MOBILE ROBOT EGO-MOTION ESTIMATION BY PROPRIOCEPTIVE SENSOR FUSION

Jose-Luis Blanco, Javier González, Juan-Antonio Fernández-Madrigal
Department of System Engineering and Automation, University of Malaga (Spain)
{jlblanco,jgonzalez,jafma}@ctima.uma.es

ABSTRACT

For any mobile robot it is a major issue that of estimating its position into the working environment. Although this task is partly carried out through external sensors, incrementally computing the ego-motion of the robot using proprioceptive sensors still is a fundamental step to obtain an estimation of the robot displacement. In this work we deal with the sensor fusion problem for the case of a mobile robot equipped with an odometer and an inertial sensor (a gyroscope). We address this problem rigorously through its formulation as a probabilistic estimation problem, developing an efficient solution in the form of an Extended Kalman Filter (EKF), which can be easily implemented in the low-level firmware of a real mobile robot. Experimental results reveal a qualitative improvement in the robot pose estimation for our sensor fusion system when compared with odometry only, which is the most wide spread technique in commercial robots.

1. INTRODUCTION

For mobile robots to become of practical utility in the industry or in the service sector, it is a fundamental prerequisite that they have a certain degree of autonomy. From the set of abilities that this implies for a robot, a remarkable one is to estimate and track its position within the operation environment, namely localization. Many practical robot tasks (e.g. navigation, picking and delivering) show up the importance of such ability. Localization is an issue extensively studied in the robotics community, where probabilistic approaches have demonstrated to be the most effective and promising ones [6]. In those approaches a fundamental constituent is the probabilistic, incremental estimation of the robot displacement for close time steps. That is the issue addressed in this work: how to obtain an optimal estimation (under Gaussianity assumption) of the robot displacement from different ego-motion sensors on the robot.

In spite of a number of proprioceptive ego-motion sensors existing for ground mobile robots [2], odometers are included into virtually all commercially available ones. Actually, in most cases odometry is the only ego-motion sensor on the robot. Although other proprioceptive sensors like inertial measurement units (IMUs) may provide valuable information to the displacement estimation, they are not usually integrated into commercial robots. Our aim is to integrate different kinds of ego-motion sensors into a mathematically grounded way, concretely probabilistic Bayesian estimation, while proposing a solution efficient enough (an EKF [4]) to be integrated into the low-level firmware onboard of a real robot. The utility of sensor fusion is revealed by noticing that different sensor weaknesses and advantages may complement to each other: typically, an odometer provides a quite precise estimation of translational movements but performs poorly when the robot turns. In turn, IMUs typically measure rotations more precisely than translations, due to the additional time integration required in the latter.

The rest of this paper is outlined as follows. In section 2 we describe the robot kinematics and the working principles of the considered ego-motion sensors. Next we set up the mathematical formulation involved in sensor fusion, whose implementation in the real system is discussed in section 4. Finally, experimental results are reported in section 5, and we present some conclusions.

2. PROPRIOCEPTIVE SENSORS

In this work we consider a robotic wheelchair [3] equipped with two ego-motion sensors: an odometer, and a gyroscope [1]. We describe next the general kinematic model of this robot and its relation with each of the sensors.

Assuming that the mobile robot moves in a planar environment, its pose is completely defined by its 2D coordinates \((x, y)\) and its heading angle \(\phi\), as sketched in Figure 1. The kinematic model is the well-known “tricycle model”, where the robot is constrained to move...
in circular paths only. Provided that the robot pose is sampled at a high rate (in relation with its speed), it is reasonable to approximate the real robot path by a sequence of short circular arcs. Odometry sensors are composed of two encoders (one on each motor wheel), from whose readings the change in the robot pose ($\Delta x, \Delta y, \Delta \theta$) can be computed. On the other hand, a gyroscope is an inertial sensor which measures the instantaneous change rate of the robot orientation $\dot{\phi}(t)$, that is the yaw-rate $\frac{\text{d} \phi}{\text{d} t}$. Please refer to Figure 1 for an illustration of how these variables relate to the robot kinematics.

Since each sensor has its own error sources and they measure different variables from the kinematic model, their combination into a single, optimal estimation is not straightforward. How to perform this is the issue discussed in the next section.

3. SENSOR FUSION

In this work we employ an EKF [4] for fusing the readings from heterogeneous sensors. This filter is an iterative Bayesian filter, where at each instant of time a probability distribution for the system state is kept. An EKF represents probability distributions through multivariate Gaussian distributions, that is, a mean value and a covariance matrix. This distribution is modified according to actuations on the system (prediction step) and next it is corrected according to the sensors measurements (update step). Probabilistic models are required for both the evolution of the system and for the sensors. In the following we present the complete design of an EKF filter for the problem of tracking the pose of a mobile robot equipped with proprioceptive ego-motion sensors only. We will take odometry readings as the action of the robot, whereas gyroscope is considered a sensor of the system state because it is completely passive.

Let $\mathbf{x}_k$ be the state of our system at the discrete time-step $k$:

$$\mathbf{x}_k = (x_k, y_k, \dot{\phi}_k, \phi_{k-1})^T$$

where the memory term $\phi_{k-1}$ stands for the robot orientation at the last time step. If we define $\Delta t$ as the filter sampling period, the memory term allows us to approximate the robot angular velocity $\omega_k$ as:

$$\omega_k \equiv \frac{\phi_k - \phi_{k-1}}{\Delta t}$$

Let the estimation at time step $k-I$ be given by the normal distribution $\mathcal{N}(\hat{\mathbf{x}}_{k-1}, \mathbf{P}_{k-1})$, with $\hat{\mathbf{x}}_{k-1}$ and $\mathbf{P}_{k-1}$ being the mean and the covariance matrix, respectively. Then, the prediction step of the EKF filter reads:

$$\hat{\mathbf{x}}_k = f(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_k)$$

$$\mathbf{P}_k = \left[ \begin{array}{cc} \mathbf{P}_{k-1} & 0 \\ \mathbf{V} \mathbf{C}_k \end{array} \right] \left( \begin{array}{c} \mathbf{V} \mathbf{F}_k \mathbf{P}_{k-1} \mathbf{F}_k^T \\ \mathbf{V} \mathbf{C}_k \end{array} \right)$$

with $f(\cdot)$ being the transition function of the system, and $u_k$ the action performed at time step $k$ whose covariance matrix is $\mathbf{C}_{uk}$. The minus sign used as a superscript in (3) means that the estimation is the prior in the Bayesian filter, that is, sensor observations have not being incorporated into the estimation yet.

Since odometry readings are considered the robot actions, we have $u_k=\delta_k=\Delta x, \Delta y, \Delta \theta$ and, according to the robot kinematic model, the transition function becomes:

$$f(\hat{\mathbf{x}}_{k-1}, u_k) = \left( \begin{array}{c} \hat{x}_{k} \\ \hat{y}_{k} \\ \hat{\theta}_{k} \\ \hat{\phi}_{k} \\ \hat{\theta}_{k-1} \\ \hat{\phi}_{k-1} \end{array} \right) = \left( \begin{array}{c} \hat{x}_{k} \\ \hat{y}_{k} \\ \hat{\theta}_{k} \\ \hat{\phi}_{k} \\ \hat{\theta}_{k-1} \\ \hat{\phi}_{k-1} \end{array} \right) + \Delta x \cos \hat{\phi}_{k-1} - \Delta y \sin \hat{\phi}_{k-1}$$

(4)

Since it is straightforward to obtain from (4) the Jacobian matrices $\mathbf{V}_{x_k}f$ and $\mathbf{V}_{u_k}f$ required to evaluate (3), they are omitted here due to space limitation. Next, the update step is performed in the iterative filter:

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k-1} + \mathbf{K}_k \hat{y}_k$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k-1}$$

(5)

where $\mathbf{I}$ is a 4x4 unit matrix, and:

$$\hat{y}_k = z_k - h(\hat{\mathbf{x}}_k)$$

$$\mathbf{S}_k = (\mathbf{V}_{x_k}h \mathbf{V}_{x_k}h^T \mathbf{V}_{u_k}h \mathbf{V}_{u_k}h^T) \begin{pmatrix} \mathbf{P}_{k-1} & 0 & 0 & \mathbf{V}_{x_k}h^T \\ 0 & \sigma_x^2 & 0 & \mathbf{V}_{y_k}h^T \\ 0 & 0 & \sigma_y^2 \end{pmatrix}$$

(6)

where $\mathbf{K}_k = \mathbf{P}_k \mathbf{H}_k \mathbf{S}_k^{-1}$

Basically, this step predicts the expected sensor (gyroscope) outcome through the sensor model $h(\cdot)$, its uncertainty (covariance matrix $\mathbf{S}_k$), and fuse this information with the prior estimation. The gyroscope sensor model could be defined as:

$$h(\hat{\mathbf{x}}_k) = \hat{\phi}(\Delta t) - \frac{\hat{\phi}_k - \hat{\phi}_{k-1}}{\Delta t}$$

(7)

However, the actual sensor readings are analog voltage values, thus we must consider two uncertainty sources in the gyroscope readings $\delta_k$: (i) electrical noise, modelled as additive white Gaussian noise (AWGN) with variance $\sigma_n^2$, and (ii) uncertainty in the actual sensor sensitivity, i.e. the volts to deg/s ratio. To integrate the effects of both error sources into the EKF we rewrite (7) to account for $S_n$ and $S_s$, the actual (unknown) and nominal sensor sensitivity, respectively, and for the noise $n_k$:

$$h(\hat{\mathbf{x}}_k) = \frac{\hat{\phi}_k - \hat{\phi}_{k-1}}{\Delta t} + \frac{n_k}{S_n}$$

(8)

According to data available in the manufacturer supplied datasheet for our gyroscope, an ADXRS401 [5], it seems that the sensitivity of the device, taken as the outcome of a random variable for each specimen, approximately follows a Gaussian distribution. Therefore, it makes sense to estimate the covariance matrix $\mathbf{S}_k$ by linearization of the sensor model in (8), as shown in (6). There, the Jacobians of the function $h(\cdot)$, $\mathbf{V}_{x_k}h$, $\mathbf{V}_{y_k}h$, and $\mathbf{V}_{u_k}h$, describe how uncertainty in the system state $\mathbf{x}_k$, the sensitivity $S_n$, and the electrical noise $n_k$
After each iteration of the filter, we obtain the updated optimal estimation, disregarding linearization errors.

4. IMPLEMENTATION

Next we discuss how the previously exposed theoretical filter has been integrated into the low-level firmware of a mobile robot.

The system has been designed to work in a timely fashion, under strict real-time requirements. At a working rate of 100Hz, the system collects readings from encoders (odometry) and the gyroscope, performs the required preprocessing of signals, and executes an iteration of the EKF as detailed in section 3. A logical overview of the system is provided in Figure 2. The signal conditioning stage is required since our gyroscope (ADXRS401, see [5]) presents a nonratiometric analog output, while analog-digital converters (ADCs) are ratiometric. Therefore, the resulting readings are highly sensitive to electrical noise coupled to the power supply, which is a major issue on a mobile robot, where motors produce large noise while in operation. To solve this problem, both the sensor output voltage and its 2.5V constant voltage reference are converted through ADCs. The purpose of the signal conditioning stage (please, refer to Figure 2) is two-fold: (i) to scale the sensor readings according to the constant voltage reference; and (ii) to remove the sensor offset, that is, to precisely determinate the voltage corresponding to a null yaw-rate. The offset voltage can be easily estimated by averaging over a time sliding window when the robot is very likely at rest, e.g. when odometry does not detect motion for a few seconds. After removing the offset from the gyroscope signal, it still contains high-frequency electrical noise, which does not carry information about the mechanical system. Since the analog circuitry of the gyroscope has been set up for a measuring bandwidth of 5Hz, we can disregard the signal components at higher frequencies as undesirable noise. In our system the noise is filtered out through a Finite Impulse Response (FIR) implementation of a fourth order elliptic low-pass filter, with a nominal pass-band ripple of 0.1dB, 30dB stopband attenuation, and a cut-off frequency of 5Hz. An example of the signal before and after noise filtering is illustrated in Figure 3.

The whole system has been implemented on an ATMEGA128, a low-cost, 8-bit microcontroller from Atmel, which runs at 16MHz. In Figure 4 it is shown the prototype developed in this work, which includes two Micro-Electro-Mechanical Systems (MEMS): the already introduced gyroscope ADXRS401, and a two-axis accelerometer ADXL203, which can be used to detect the gravity vector, i.e. tilt sensing, although such issue is not addressed here. The system runs autonomously and periodically reports the EKF estimation results to a host-PC via a high-speed USB connection. This prototype has been designed with the aim of minimizing costs and weight.

5. RESULTS AND DISCUSSION

Two comparative experiments are reported next, where the robot follows two different paths while its pose is estimated simultaneously from odometry only, and from our sensor fusion system. The actual final robot pose for each trajectory has been determined by a highly-precise laser range scan matching algorithm [1], which we will consider the ground-truth for comparison purposes. The two different paths consist of moving the robot on a twisty forward, and a spinning trajectory, respectively. Results for the first experiment are summarized in Figure 5(a)-(d). It is noticeable the reduction in the final pose uncertainty, both in the robot position and its orientation, for the case of sensor fusion with respect to the odometry estimation only. This is numerically confirmed by the
values of the covariance matrix determinant: $1.5461 \times 10^{-11}$ from odometry only, and $2.0429 \times 10^{-13}$ for sensor fusion. In the case of the spinning trajectory, the robot turns three times, which makes the odometry-only estimation to completely lose the robot orientation, as illustrated in Figure 5(e) and (g). The incorporation of yaw-rate information in the estimation provides an impressive qualitative improvement here: the determinant of the covariance matrix, $3.9538 \times 10^{-3}$ for the odometry-only estimation for a mobile robot. An efficient solution has been proposed for the case of a robot equipped with odometry and a gyroscope, which has been implemented in the low-level firmware of a real robot and runs in real-time. Experimental results demonstrate that the sensor fusion system provides a major improvement in the quality of the robot pose estimation.

REFERENCES


<table>
<thead>
<tr>
<th>Experiment I: Forward</th>
<th>Experiment II: Spinning</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>y</td>
</tr>
<tr>
<td>Odometry</td>
<td>4.194m</td>
</tr>
<tr>
<td>Sensor fusion</td>
<td>4.187m</td>
</tr>
<tr>
<td>Ground truth</td>
<td>4.169m</td>
</tr>
</tbody>
</table>

Table I. Review of experimental results: final estimated pose from each method.

To summarize, in this paper we have presented the problem of proprioceptive sensor fusion for ego-motion estimation for a mobile robot. An efficient solution has been proposed for the case of a robot equipped with odometry and a gyroscope, which has been implemented in the low-level firmware of a real robot and runs in real-time. Experimental results demonstrate that the sensor fusion system provides a major improvement in the quality of the robot pose estimation.