A REAL-TIME LOCALIZATION METHOD FOR AGVS IN SMART FACTORIES

Hakan Temeltas¹
Istanbul Technical University¹ - Istanbul, Turkey
E-mail: temeltas@itu.edu.tr

Abstract— In this study we present a method for the localization problem of Autonomously Guided Vehicles (AGVs) in the framework of well-known Simultaneously Localization and Map Building (SLAM) problem. Methods that are using odometer and IMU together still open to development because of incapability of obtaining significant noise-free measurements from ordinary IMU sensor. Nowadays localization techniques for industrial AGVs are also include specific Landmarks that brings big amount of precision for particularly indoor since that allow sparse mapping without future extraction efforts. This property is also useful for implementation of estimation tools such as Kalman filtering. Our work is based on EKF based localization of based on landmark detection, odometry and IMU measurements together. The proposed method is tested on our AGVs experimentally equipped by various sensors for some navigation scenarios successfully.

Keywords: AUTONOMOUSLY GUIDED VEHICLES (AGVS), SLAM, KALMAN FILTERING, LOCALIZATION

1. INTRODUCTION

One of the major component in Industry 4.0 paradigm is smart factory which can be connected other smart factories or business entities, customers etc., via internet of services [1]. One of the big advantage of the system is use of huge variety of data and reaching an extra computational power. Inside Smart factory basic components are Cyber Physical Systems(CPS) that are appearing as AGV, production units, smart warehouses such as Automatic Storage and Retrieval Systems (ASRSs) [2][3]. All of those CPS are connected via Internet of Things (IoT) and works decentralized manner[4]. For that reasons the AGVs in the smart factories has to be working alone by making some decisions. In order to reach the goal localization of the AGVs is rather important problem to be solved[5].

Localization and map building is today’s one of the challenging research area that attracts so many groups interest in robotics research field. There are several issues such as robust localization and accurate matching of the mapping elements particularly on long range and long endurance operations should be investigated and implemented on real applications. In the absence of global positioning equipment or the situations, which intercept the availability of cumulative error resetting structure, there should be a robust estimation of robot localization against unpredictable disturbances and parametric uncertainties in the system model. So far, there are several studies on localization and mapping problems, which use laser, IMU, vision sensors separately, or in a sensor cluster, in order to correct the dead-reckoning errors in localization and mapping processes of autonomous mobile robots. Early works are using line segments and perspective information for estimating motion [6][7]. Afterwards line matching based works are supported by cross correlation of the image patches in the scene to estimate ego-motion [8]. There are also methods that are using learned landmarks and scale invariant visual landmarks for correction routines,[9]. Laser and odometry fusion is another approach after laser scanner are commercially become widespread in the area[10]. Combination of 2D laser scan and vision is another aspect of view to data fusion [11]. Multi sensor approaches are also frequently used methods in robotics [12]. Highly related works to our study are using maximum likelihood of 3D positions of the feature points and approaches based on minimizing squared error between image frames with using stereo cameras [13], [14]. However, none of those studies listed above can offer efficient solutions to the cumulative error problems in long-range navigation tasks because of their computational efficiencies. In our approach, on the other hand, a robust estimation method based on an EKF based SLAM method with landmark corrections suitable for industrial localizations and this is implemented on AGVs to demonstrate its real time performances. This localization technique will find great opportunity in future smart factories.

The paper is organized as follows. Section 2 defines smart factory as a part of Industry 4.0 paradigm, Section 3 presents the kinematic scheme of differential drive type AGVs. EKF-based SLAM methods and Data fusion of odometer and IMU is studied at Section 4. Robot Operating System (ROS) based implementation of modelling, planning and control actions are given in Section 5. Experimental results for the proposal are presented at Section 6. Conclusions and future works are discussed in a final section.

2. SMART FACTORIES FOR INDUSTRY 4.0

In the smart factories major logistic units are AGVs that make connection between smart warehouse and manufacturing stations as shown in Figure 1. Since paths between the stations are not predefined paths, AGV must plan their paths independently from other AGV’s paths.

![Figure 1: Conceptual diagram for Smart Factories](image)

For this kind of navigation in factory shop floors, precise localization is required. The other important issue in the smart wireless connection between CPS (particularly AGVs) via industrial wireless internet infrastructure. Following chapters consider localization AGVs to be used in advanced manufacturing tasks inside mast factories.

3. KINEMATIC MODEL OF DIFFERENTIAL DRIVE TYPE AGVS

As the study based on practical application for an AGV also called mobile robot platform, there is an entailment on derivation of kinematic equations for two-wheeled differential mobile robot with rear and front castor wheels. Slip-skill effects of motion are not
taken into account in the kinematic model. However, these disturbances on the odometric sensing accuracy are corrected with IMU and vision sensor fusion.

A. Two-wheeled differential drive robot system

Two-wheeled differential drive robots have non-holonomic constraints, which limit the robots movement to the direction of wheels (Figure 2). Addition to that, castor wheels may conduct to nonlinear behavior of robot on startup and sudden movements during the task. On the other hand, in the most cases this effect is considered as disturbance instead of using directly in kinematic equations.

First of all we define $\Delta V_L$ and which are the velocity of left and right wheels respectively. One can write the equation on vehicle wheel velocities as;

$$\Delta V_L = \frac{\pi D_l}{nC_r} N_L, \quad \Delta V_R = \frac{\pi D_r}{nC_r} N_R$$  \hspace{1cm} (1)

Here;

- $D_l$: The diameter of wheels
- $C_r$: Encoder resolution
- $n$: Conversion ratio of gearbox
- $N_L$: Relative change of incremental encoder value of left wheel
- $N_R$: Relative change of incremental encoder value of right wheel

So directional velocity $\Delta V$ and rotational angle $\Delta \theta$ can be found as;

$$\Delta V = \frac{\Delta V_L + \Delta V_R}{2}, \quad \Delta \theta = \frac{\Delta V_R - \Delta V_L}{2}$$  \hspace{1cm} (2)

Here left and right wheel velocities can be found as;

$$V_L = \omega (R + l/2)$$
$$V_R = \omega (R - l/2)$$  \hspace{1cm} (3)

$R$: Radius of turning circle, $l$: width of vehicle

Relative displacement on $x$ and $y$ dimensions and variation in the direction angle can be written in matrix form depending on nonholonomic constraints.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta_L) & 0 \\ \sin(\theta_L) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta V_L \\ \Delta V_R \end{bmatrix}$$  \hspace{1cm} (4)

Thus, calculation of current vehicle location from previous location with respect to sampling time instants can be found as;

$$x_k = x_{k-1} + \Delta V \cos(\theta_{k-1} + \Delta \theta)$$
$$y_k = y_{k-1} + \Delta V \sin(\theta_{k-1} + \Delta \theta)$$
$$\theta_k = \theta_{k-1} + \Delta \theta$$  \hspace{1cm} (5)

Here $F_x$ and $F_u$ are the partial derivatives of nonlinear system w.r.t $x$ and $u$ respectively;

$$F_x = \frac{\partial f(x,u,w)}{\partial x}, \quad F_u = \frac{\partial f(x,u,w)}{\partial u}$$  \hspace{1cm} (9)

Figure 2: Kinematic model of differential drive robot

4. EKF BASED SLAM METHODS

Extended Kalman Filter(EKF) based SLAM methods are one of the most commonly used algorithms for robot localization and environmental map building in the literature. In spite of the fact that linearization step in the EKF algorithm annihilates the high order effects of nonlinear systems, still has a application area in relatively lower speeds at indoor where the environmental effects are approximately linear (i.e. low order nonlinearities). Basic derivation of EKF-SLAM algorithms, which are used in this study, is given at the following section. Later on data fusion of odometry and IMU algorithms is demonstrated without vision part.

A. Derivation of EKF-SLAM

A schematic structure of an EKF is given at Figure 3, EKF algorithm is constituted at two main parts. First part is the prediction step where the current state is estimated from previous state through nonlinear system model;

$$\hat{x}_{k|k-1} = f( x_{k-1|k-1}, u_k, w )$$  \hspace{1cm} (6)

$$P_{k|k-1} = \Delta F_x \, P_{k-1|k-1} \, \Delta F_x^T + \Delta F_u Q \Delta F_u^T$$  \hspace{1cm} (7)

$$z_{k|k-1} = h( \hat{x}_{k|k-1}, v )$$  \hspace{1cm} (8)

Here $F_x$ and $F_u$ are the partial derivatives of nonlinear system w.r.t $x$ and $u$ respectively;
Second part of the EKF algorithm is the update state where the current measurements are taken into account and compared with predicted measurement values. Difference between observed and predicted values is the prediction and system modeling error, which is used for correction of state estimation:

\[ e_k = z_k - \hat{z}_{k|k-1} \]

Kalman gain \( K_k \) is demonstrated as follows:

\[ K_k = P_{\hat{x}|k-1} H_k^T (H_k P_{\hat{x}|k-1} H_k^T + R_k)^{-1} \]

So, state and covariance update can be written together using Kalman gain:

\[ \hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k e_k \]  \hspace{1cm} (12)
\[ P_{\hat{x}|k} = (I - K_k H_k) P_{\hat{x}|k-1} \]  \hspace{1cm} (13)

SLAM algorithms are basically separated into two different parts. First part is the localization term, which is definitively related to vehicle position, and orientation terms. Second part of the algorithm concerns landmark positions, which are evaluated with augmentation of state vectors and covariance matrices. The key point here is the data association task, which is a matching paradigm with observed feature points and estimated feature points from previous states. Data association can be applied as local feature matching and global feature matching. Latter one is also known as loop closure which is the re-identification of features after a close loop path which provides global resetting of cumulative errors on both vehicle and feature localizations.

B. Data Fusion of Odometry and IMU sensors

Using only Odometry values to determine the localization of autonomous robots cause large cumulative dead-reckoning errors in time. To remove this characteristic of odometry we use an IMU sensor in order to reduce the errors especially in rotational motions.

Autonomous robot localization relative to the previous localization is shown in Figure 4. Here rotational movement is corrected by IMU measurements. Corresponding system model can be stated in this situation as;

\[ \begin{bmatrix} x_k \\ y_k \\ \psi_k \end{bmatrix} = \begin{bmatrix} x_{k-1} + \Delta x_k \cos(\psi_{k-1}) - \Delta y_k \sin(\psi_{k-1}) \\ y_{k-1} + \Delta y_k \sin(\psi_{k-1}) + \Delta x_k \cos(\psi_{k-1}) \\ \psi_{k} = \psi_{k-1} + \Delta \psi_k \end{bmatrix} \]  \hspace{1cm} (14)

Note: \( \cos(\psi) \), \( \sin(\psi) \)

However, if one makes a mention of practical issues, an ordinary IMU sensor is incapable of obtain significant noise-free measurements, exclusively in lower speeds for accelerometers and wide radius turns for gyroscopes. These noise levels might be decreased in precise and well-calibrated gyro and accelerometers, however the usage of such expensive devices on the mobile robot platform does not give a cost effective solution. As a consequence, a better solution in terms of precision and low cost, is required to estimate the robot translation and rotation effectively.

In this study we are dealing with localization errors which are caused by the system parametric uncertainties, system disturbances and sensor noises and also assumptions while deriving system kinematic model equations for the sake of simplicity. Therefore, we are using local feature matching based data association and optical flow based motion extraction which are mentioned in the following sections.

5. ROS Based Application Development

It is needed to have a software basis for ITU-AGVs, as seen in Figure 5 that will be used to develop specific applications for various tasks in the future possible projects, theses and works. ITU-AGVs have built before ROS was developed. At the time when ITU-AGVs built, the software systems that used were different and custom so the software of the robots was written concerning them, which became out-of-date now. As mentioned previously in the Introduction chapter, the goal is to construct a set of applications for the basic problems and needs using up-to-date tools. It is desired to write the embedded code for LLPL so the robots can be communicating with ROS and to develop ROS applications for tele-operation, sensor integration and reading, odometry estimation, data collection and offline map building.
work is divided with a hierarchy. Low Level Processing Layer (LLPL) is responsible for getting commands, communicating motor drivers to drive the motors as desired in the given commands, requesting encoder values and sending them to High Level Processing Layer. High Level Processing Layer is responsible for complex calculations and is the part where the ROS runs. The signal chart of ITU-AGVs can be seen in Figure 6.

A. ROS interfacing

To successfully program ITU-AGV, a node is created so that it would subscribe to the joystick topic and every time the joystick data is received it takes the needed button values.

In the main loop, the node publishes an array of the variables which are configured in the callback function. In order to send the commands to ITU-AGVs, another node is subscribed to the topic in which the array is published and it sends the array to LLPL of ITU-AGVs over a serial COM port. After the nodes are built, teleoperation of ITU-AGVs is successfully achieved. Next goal is to reading stable data from the sensors, since the teleoperation is applicable.

The encoder values are sending over SCI-B as they have configured and the related ROS package publishes laser scan data on ROS environment which is shown in Figure 8.

Similarly, to read Laser sensor data, the related package publishes laser scan data on ROS environment which is shown in Figure 8.

Before using LASER measurement data, the coarse localization data is optioned by fusing IMU and odometry data on an EKF filter implemented in ROS environment. However, there is a ROS package that provides data fusion for IMU and encoder data to estimate the pose of a robot using Extended Kalman Filter (EKF) named robot_pose_ekf [15]. The necessary launch file is created so the nodes that publish IMU data and encoder values are started and the fused odometry information is published on a topic. The node graph can be seen in Figure 9.

The encoder values are sending over SCI-B as they have configured and the related ROS package for Xsens MTi IMU was installed to read IMU sensor information. In order to calibrate the data and make certain settings the software provided by Xsens is used. As shown in Figure 7, the robot is rotated around its center for approximately 90° clockwise and counter clockwise and the yaw angle is plotted from the IMU data.

The encoder values are sending over SCI-B as they have configured and the related ROS package for Xsens MTi IMU was installed to read IMU sensor information. In order to calibrate the data and make certain settings the software provided by Xsens is used. As shown in Figure 7, the robot is rotated around its center for approximately 90° clockwise and counter clockwise and the yaw angle is plotted from the IMU data.
data is replayed and using another package called *g-mapping* an offline map of Control and Automation Department corridor is built as shown in Figure 10.

**6. EXPERIMENTAL WORK**

Experimental work for verification and validation of theoretical arguments is performed in the corridors and other similar common areas in our departments. Detailed figures about experimental environment are illustrated in the following figures. Although there are poor conditions on landmark readings and surface roughness, we obtain acceptable results satisfying correction of vehicle motions for indoor environment.

In this section several EKF based localization studies using passive landmarks and different path pattern are presented for single and double AGVs. Also the experiments are conducted under different $\chi$ parameters which classifies the confidence levels of results. These experimental studies result are listed below:

**A. Application 1: Single AGV**

In this application, a single vehicle (AGV) follows a square path, that each of edge is 4 meters, and surrounded by passive landmarks as shown in Figure 11. Total 32 passive landmarks are located inside or outside of the square path. Along the path the AGV takes 20 sample from the environment. Number of these samples are shown in the Figure 11. After completing a tour on the square path, the errors on $x$, $y$ and $\theta$ are shown in Figure 12 (a), (b) and (c) respectively.

![Figure 11: (a) Path and Landmarks for Single AGV, (b) Physical setup.](image1)

Similarly detection of passive landmarks with their uncertainty metrics, the other term, uncertainty ellipsoids in Figure 12 (d) for $\chi = 0.1$. In same figure errors on $x$, $y$ and $\theta$ are represented under different $\chi$ values. This parameter also changes the level of uncertainty at passive landmarks. For instance, uncertainty at passive landmarks decreases dramatically when the $\chi = 1$ and 10 is chosen as shown in Figure 12 (e), (f).

![Figure 12: Results for Single AGV application](image2)

**B. Application 2: Double AGV**

Main issue in a Smart Factory, number of AGVs are operating simultaneously without central unit organization. We demonstrated decentralized operation for 2 AGVs. They both have linear paths in this application and those 2 paths are surrounded by passive landmarks as shown in Figure 13 (a) and (b).

![Figure 13: (a) Path and Landmarks for Dual AGV, (b) Physical setup.](image3)

Along the path of AGV-1, totally 5 samples are collected by the sensors from the environment and similarly, AGV-2 moves in an orthogonal trajectory with 5 samples. Number of these samples for AGV-1 and AGV-2 are shown in the Figure 13 (a).
In this work we use EKF due to low order nonlinearities and experimentally. Results are satisfied for industrial use the AGV autonomously. Also studies show that the method is scalable since the number of AGV do not affect the individual performance of AGVs.

Feature work is stated as using image patches and vanishing point knowledge to obtain more reliable results. Furthermore, an advance passive landmark including passive laser reflector, RFID tags and QR codes will be studied for more precise localization estimation under uncertainty. Finally, estimation of dynamical movements is another intention.

REFERENCES


