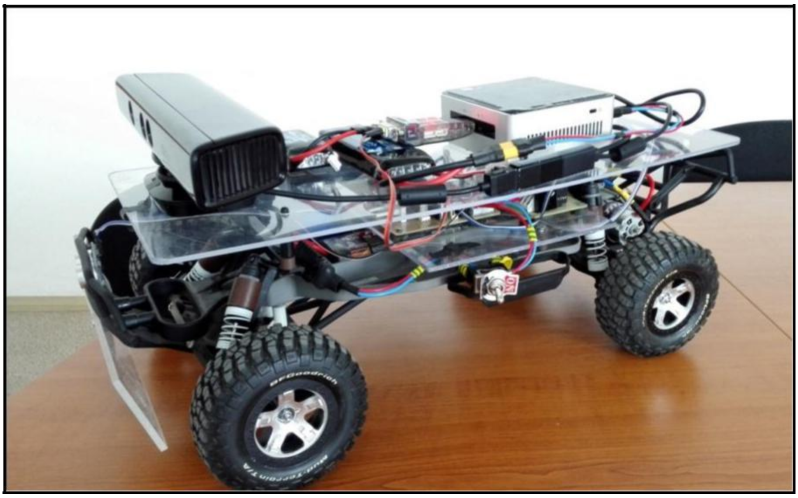

Fig. 1 Physical model of experimental vehicle

The main component for computing and data processing is mini-PC Intel Next Unit Computing (NUC). NUC is a miniature PC, with performance as high as desktop PCs. Therefore, NUC is a great choice as the main computing unit for mobile platforms.

2.2 Mathematical model of experimental vehicle

To verify SLAM methods it was necessary to define the mathematical model of our experimental vehicle. The mathematical model will serve not only to determine the motion (position) of the vehicle during experiments but it also can be used to simulate the movement of a vehicle in a graphical environment based on the ROS platform. Mathematical model of experimental vehicle is based on the Ackerman chassis design. Therefore, we can imagine the experimental vehicle as a rigid body that moves in the plane.

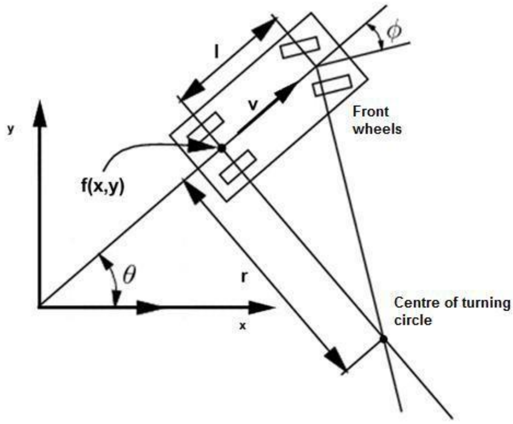

Fig. 2 Vehicle with three degrees of freedom

Fig. 2 indicates several parameters associated with the car. A configuration is denoted by q = (x,y,θ). The body frame of the car places the origin at the center of rear axle, and the x-axis points along the main axis of the vehicle. Let *v* denote the speed of the vehicle. Let ϕ denote the steering angle. We specify the distance between the front and rear axles as *l*. If the steering angle is fixed at ϕ the vehicle travels in a circular motion, in which the radius of the circle is ρ. Note that ρ can be determined from the intersection of the two axes shown in Fig. 2. Using the current notation, the task is to represent the motion of the vehicle as a set of equations of the form

$$= {}_1 \;\; ,,,,\phi$$
$$= \qquad {}^2 \;\; ,,,,\phi \qquad (1)$$
$$= \qquad {}^3 \;\; ,,,,\phi$$

where:

x – position in x-axis $[m]$,

y – position in y-axis $[m]$,

θ – orientation of vehicle $[\circ]$,

v – vehicle speed $[\mathrm{m} \cdot s^{-1}]$,

ϕ – steering angle $[\circ]$.

In a minor interval, Δt, the vehicle must move approximately in the direction that the rear wheels are pointing. In the limit as Δt tends to zero, this implies that *dy/dx = tan(θ).* Since *dy/dx =* $\dot{y}/\dot{x}$ and tan*(θ) = sin(θ)/cos(θ)*condition can be written as a Pfaffian constraint:

$$- \sin + \cos = 0 \qquad (2)$$

The constraint is satisfied if $\dot{x}$ *= cos (θ)* and $\dot{y}$ *= sin(θ)*. Furthermore, any scalar multiple of this solution is also a solution, the scaling factor corresponds directly to the speed *v* of the vehicle. Thus, the first two scalar components of the configuration transition equation are $\dot{x}$ *= v .cos(θ)* and $\dot{y}$ *= v.sin(θ)*. The next task is to derive the equation for . Let *s* denote the distance traveled by the vehicle (the integral of speed). As shown in Fig.2, ρ represents the radius of a circle that is traversed by the center of the rear axle, if the steering angle is fixed. Note that $ds = \rho\ .\ d\theta$. From trigonometry, $\rho = l/tan(\phi)$, which implies

$$= \quad \frac{\tan\phi}{\quad} \tag{3}$$

Dividing both sides by dt and using the fact that w= s yields

$$= \quad - \tan\phi \tag{4}$$

where:

l – distance between axles $[\mathrm{m}]$,

s – traveled distance by vehicle $[\mathrm{m}]$.

So far, we have modeled the motion of the vehicle, but we did not specify any action variables. Suppose that the speed v and steering angle ϕ are directly specified by the action variables u_s and $u\phi$ respectively. The convention of using a u variable with the old variable name appearing as a subscript will be followed. This makes it easy to identify the actions in a configuration transition equation. A two-dimensional action vector, $u = (u_s, u_\phi)$, is obtained. The configuration transition equation for the simple vehicle is

$$\begin{aligned} &= \cos \\ &= \sin \\ &= \ - \ \tan\phi \end{aligned} \tag{5}$$

where:

u_s – speed of vehicle $[m \cdot s^{-1}]$,

u_ϕ – steering angle of vehicle $[\circ]$,

We performed a verification of the mathematical model by setting input velocity and rotation (u_s, u_ϕ). Results are on Fig.3. The green line is model input;the red line represents the output of the mathematical model.

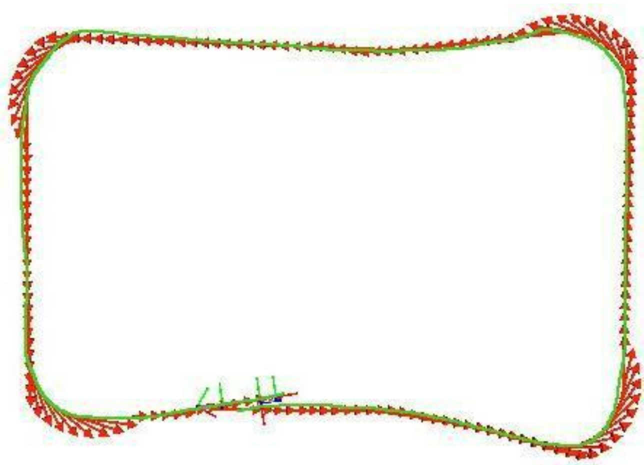

Fig. 3 Verified mathematical model

3 ROS

The Robot Operating System (ROS) is a flexible framework for writing robot software. A collection of tools, libraries, and conventions aims to simplify the task of creating complex and robust robot behavior across a wide variety of robotic platforms[8]. ROS is a trend in the field of robotics and automation, which allows creating a complex real-time control system. These systems can manage difficult tasks at the same time. This is the main reason we consider the ROS as our superior control system. We are using ROS as a tool for evaluating data from Kinect, mapping, localization and visualization of the whole process. The point of this is using existing ROS packages and libraries while we create a new package. Advantages of packages in ROS are wide versatility and "black box" principle. Basically, we can create the own package on the specific robot and then use that package on the different robot, for example much bigger. In addition, the system is based on "black box" principle. We do not need to know what is inside in package (equations, calculations, etc.). We need to know inputs (subscribers), outputs (publishers) and parameters of the package. Therefore, this package for mapping, localization and vehicle control we could apply to any system, which complies given specifications.

3.1 3D SLAM

This SLAM method is based on an algorithm called RGBDSLAM, which has six degrees of freedom (x,y,z, roll, pitch and yaw) [9]. The algorithm uses both the colour (RGB) and the depth (D) image stream as input data. The Fig. 4 visualizes a schematic overview of the algorithm. To use this algorithm it was necessary to define its parameters.

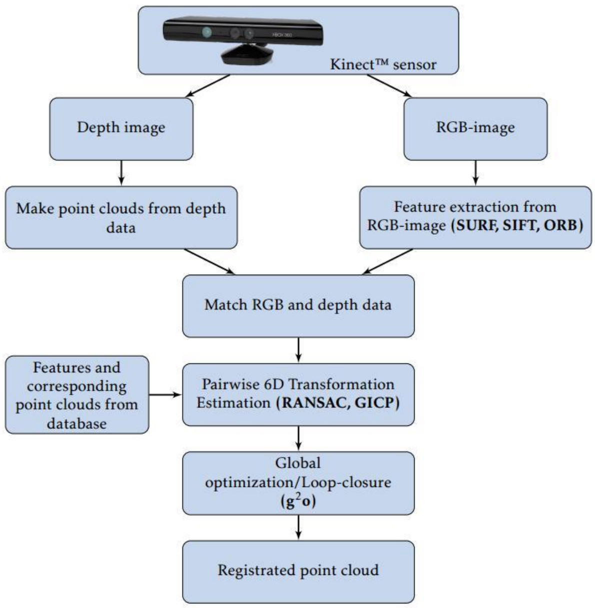

Fig. 4 Schematic overview of the RGBDSLAM algorithm provided as a ROS package [9]

The first step was to determine the parameters to work with Kinect. Parameter /map_filter_angle defines the range of the camera angle, its value was set at 57.3. Maximum depth range image defines parameter /cloud_max_depth range, which in our case is5 m. Parameter /cloud_voxel_size determines the point size (point-UT), therefore, affects the visual display RGB point cloud maps. The smaller the value, the more points are displayed and the image (map) is more detailed and accurate. When displaying a large amount of points problems with the handling of the map may occur. Too many RGB points in point cloud map also affect the processing power of the graphics processor. The Fig.5 shows the resulting map visual SLAM with the value of parameter /cloud_voxel_size= 0.05.

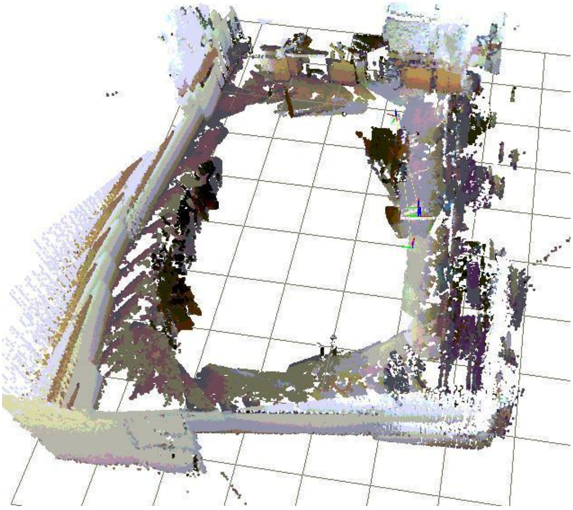

Fig. 5 Map created by method 3D SLAM with parameter /cloud_voxel_size= 0.05

We performed the experimental measurements at the speed of 2 km/h. The resulting map is good but it has a high number of points. Therefore, we are unable to analyze it. In addition, the graphics performance of our NUC is not enough for the smooth handling of the map. Thus, we have reduced the value of the parameter /cloud_voxel_size to 0.1. Results of SLAM methods are in Fig.6.

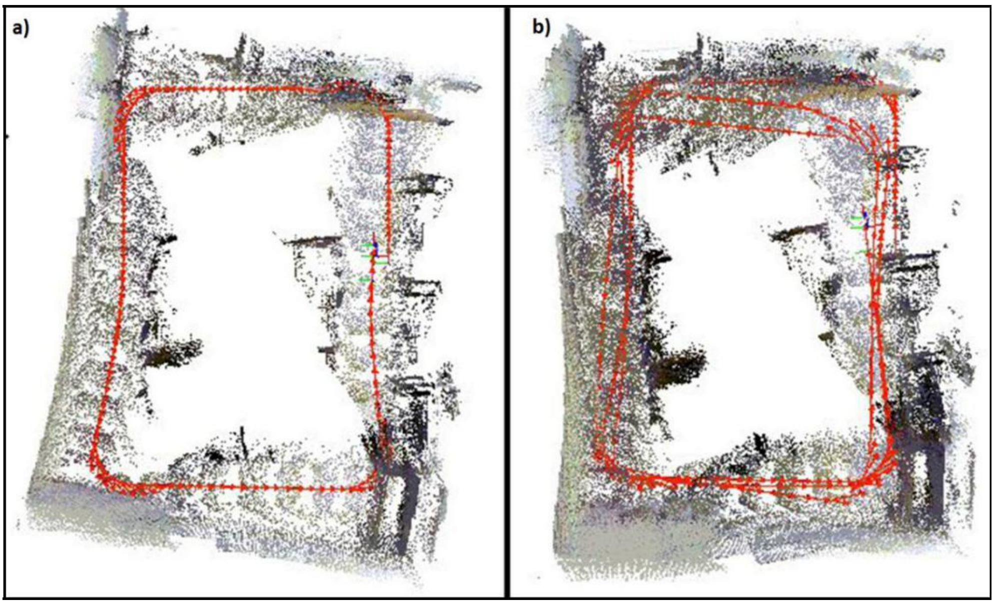

Fig. 6 Map created a) performing one cycle b) performing three cycles

We created the map on the Fig.6a by performing one circle around the room. The map on Fig 6.b was created by driving three circles around the room. By comparing both figures, it is clear that increasing amount of driven circles has improved the quality of the resulting map. By comparing measured result, it is clear that by performing more than three circles the experimental measurements show no significant improvement in the quality of the resulting map. The final quality of map affects the homogeneity of the environment. The term homogeneity of the environment means an environment that lacks elements (shape and color).Image processing algorithm is unable to change the position of the vehicle in this type of environment to update the mapping process without introducing a drift.

Fig.7 shows an impact of the homogenous corner to SLAM process. When a vehicle passes this corner, then SLAM process has a problem to determine the exact estimation of vehicle position based on the image of Kinect. The error can occur when the highest probability of the correlation of two consecutive images in the optimization process acquires the wrong image. Resulting map has too many instances in this homogenous corner. It also may occur small drift in the map.

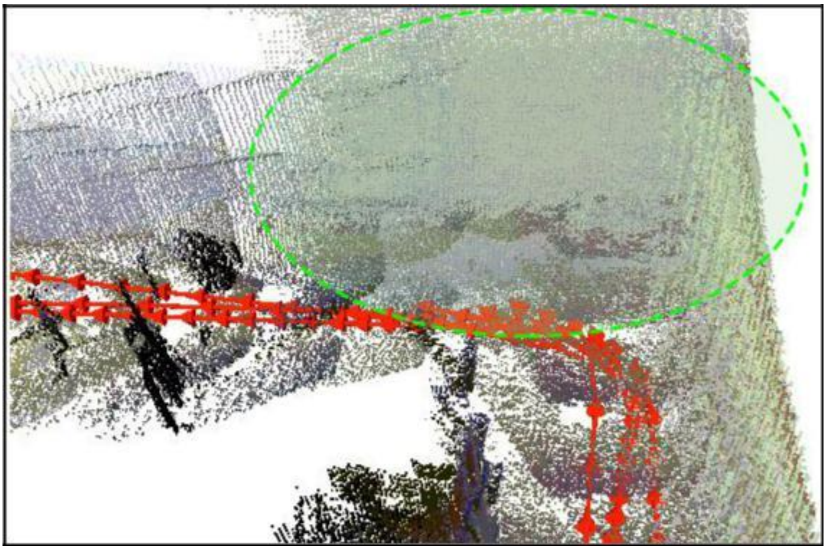

Fig. 7 Homogeneous corner on the resulting map

3 CONCLUSIONS

The goal of this paper was to describe the design and implementation of the experimental vehicle, designing to test simultaneous localization and mapping methods. We define the physical and mathematical model of the experimental vehicle used for SLAM in the unknown environment. We have implemented and verified method 3D SLAM, which handles RGB and depth images from the camera sensor system. The output of this method was the 3D map of the environment. This method is suitable for use with Kinect sensor after some adjustments of its parameters. The method limitation is the demand on computing power, which can be a problem at the low-cost solution. The SLAM method had problems with a homogeneous environment during the measurement. Homogeneous environment brings drift to the process of mapping. We should note that the mathematical model of vehicle worked only with front axle data, speed and angle. We did not use any external measurement components (gyroscope, accelerometer) in our experimental vehicle.

REFERENCES

[1] ENGELHARD N. Real-time 3D visual SLAM with a hand-held RGB-D camera,

[2] HARTMANN J. Real-Time Visual SLAM Using FastSLAM and the Microsoft Kinect Camera, Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on Germany 21-22 May 2012.

[3] TAGUCHI Y. & YONG-DIAN J. & RAMALINGAM S. & FENG CH. SLAM using both points and planes for hand-held 3D sensors. 2012 IEEE International Symposium on USA. Atlanta, 5-8 Nov. 2012.

[4] STOWERS J. & HAYES M. & SMITH A. Altitude control of a quadrotor helicopter using depth map from Microsoft Kinect sensor. In *Mechatronics (ICM), 2011 IEEE International Conference on*. Instanbul 13-15 April., pp. 358-362 ISBN 978-1-61284-982-9.

[5] VAN DEN BERGH M. & CARTON D. & DE NIJS R. & MITSOU N. n LANDSIEDEL C. & KUEHNLENZ K. & WOLLHERR D. & VAN GOOL L. & BUSS M. Real-time 3D Hand Gesture Interaction with a Robot for Understanding Directions from Humans, IEEE RO-MAN, 357-362, 2011.

[6] BAILEY T. & DURRANT-WHYTE H., Simultaneous Localization and Mapping (SLAM): Part II, unpublished.

[7] DISSANAYAKE G. A computationally efficient solution to the simultaneous localisation and map building (SLAM) problem, Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on USA, San Francisco, 28 Apr 2000.

[8] Available online: "http://www.ros.org/about-ros/"

[9] HJELMARE F. & RANGSJÖ J. Simultaneous Localization And Mapping Using a Kinect™ In a Sparse Feature Indoor Environment, Institutionen för systemteknik Department of Electrical Engineering, Linköping, Sweden, 2012