Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnline
CONFERENCE PAPER ARCHIVE

IFAC-PapersOnLine 49-25 (2016) 493–498

Development of a Laboratory Framework for Testing Simultaneous Localization and Mapping Approaches

Petr Neduchal * **Miroslav Flídr** *

* *New Technologies for the Information Society, Faculty of Applied Sciences, University of West Bohemia, Pilsen, Univerzitni 8, Czech Republic (e-mail: neduchal,flidr@ntis.zcu.cz)*

Abstract: This paper deals with the task of developing a laboratory framework for testing Simultaneous Localization And Mapping algorithms (SLAM). The laboratory framework is composed of three parts. The first one are sensors attached to some mobile device like an unmanned ground or aerial vehicle. The second part is an embedded computer which is employed for collecting data from attached sensors and then it sends it to a remote computing node which is the last part of the proposed laboratory framework. The main purpose of the computing node is to solve the SLAM task. For computation and communication between the embedded computer and the computing node the Robot Operating System (ROS) is used. It provides communication layer based on standard ROS message mechanism that simplifies data exchange over a network. The laboratory framework is designed to work with common combinations of sensors used in various SLAM algorithms. Moreover, the laboratory framework makes it possible to replace the real environment or the robot platform with a simulated alternative. The laboratory framework builds on open source projects and on a proprietary software that is freely available for academic use.

Keywords: SLAM, Robotics, Sensors, Cameras, Inertial measurement Units, Encoders

1. INTRODUCTION

Simultaneous Localization And Mapping (SLAM) is an active research area in robotics. The first mention of SLAM occurred at the 1986 Robotics and Automation conference. The goal of SLAM problem is to create a consistent map of an unknown environment using sensors attached to a mobile platform. In the same time, a pose of the mobile platform is estimated in the map. With the expansion of low-cost robotics, the SLAM task becomes more important. There are a number of possible applications in agriculture, industry, entertainment and security. The solution of a task called "search and rescue" is another useful application of SLAM.

In theory, probabilistic formulation of SLAM problem is independent of used mobile platform configuration, type of a map and solution method. The reality isn't usually that straightforward. The particular algorithm which is used for SLAM is influenced by the types of used sensors. This also influences in which way is the map of the environment represented. There are two main approaches to SLAM. They are based on either Bayesian filtering or optimization. The first one uses filter – e.g. extended Kalman or particle filter – and the second uses least square optimization. The first approach is explained in papers of Durrant-Whyte and Tim Bailey Durrant-Whyte and Bailey (2006a), Durrant-Whyte and Bailey (2006b). A principle of the optimization approach is described in paper Grisetti et al. (2010). The book Probabilistic robotics Thrun et al. (2005) is a highly relevant source of information too.

Another classification of SLAM approaches is based on used sensors. There are many possible classes of SLAM which work with particular sensors or their combination. For testing and development of SLAM systems, it is important to choose a set of sensors which are applicable to wide range of approaches. This implies requirements on used unmanned vehicle (UV) and on computational infrastructure for testing purposes.

The main goal of this paper is to describe a unified laboratory framework for testing existing SLAM approaches and a development of new algorithms. It contains three particular tasks. The first one is an analysis of an available SLAM software in terms of used sensors and required computing resources. The second one is a development of various types of unified UVs. And the last one, a development of a uniform testing framework usable for work either in the real world or in a simulated environment. The main contribution of this paper is a description of the development process of the laboratory framework for testing SLAM approaches. The second contribution is a summary of available SLAM approaches based on used sensors.

The paper is structured as follows. In section 2, the problem of SLAM is described. There are mentioned several approaches to solving SLAM. Most of the mentioned approaches use RGB or RGBD camera as an input sensor. Such approaches are called Visual SLAM and it is mainstream at the time of writing this paper. Section 3 contains

2405-8963 © 2016, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.12.089

information about a development of the laboratory framework. It captures a process of creating the framework and achieving goals mentioned in the previous paragraph. In section 4, scenarios for using the laboratory framework are described. Our experiences are summarized in section 5.

2. SIMULTANEOUS LOCALIZATION AND MAPPING

In this section, the problem of SLAM will be described. The goal of SLAM task is mentioned in Section 1. More formally the solution of probabilistic SLAM is defined as searching for joint posterior density function in the form

$$p\left(x_k, m \mid Z_{0:k}, U_{0:k}, x_0\right), \qquad (1)$$

where x_k is a vehicle location and m is the map that contains landmarks. The initial pose of the vehicle x_0, a set of observation $Z_{0:k}$ and all control inputs $U_{0:k}$ are given at time k. A vehicle location x_k and the map m together defines the state space.

The diagram in Fig. 1 shows three steps in SLAM algorithm. UV observes landmarks in the environment and estimates their positions as well as the position of itself. There is shown a difference between green and gray positions. It is called a drift and its reduction is the goal of the SLAM problem.

The SLAM algorithm can be divided into two parts. The first one used for data acquisition and preprocessing called front-end and the second one for the data processing called back-end. Since the front-end contains the abstractions of used sensors its implementation depends on the particular case. Other tasks for the front-end are the feature extraction and the data association which are also dependent on the sensors type. The back-end processes the data and contains the map generation and the map management modules.

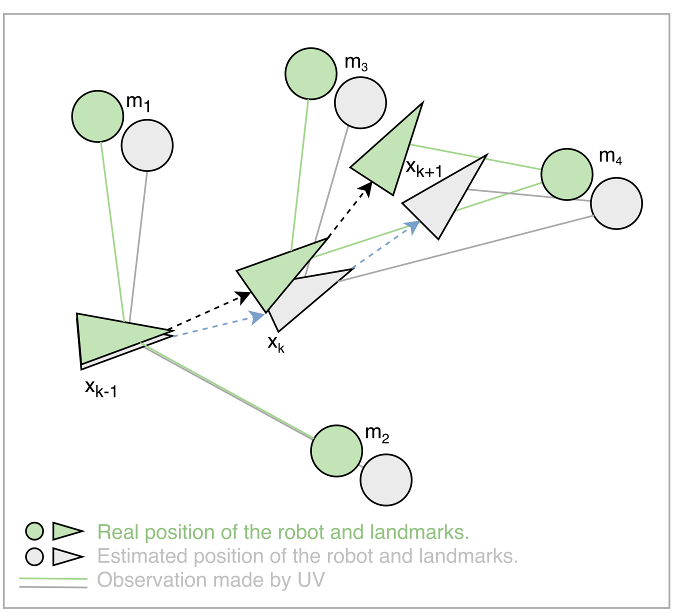

Fig. 1. Diagram of the SLAM task of the UV moving in the environment

In Table 1, the list of the best-known software for dealing with SLAM is shown. Software solutions are arranged based on used sensors. In general, there are three types of sensors.

The first category is non-vision sensors represented by Light Detection And Ranging (LIDAR) sensor which is usually referred to as laser scanning or 3D scanning. LIDAR provides 2D scans of the environment. The data is range-bearing which means that it contains both, the direction and the distance from the vehicle to the observed point in the environment. One of the well-known SLAM algorithms that use LIDAR is the Hector SLAM (Kohlbrecher et al., 2013).

The second category – vision sensors – is the most popular one. It is represented by a wide range of monochrome and RGB cameras. There are a lot of research and software that deals with the Visual-SLAM. A lot of information about the environment is contained in the camera image. As an example, the single camera SLAM system called MonoSLAM created by Andrew Davison (Davison et al., 2007) can be mentioned. MonoSLAM is a feature based – i.e. only extracted features are used from captured image data. The newest mentioned software package for feature based Visual-SLAM is ORB SLAM (Mur-Artal et al., 2015) which uses ORB feature detector and descriptor (Rublee et al., 2011) in multiple tasks of proposed approach. ORB algorithm detects local features in the image – i.e. locally significant pixel – and describes it with a binary vector. Some research is focused on more than one type of the camera. The LSD-SLAM represents what is called direct approach to SLAM. It uses the whole image as the input instead of extracted features. The first research of LSD-SLAM (Engel et al., 2014) deals with the single camera SLAM. In Engel et al. (2015) the version for stereo-cameras was introduced. In the same year the LSD-SLAM for omnidirectional camera (Caruso et al., 2015) was proposed as well. The disadvantage of this sensors is the absence of information about the distance between camera and points. It makes single camera Visual-SLAM harder to solve. Distance or depth information needs to be computed from the motion in the sequence of images. Another possibility is to use so-called RGBD cameras where D denotes depth which can be seen also as the part of previous sensors category. One of the best-known devices of this type is the Kinect device. RGBD Camera provides two images. The first one is classic RGB colour image and the second is the depth image that contains distance information. This sensor is used a lot in last years because there is no need to compute the distance to the particular landmark. The SLAM approach called ElasticFusion by Whelan et al. (Whelan et al., 2015) is one of the well-known systems. ElasticFusion uses GPU technologies like NVIDIA CUDA and OpenGL Shading Language to achieve real-time data processing and Map Managment.

The last category is inertial sensors which are used in the SLAM as a source of a support information about egomotion of the vehicle. One typical example of an inertial sensor is an incremental rotary encoder. Information about angular position of a vehicle wheel is transformed by an encoder to an analogue signal or a digital code which can be used as an information about a speed and a direction of vehicle motion. Other members of inertial sensors are accelerometers and gyro sensors. Both are usually contained