Navigation and Control of Mobile Robot Using Sensor Fusion

Yong Liu
CareFusion
United States of America

1. Introduction

Mobile robot, especially wheeled mobile robot, with its simple mechanical structure and inherent agility, has attracted significant attentions for dynamic environment applications in the last two decades (Pin et al. 1994 and Purwin 2006).
A general mobile robot Guidance, Navigation and Control (GNC) system is illustrated in the following figure. The main function of guidance system is to generate a feasible trajectory command, usually in multiple dimension of freedom (DOF) to achieve a robot task. The objective of control system is to drive the mobile robot following the commanded trajectory with acceptable tracking errors and stability margin. The function of robot navigation system is to provide accurate position, velocity and/or orientation information for the control system and guidance system. A stable and accurate navigation system is the bridge between the guidance system and control system of mobile robots, which ultimately determines the robot performance.

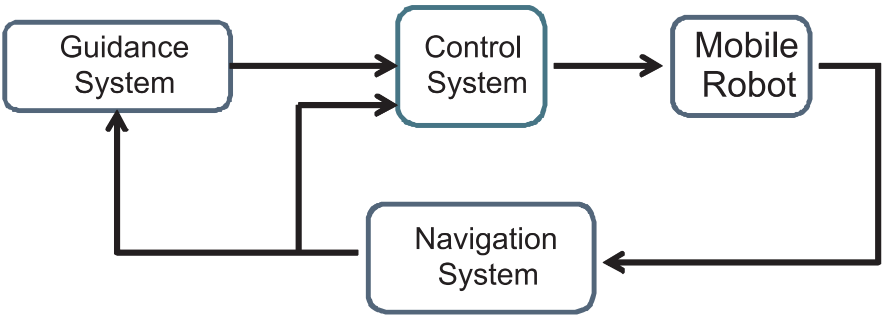


Fig. 1. Mobile Robot GNC System

2. Mobile Robot Navigation System

Mobile robot navigation system, also known as robot positioning system, is to measure the robot's position and/or orientation. On mobile robot, many different sensors and navigation system structures are available (Borenstein et al. 1996).

In mobile robot navigation systems, onboard navigation sensors based on dead-reckoning are widely installed. This type of sensors include various odometry encoder, and inertial navigation system, such as accelerometer and gyroscope. These sensors measure the robot translational and rotational velocity or acceleration rates. By integrating sensor measurements, the position and orientation of mobile robot are estimated, which is known as dead-reckoning. Onboard navigation sensors are usually low-cost with high bandwidth and high sampling rate. Dead-reckoning method is well known suitable for short-term navigation. The advantages of these onboard sensors is that they are totally self-contained. The recent advance on inertial navigation system makes most onboard sensors attractive and affordable. However, the onboard sensors usually have inevitable accumulated errors due to the nature of dead-reckoning process. Onboard sensors require an external reference for continuous calibration.

Many mobile robots are also installed with external absolute position sensors. These sensors include cameras, global positioning system (GPS), infrared radar, active Beacons and artificial landmark recognition. They sense the absolute position and orientation of the robot without drifting. The external sensors are usually working at low bandwidth and sampling rate. The built-in signal processing of these sensors may introduce delay and outlier measurements. If using merely external positioning sensor, the delay and failures of signal processing may lead to a deteriorated performance of robot control system. In mobile robot, vision system is a typical external positioning sensor. By using image processing algorithm, a vision system is able to detect the position or velocity of a mobile robot.

Sensor fusion is a useful technique to combines both types of positioning sensors to provide fast measurement without drifting (Goel et al. 1999).

The main technical challenge of sensor fusion on mobile robot is that robot kinematics is a nonlinear process; therefore the traditional Kalman filter based technique has to be adapted to address the nonlinearity in the system. The sensor fusion system also has to eliminate potential outlier measurement from external sensor and compensate for the different sampling rate between external sensors and onboard sensors.

When using a sensor fusion technique in mobile robot navigation, it should be noted that the fusion process and control system dynamics interact. Additional theoretical analysis and practical consideration should be taken to ensure the stability of the overall navigation and control system.

3. Sensor Fusion for Robot Navigation and Control

3.1 Robot Kinematics and Navigation Equation

Mobile robot dynamics are usually described in two coordinate frames: the body frame {B} and the world frame {W}. The body frame is fixed on the moving robot, usually with the origin at the center of mass. The world frame is fixed on the field. Figure 2 illustrate the the relationship of the two frames.

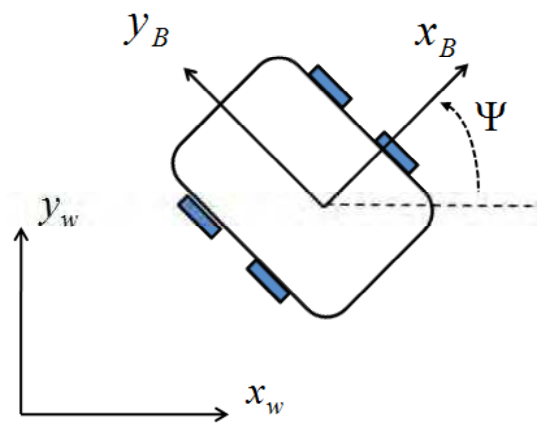
The kinematics of the robot is given by a coordinate transformation from the body frame {B}to the world frame {W}

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\Psi} \end{bmatrix} = \begin{bmatrix} \cos\Psi(t) & -\sin\Psi(t) & 0 \\ \sin\Psi(t) & \cos\Psi(t) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ r \end{bmatrix} \tag{1}$$

where x and y denote the robot location in {W}, Ψ denotes the robot orientation in {W}, u and v are the robot translational velocity in {B}, and r is the robot rotation angular velocity in {B}. Velocity u, v and r are also called body rate.

In order to formulate the robot navigation system with sensor fusion, the robot kinematics (1) are first approximated by using forward Euler method

$$\begin{bmatrix} x_k \\ y_k \\ \Psi_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \Psi_{k-1} \end{bmatrix} + \begin{bmatrix} \cos\Psi_{k-1}\cdot T & -\sin\Psi_{k-1}\cdot T & 0 \\ \sin\Psi_{k-1}\cdot T & \cos\Psi_{k-1}\cdot T & 0 \\ 0 & 0 & T \end{bmatrix} \begin{bmatrix} u_{k-1} \\ v_{k-1} \\ r_{k-1} \end{bmatrix} \tag{2}$$



(a) World frame$\{W\}$ (b) Body frame$\{B\}$

Fig. 2. Mobile Robot Coordinate Frames

where T is the sampling time and k denotes the sampling tick. Equation (2) is a first order approximation of robot kinematics. In this Chapter, it is called navigation equation. The navigation system analysis and design are based on Equation (2). It should be noted that higher order approximation is also available and the sensor fusion method described in this chapter can be applied with minor changes.

In this chapter, without loss of generality, it is assuming that the body rate measurement is from onboard sensors; and the robot position and orientation is from an external absolute positioning sensor, such as a vision system. By defining $\begin{bmatrix} \hat{u}_k & \hat{v}_k & \hat{r}_k \end{bmatrix}^T$ as the body rate measurement, $\begin{bmatrix} w_{1k} & w_{2k} & w_{3k} \end{bmatrix}^T$ as the body rate measurement noise,the body rate measurement model is described as

$$\begin{bmatrix} \hat{u}_k & \hat{v}_k & \hat{r}_k \end{bmatrix}^T = \begin{bmatrix} u_k & v_k & r_k \end{bmatrix}^T + [w_{1,k}, w_{2,k}, w_{3,k}]^T \tag{3}$$

By defining $\begin{bmatrix} z_{1,k} & z_{2,k} & z_{3,k} \end{bmatrix}^T$ as position and orientation measurements by external absolute position sensor, and $\begin{bmatrix} d_{1,k} & d_{2,k} & d_{3,k} \end{bmatrix}^T$ as vision system measurement noise, the vision system measurement model is described as

$$\begin{bmatrix} z_{1,k} \\ z_{2,k} \\ z_{3,k} \end{bmatrix} = \begin{bmatrix} x_k \\ y_k \\ \Psi_k \end{bmatrix} + \begin{bmatrix} d_{1,k} \\ d_{2,k} \\ d_{3,k} \end{bmatrix} \tag{4}$$

Both $\begin{bmatrix} w_{1,k} & w_{2,k} & w_{3,k} \end{bmatrix}^T$ and $\begin{bmatrix} d_{1,k} & d_{2,k} & d_{3,k} \end{bmatrix}^T$ are assumed to be zero-mean white noise with normal distribution, such that

$$p\left(\begin{bmatrix} w_{1,k} & w_{2,k} & w_{3,k} \end{bmatrix}^T\right) \sim N\left(0, Q_k\right)$$
$$p\left(\begin{bmatrix} d_{1,k} & d_{2,k} & d_{3,k} \end{bmatrix}^T\right) \sim N\left(0, R_k\right)$$

where $Q_k \in \mathbb{R}^{3\times3}$ is the body rate measurement noise covariance, and $R_k \in \mathbb{R}^{3\times3}$ is the vision system observation noise covariance.

3.2 Kalman Filter and Its Variations

Kalman filter is an optimal filter design for a class of discrete-time linear stochastic system (Kalman 1960 and Welch et al. 2001). Kalman filter and its various adapted formulas are widely employed in sensor fusion (Hall et al. 2004). The standard Kalman filter is for linear dynamic system with Gaussian noise distribution. In this subsection, the Kalman filter algorithm is briefly reviewed to facilitate deriving the mobile robot sensor fusion filter algorithm. A general discrete-time linear stochastic system is described as

$$X_k = AX_{k-1} + BU_{k-1} + w_{k-1} \tag{5}$$

where $X \in \mathbb{R}^n$ is the system state and $w \in \mathbb{R}^n$ is the process noise. The goal of Kalman filter is to estimate the best X_k by a noisy measurement $Z \in \mathbb{R}^m$, where

$$Z_k = HX_k + d_k \tag{6}$$

d_k represents the measurement noise. The process noise and measurement are usually assumed independent, white with normal distribution

$$p\left(w\right) \sim N\left(0, Q\right)$$
$$p\left(d\right) \sim N\left(0, R\right)$$

The Kalman filter consists of two steps: prediction and correction
(1) Prediction

$$\begin{aligned} \hat{X}_k^- &= A\hat{X}_{k-1} + BU_{k-1} \\ P_k^- &= AP_{k-1}A^T + Q \end{aligned} \tag{7}$$

(2) Correction

$$\begin{aligned} K_k &= P_k^- H^T \left(HP_k^- H^T + R\right)^{-1} \\ \hat{X}_k &= \hat{X}_k^- + K_k\left(z_k - H\hat{X}_k^-\right) \\ P_k &= \left(I - K_k H\right)P_k^- \end{aligned} \tag{8}$$

In the above formula, $\hat{X}_k^-$ is referred as a priori estimate of the true state X_k and $\hat{X}_k$ is referred as a posteriori estimate.
For nonlinear systems, the process matrices A, B and H are not constant. In practice, they are usually derived by linearizing the nonlinear model along some nominal trajectories, which results in time-varying matrices. Techniques such as linearized Kalman filter, extended Kalman filter, unscented Kalman filter and Particle filter have been developed and applied successfully in many practical applications(Brown et al. 1996). Both linearized Kalman filter and extended Kalman filter (EKF) extend standard Kalman filter by linearizing the original nonlinear system (Brown et al. 1996). In a linearzed Kalman filter, linearization is along a predefined trajectory.

The disadvantage of linearized Kalman filter is that the dynamic system state may diverge from the predefined trajectory over time. Linearized Kalman filter is usually used in short-time mission. In EKF, the linearization is about the state estimation. Thus there is a danger that error propagation and filter divergence may occur. To overcome such a disadvantage, unscented Kalman filter (UKF) and particle filters (PFs) are developed. In UKF, the state distribution is approximated by Gaussian Random Variable (GRV) and is represented using a set of carefully chosen sample points (Richard et al. 1983). Particle filters (PFs) are a group of optimal and suboptimal Bayesian estimation algorithms for nonlinear and non Gaussian system (Arulampalam et al. 2001). PFs employ sequential Monte Carlo methods based on particle representations of state probability densities. UKF and PFs require much larger computational power to implement, compared to linearized Kalman filter and EKF.

3.3 Nonlinear Kalman Filter for Mobile Robot Navigation and Control System

In this subsection, the sensor fusion for mobile robot is illustrated using the framework of nonlinear Kalman filter.

First, the navigation equation of mobile robot is rewritten in the canonical formular for Kalman Filter.

$$\begin{bmatrix} x_k \\ y_k \\ \Psi_k \end{bmatrix} = \begin{bmatrix} cos(\Psi_{k-1}) \cdot T & -sin(\Psi_{k-1}) \cdot T & 0 \\ sin(\Psi_{k-1}) \cdot T & cos(\Psi_{k-1}) \cdot T & 0 \\ 0 & 0 & 1 \cdot T \end{bmatrix} \begin{bmatrix} u_{k-1} \\ v_{k-1} \\ r_{k-1} \end{bmatrix} + \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \Psi_{k-1} \end{bmatrix} + \begin{bmatrix} cos(\Psi_{k-1}) \cdot T & -sin(\Psi_{k-1}) \cdot T & 0 \\ sin(\Psi_{k-1}) \cdot T & cos(\Psi_{k-1}) \cdot T & 0 \\ 0 & 0 & 1 \cdot T \end{bmatrix} \begin{bmatrix} w_{1,k-1} \\ w_{2,k-1} \\ w_{3,k-1} \end{bmatrix} \tag{9}$$

The external sensor observation is represented by

$$\begin{bmatrix} z_{1,k} \\ z_{2,k} \\ z_{3,k} \end{bmatrix} = H_k \cdot \begin{bmatrix} x_k \\ y_k \\ \Psi_k \end{bmatrix} + \begin{bmatrix} d_{1,k} \\ d_{2,k} \\ d_{3,k} \end{bmatrix}, \text{ where } H_k = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{10}$$

The sensor fusion for mobile robot navigation is to calculate an optimal estimate of $\begin{bmatrix} x_k & y_k & \Psi_k \end{bmatrix}^T$,from the onboard measurement $\begin{bmatrix} \hat{u}_k & \hat{v}_k & \hat{r}_k \end{bmatrix}^T$ and external sensor reading $\begin{bmatrix} z_{1,k} & z_{2,k} & z_{3,k} \end{bmatrix}^T$. It should be noted that equation (9) is a linear process if either the robot is stationary or does not rotate. In either cases, standard linear Kalman filter can be applied. For multiple DOF motion, equation (9) is a nonlinear process.

Define $\begin{bmatrix} \hat{x}_k & \hat{y}_k & \hat{\Psi}_k \end{bmatrix}^T$ as estimated robot location, then the projected robot location using onboard sensor measurement is

$$\begin{bmatrix} x_k^- \\ y_k^- \\ \Psi_k^- \end{bmatrix} = \begin{bmatrix} cos(\hat{\Psi}_{k-1}) \cdot T & -sin(\hat{\Psi}_{k-1}) \cdot T & 0 \\ sin(\hat{\Psi}_{k-1}) \cdot T & cos(\hat{\Psi}_{k-1}) \cdot T & 0 \\ 0 & 0 & 1 \cdot T \end{bmatrix} \begin{bmatrix} \widehat{u}_{k-1} \\ \widehat{v}_{k-1} \\ \widehat{r}_{k-1} \end{bmatrix} + \begin{bmatrix} \hat{x}_{k-1} \\ \hat{y}_{k-1} \\ \hat{\Psi}_{k-1} \end{bmatrix} \tag{11}$$

where $\begin{bmatrix} x_k^- & y_k^- & \Psi_k^- \end{bmatrix}$ is the location a priori prediction from on-board sensor.

Then the predicted external sensor observation is

$$\begin{bmatrix} z_{1,k}^- \\ z_{2,k}^- \\ z_{3,k}^- \end{bmatrix} = H_k \cdot \begin{bmatrix} x_k^- \\ y_k^- \\ \Psi_k^- \end{bmatrix} = \begin{bmatrix} x_k^- \\ y_k^- \\ \Psi_k^- \end{bmatrix} \tag{12}$$

Define prediction error and observation error as

$$\begin{bmatrix} e_{x_k} \\ e_{y_k} \\ e_{\Psi_k} \end{bmatrix} = \begin{bmatrix} x_k^- \\ y_k^- \\ \Psi_k^- \end{bmatrix} - \begin{bmatrix} x_k \\ y_k \\ \Psi_k \end{bmatrix} \tag{13}$$

$$\begin{bmatrix} e_{z_{1,k}} \\ e_{z_{2,k}} \\ e_{z_{3,k}} \end{bmatrix} = \begin{bmatrix} z_{1,k}^- \\ z_{2,k}^- \\ z_{3,k}^- \end{bmatrix} - \begin{bmatrix} z_{1,k} \\ z_{2,k} \\ z_{3,k} \end{bmatrix} \tag{14}$$

By linearizing equation (11) and along $\begin{bmatrix} x_k & y_k & \Psi_k \end{bmatrix}^T$, the prediction error dynamics are

$$\begin{bmatrix} e_{x_k} \\ e_{y_k} \\ e_{\Psi_k} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin(\overline{\Psi}_{k-1}) \cdot \overline{u}_{k-1} \cdot T - \cos(\overline{\Psi}_{k-1}) \cdot \overline{v}_{k-1} \cdot T \\ 0 & 1 & \cos(\overline{\Psi}_{k-1}) \cdot \overline{u}_{k-1} \cdot T - \sin(\overline{\Psi}_{k-1}) \cdot \overline{v}_{k-1} \cdot T \\ 0 & 0 & 1 \end{bmatrix} \cdot \left(\begin{bmatrix} \hat{x}_{k-1} \\ \hat{y}_{k-1} \\ \hat{\Psi}_{k-1} \end{bmatrix} - \begin{bmatrix} x_{k\text{-}1} \\ y_{k\text{-}1} \\ \Psi_{k\text{-}1} \end{bmatrix} \right) + \begin{bmatrix} cos(\Psi_{k-1}) \cdot T & -sin(\Psi_{k-1}) \cdot T & 0 \\ sin(\Psi_{k-1}) \cdot T & cos(\Psi_{k-1}) \cdot T & 0 \\ 0 & 0 & 1 \cdot T \end{bmatrix} \cdot \begin{bmatrix} w_{1,k-1} \\ w_{2,k-1} \\ w_{3,k-1} \end{bmatrix} \tag{15}$$

The observation error is

$$\begin{bmatrix} e_{z_{1,k}} \\ e_{z_{2,k}} \\ e_{z_{3,k}} \end{bmatrix} = H_k \cdot \begin{bmatrix} e_{x_k} \\ e_{y_k} \\ e_{\Psi_k} \end{bmatrix} + \begin{bmatrix} d_{1,k} \\ d_{2,k} \\ d_{3,k} \end{bmatrix} \tag{16}$$

Equation (15) and (16) are linear. However, in Equation (15), the real position $[x_{k-1}\, y_{k-1}\, \Psi_{k-1}]^T$ are unknown. One option is to employ extended Kalman filter, in which the $[x_{k-1}\, y_{k-1} \Psi_{k-1}]^T$ in Equation (15) is replaced by the filter estimate $\begin{bmatrix} \hat{x}_k & \hat{y}_k & \hat{\Psi}_k \end{bmatrix}^T$. Note that the filter output $\begin{bmatrix} \hat{x}_k & \hat{y}_k & \hat{\Psi}_k \end{bmatrix}^T$ is fed back to the process, which renders the filter process into a nonlinear process. It is well known that such a structure may introduce instability. The convergence of extended Kalman filter has been recently proven for a class of nonlinear systems given a small initial estimation error (Krener, 2003). In (Chenavier, 1992), experimental result were demonstrated for mobile robot location sensor fusion based on EKF.

In mobile robot GNC system, the navigation system is coupled with control system. Therefore, additional concerns have to be taken to guarantee the stability of the coupled system. Kalman filter in Navigation system, in certain degree, can be considered as an observer for the control system. Therefore, observer design theory can be used to analyze the interaction and stability of both systems. In general, the navigation system is required to converge faster than the control system.

Another option is to integrate the navigation system and control system together (Liu, et al. 2007). The structure of integrated approach is similar to linearized Kalman filter, as shown in Figure 3.

In this structure, the nominal system trajectory generated by the control system is used to linearize Equation (15). The nominal trajectory is essentially the filtered position and orientation commands. The filter dynamics are a time-varying linear system instead of a nonlinear system. The integrated control and navigation structure is motivated by TLC observer design (Huang et al. 2003). The stability of the coupled control system and navigation system can be analyzed within the framework of trajectory linearization observer design using linear time-varying system theory. In this structure, the controller drives the robot to follow the nominal trajectory (command). Thus the divergence problem in the linearized Kalman filter is alleviated. The stability of the filter and controller are guaranteed locally around the commanded trajectory. The system stability is guaranteed given a small initial tracking error and a small initial filter observation error. One advantage of such a structure is the computational efficiency.

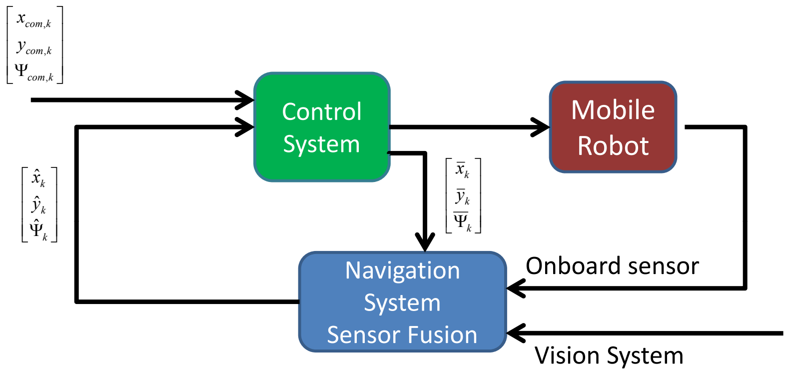


Fig. 3. Integrated navigation and control system for mobile robot

The overall navigation and control system algorithm is illustrated below
Step 1:
Read robot command trajectory $\begin{bmatrix} x_{com,k} & y_{com,k} & \Psi_{com,k} \end{bmatrix}^T$ and calculate the nominal trajectory $\begin{bmatrix} \overline{x}_k & \overline{y}_k & \overline{\Psi}_k \end{bmatrix}^T$
Read onboard sensor measurement and estimate robot position and orientation

$$\begin{bmatrix} x_k^- \\ y_k^- \\ \Psi_k^- \end{bmatrix} = \begin{bmatrix} \hat{x}_{k-1} \\ \hat{y}_{k-1} \\ \hat{\Psi}_{k-1} \end{bmatrix} + \begin{bmatrix} \cos(\hat{\Psi}_{k-1}) \cdot T & -\sin(\hat{\Psi}_{k-1}) \cdot T & 0 \\ \sin(\hat{\Psi}_{k-1}) \cdot T & \cos(\hat{\Psi}_{k-1}) \cdot T & 0 \\ 0 & 0 & 1 \cdot T \end{bmatrix} \begin{bmatrix} \hat{u}_{k-1} \\ \hat{v}_{k-1} \\ \hat{r}_{k-1} \end{bmatrix} \tag{17}$$

Calculate the time-varying matrices

$$A_k = \begin{bmatrix} 1 & 0 & -\sin(\overline{\Psi}_{k-1}) \cdot \overline{u}_{k-1} \cdot T - \cos(\overline{\Psi}_{k-1}) \cdot \overline{v}_{k-1} \cdot T \\ 0 & 1 & \cos(\overline{\Psi}_{k-1}) \cdot \overline{u}_{k-1} \cdot T - \sin(\overline{\Psi}_{k-1}) \cdot \overline{v}_{k-1} \cdot T \\ 0 & 0 & 1 \end{bmatrix}$$

$$W_k = \begin{bmatrix} \cos(\hat{\Psi}_{k-1}) \cdot T & -\sin(\hat{\Psi}_{k-1}) \cdot T & 0 \\ \sin(\hat{\Psi}_{k-1}) \cdot T & \cos(\hat{\Psi}_{k-1}) \cdot T & 0 \\ 0 & 0 & 1 \cdot T \end{bmatrix}$$

Step 2: Calculate the prediction covariance matrix and predicted external sensor observation

$$P_k^- = A_k \cdot P_{k-1} \cdot A_k^T + W_k \cdot Q_{k-1} \cdot W_k^T$$

$$\begin{bmatrix} z_{1,k}^- \\ z_{2,k}^- \\ z_{3,k}^- \end{bmatrix} = \begin{bmatrix} x_k^- \\ y_k^- \\ \Psi_k^- \end{bmatrix} \tag{18}$$

Step 3: Correction with valid external sensor data.
Read sensor measurement $\begin{bmatrix} z_{1,k} & z_{2,k} & z_{3,k} \end{bmatrix}^T$
Calculate correction matrix and update prediction error covariance matrix

$$\begin{aligned} K_k &= P_k^- \left(P_k^- + R\right)^{-1} \\ P_k &= \left(I - K_k\right) P_k^- \end{aligned} \tag{19}$$

where R_k is the external measurement noise covariance. Then a posteriori position estimate is

$$\begin{bmatrix} \hat{x}_k \\ \hat{y}_k \\ \hat{\Psi}_k \end{bmatrix} = \begin{bmatrix} x_k^- \\ y_k^- \\ \Psi_k^- \end{bmatrix} + K_k \left(\begin{bmatrix} z_{1,k} \\ z_{2,k} \\ z_{3,k} \end{bmatrix} - \begin{bmatrix} z_{1,k}^- \\ z_{2,k}^- \\ z_{3,k}^- \end{bmatrix} \right) \tag{20}$$

Goto Step 1.

3.4 Implementation for Mobile Robot Navigation System

(a) Eliminate outlier measurement
The actual external sensor, such as a vision system, may encounter errors in signal processing or communication error. These inevitable errors usually result in measurement with significant large error. These errors are difficult to predict. If these outlier measurements are input into the sensor fusion filter, the filter response has large estimate error, which may cause instability of navigation and control systems. In order to improve the stability of navigation system, gating, is employed to eliminate the outlier vision measurement.
Gating is used in sensor fusion to remove most unlikely observation-to-track pairings (Brown et al. 1996). In mobile robot navigation system, rectangular gate has good performance and is simple to implement. Rectangular gate removes the most unlikely observation.
Rectangular Gating is defined as the following

$$|e_{z_{1,k}}| \leq 3\sqrt{\sigma_{R(i)}^2 + \sigma_{P(i)}^2}, i = 1, 2, 3 \tag{21}$$

where $\sigma_{R(i)}^2$ is the diagonal element of the external sensor noise covariance R, and $\sigma_{P_k(i)}^2$ is the appropriate diagonal element of the prediction covariance P_k^-. If all innovation residues satisfy the above gating inequality, the external sensor data is considered as valid, and will be used in filter correction. Otherwise, the external sensor data is determined as invalid.
(b) Synchronization between external and onboard sensor
In the mobile robot navigation system, the onboard sensor usually has higher bandwidth and faster sampling time than the external positioning sensor. The sensor fusion algorithm described above is executed at the onboard sensor sampling rate. When the external sensor data is not available due to the slow update rate or the sensor data is considered as an outlier, the sensor fusion filter is executed without correction. Step 3 in the filter is reduced as

$$\begin{bmatrix} \hat{x}_k & \hat{y}_k & \hat{\Psi}_k \end{bmatrix}^T = \begin{bmatrix} x_k^- & y_k^- & \Psi_k^- \end{bmatrix}^T \tag{22}$$

$$P_k = P_k^- \tag{23}$$

4. An Example of Mobile Robot Navigation and Control

4.1 Introduction of Omni-directional Mobile Robot and Navigation System

(a) Introduction of Omni-directional mobile robot.
An omni-directional mobile robot is a type of holonomic robot. It has inherent agility which makes it suitable for dynamic environment applications (Purwin 2006). One interesting application is Robocup competition in which mobile robots compete in soccer-like games. In this section, an example of navigation system of omni-directional robot is illustrated. The robot mechanical configuration is shown in Figure 4.
The three omni-directional wheels are driven by electrical motors individually. An Optical encoder are installed on each motor shaft.
(b) Introduction of Navigation System
In the sample system, a roof camera senses the position and the orientation of robots by image processing. The vision system communicates with the robot control system via a serial port. The vision system identified position and orientation are in the camera frame. A second-order polynomials is used to map the camera frame to world frame.

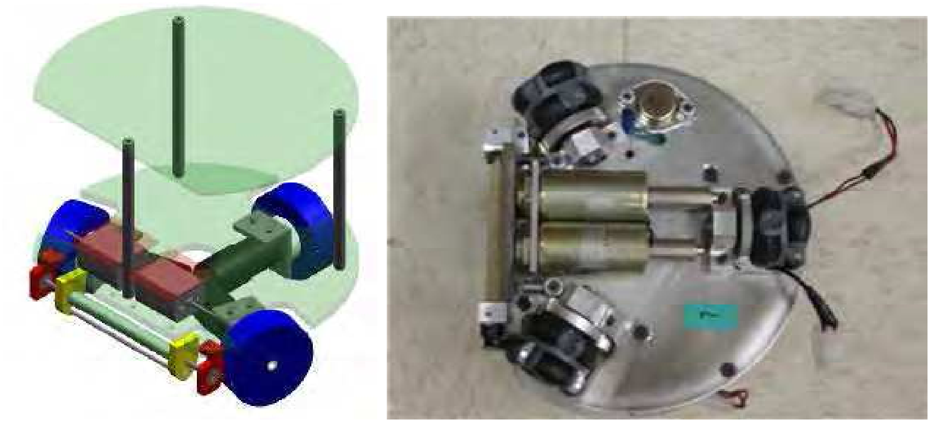


Fig. 4. Omni-directional mobile robot

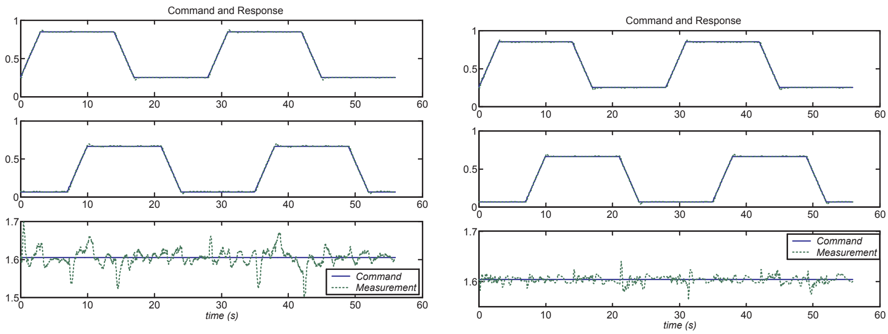
In the illustrated sample system, the robot control system is a dual-loop trajectory linearization control (TLC) system (Liu et al. 2008). The illustrated sample system has the most important features for a general mobile robot navigation and control system. The objective of mobile robot control system is to track a feasible trajectory, which requires an accurate and fast position/orientation measurement. The robot and orientation can be estimated by odometry, which has accumulated error. The overhead camera has accurate but slow measurements. The overhead camera can only capture as high as 60 fps and the period to process each frame varies randomly. The vision system may also lose frames or misidentify objects on the field. The asynchronous serial communication has randomly mismatched data frame, which results in incorrect vision data.

4.2 Omni-directional Mobile Robot Navigation Performance

Omni-directional mobile robot navigation system following sensor fusion is developed using the approach in section 3. In this subsection, several test results are demonstrated. It can be seen that the sensor fusion improves the navigation system accuracy and overall robot tracking peformance.

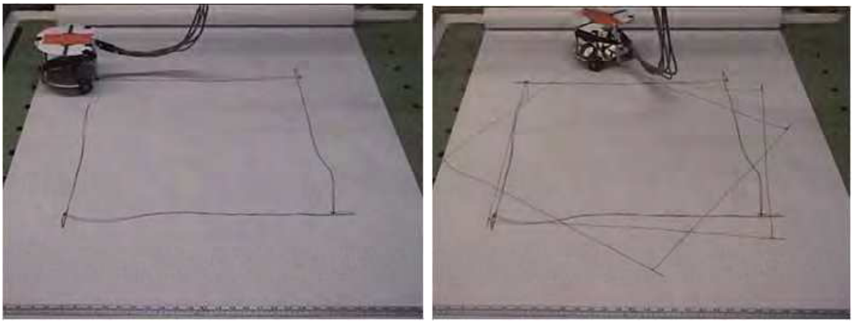
(a) Square Trajectory with Fixed Orientation

In this test, the robot command trajectory is a square curve with a fixed orientation. The actual robot trajectory was plotted on the field of play by an attached pen. The robot trajectory is also plotted by recorded experiment data. Two experiments were performed and compared. One used only the onboard encoder data, the other used navigation system with sensor fusion. Controller tracking performances using both methods are shown in Figure 5 (a) and (b). In both cases, the tracking error is very small relative to the given measurement. The actual robot tracking performance is determined by the navigation system. Figure 6 (a) and (b) are photos of actual robot trajectories drawn by the attached pen. From these two photos, it can be seen that by using sensor fusion, the robot drew a correct square curve with good repeatability in two rounds; while using encoder alone, the robot was totally disoriented with an accumulated error. Figure 7 and 8 show the Kalman filter performance and gating decision. Figure 7 shows that vision system generated many outlier measurements, while the sensor fusion with gating is able reject the wrong vision data and provide stable and accurate reading. In Figure 8, 1 means acceptance of the vision data, 0 means rejection of the vision data. Experiment using vision system alone was also conducted. The disturbance induced by vision system failure destabilized the robot very soon.



(a) Using Sensor Fusion (b) Using Onboard Sensor

Fig. 5. Square Trajectory Tracking Performance using Sensor Fusion



(a) Using Sensor Fusion (b) Using Onboard Sensor

Fig. 6. Trajectory for Square Trajectory Command

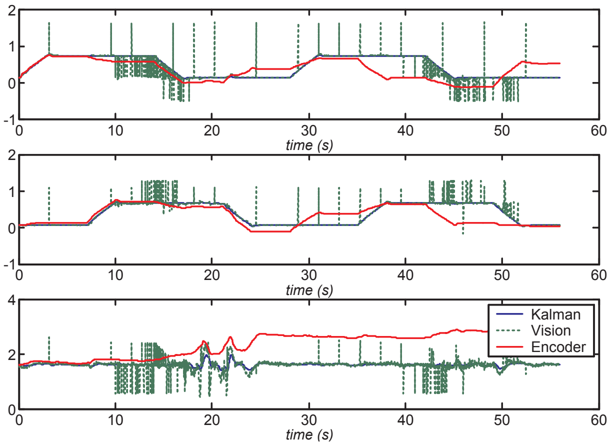


Fig. 7. Square Trajectory Sensor Fusion Kalman Filter Performance

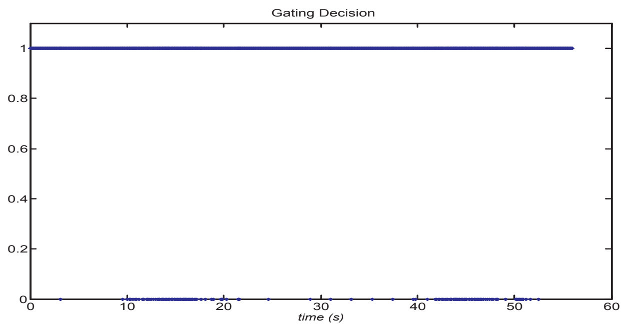


Fig. 8. Square Trajectory Sensor Fusion Gating Decision

(b) Circular trajectory with rotation
In this test, the robot was commanded to accelerate from the initial position, and draw a circle of 0.25 m radius at an angular rate of 1 rad/s. The robot orientation is commanded to change between ±1 rad. In this test, the robot nonlinear dynamics are stimulated. In Figure 9, the robot controller showed accurate tracking performance. Figure 10 illustrates the navigation system performance. It should be noted that the navigation system performance is similar to test (a) though the robot's motion is nonlinear due to the rotation.

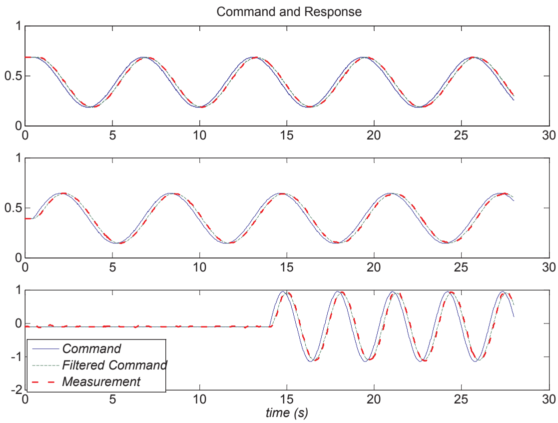


Fig. 9. Circular Trajectory Tracking Performance using Sensor Fusion

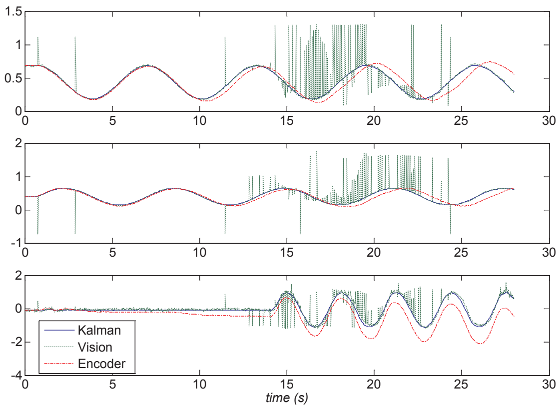
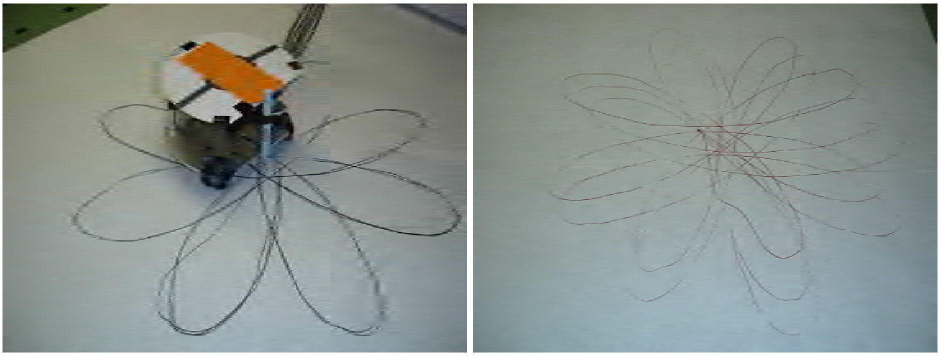


Fig. 10. Circular Trajectory Sensor Fusion Performance

(c) Rose Curve
In this test, the robot is commanded to draw a rose curve generated by a function: $r = a\sin(n\theta)$, where r and θ are the radius and the rotation angle in polar coordinate, and n is

an integer determining the number of petals. The robot orientation was fixed. Figure 11 is the result for $n = 4$. Figure 11 (a) and (b) are pictures of robot trajectory using sensor fusion and onboard encoder alone respectively. The robot trajectory shifted significantly when using the onboard sensor alone. As the result, the trajectory plotted when using encoder only is much lighter than the one using sensor fusion. Figure 12 is the recorded data for test with sensor fusion navigation. From this figure, it is clear without vision system correction, the onboard encoder reading slowly drifts away.



(a) Using Sensor Fusion (b) Using Onboard Sensor

Fig. 11. Actual Robot Trajectory for Rose Curve

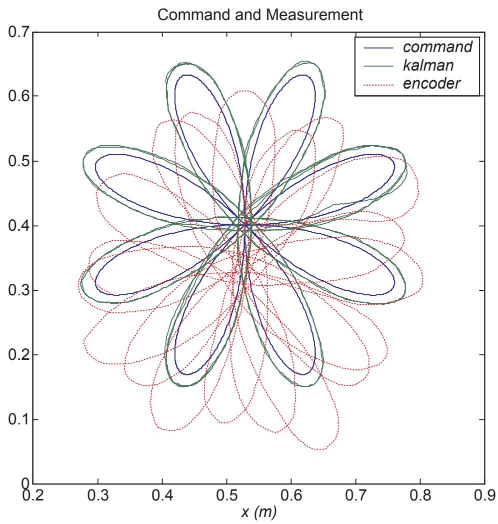


Fig. 12. Robot Tracking Performance Using Sensor Fusion

5. References

Arulampalam, S.; Maskell, S. ; Gordon, N. J.; Clapp, T. (2001). A Tutorial on Particle Filters for On-line Non-linear/Non-Gaussian Bayesian Tracking, *IEEE Transactions of Signal Processing*, Vol. 50, No.2, pp. (174-188), 2001, ISSN 1053-587X.

Brown, R. G.; Hwang, P. Y.C. (1996). *Introduction to random signals and applied Kalman filtering : with MATLAB exercises and solutions*, 3rd ed, ISBN ISBN: 978-0-471-12839-7, Wiley, New York.

Borenstein, J., Everett H. R.; Feng, L. (1996). *Sensors and Methods for Mobile Robot Positioning, The University of Michgan*, 1996.

Chenavier, F.; Crowley, J. (1992), Position Estimation for a Mobile Robot Using Vision and Odometry, *1992 IEEE International Conference on Robotics and Automation*, pp. 2588-2593, Nice, France, 1992.

Goel, P.; Roumeliotis, S. I.; Sukhatme, G. S. (1999). Robust localization using relative and absolute position estimates, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (1999)*, pp. 1134-1140, ISBN: 0-7803-5184-3, Kyongju, South Korea.

Hall, D. L.; McMullen , S. A. H. (2004). *Mathematical Techniques in MultiSensor Data Fusion*, 2nd ed., ISBN 1580533353, Artech House, Boston, MA.

Huang, R.; Mickle, M. C. and Zhu, J. J. (2003), Nonlinear Time-varying Observer Design Using Trajectory Linearization, *Proceedings of the 2003 American Control Conference*. v 6, , p 4772-4778, ISBN: 0-7803-7896-2, Denver, CO, 2003.

Kalman, R. E. (1960), *Transactions of the ASME- Journal of Basic Engineering*, 82(series D): 35-45, 1960.

Krener, A.J. (2003), The convergence of the extended Kalman filter, *Directions in mathematical systems theory and optimization, (Lect. Notes in Control and Information Sciences).*, pp. 173-182, Springer, ISBN: 3540000658,Berlin, 2003.

Liu, Y.; Williams II, R. L.;Zhu, J. J. (2007). Integrated Control and Navigation for Omni-directional Mobile Robot Based on Trajectory Linearization, *Proceedings of the 2007 American Control Conference*, pp. 3053-3058, New York, NY, July 2007, ISSN:0921-8890.

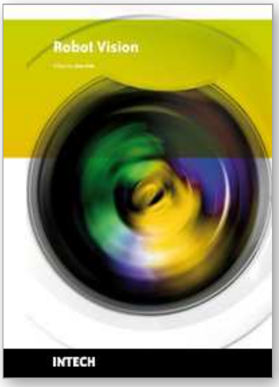
Liu, Y.; Williams II, R.; Zhu J. J.; Wu, J. (2008). Omni-Directional Mobile Robot Controller Based on Trajectory Linearization, *Robotics and Autonomous Systems*, vol. 56, no. 5, pp. 461-479, DOI :10.1016/j.robot.2007.08.007 , 2008, ISSN: 0921-8890.

Purwin, O; D'Andrea, R (2006). Trajectory Generation and Control for Four Wheeled Omnidirectional vehicles, *Robotics and Automation Systems*, vol. 54, pp. 13-22, ISSN: 0921-8890.

Pin, F.G.; Killough, S. M; (1994). A new family of omnidirectional and holonomic wheeled platforms for mobile robots, *IEEE Trans. Robotics Automat.*, vol. 10, no. 2, 1994, pp.480-489, ISSN: 1042-296X.

Richard, J.; Sigpurwalla, N. D.(1983). Understanding the Kalman Filter, *The American Statistician*, May 1983, pp. ,Vol. 37, No. 2.

Welch, G.; Bishop, G.(2001), *An Introduction to Kalman Filter*, ACM, INC, 2001



Robot Vision

Edited by Ales Ude

ISBN 978-953-307-077-3

Hard cover, 614 pages

Publisher InTech

Published online 01, March, 2010

Published in print edition March, 2010

The purpose of robot vision is to enable robots to perceive the external world in order to perform a large range of tasks such as navigation, visual servoing for object tracking and manipulation, object recognition and categorization, surveillance, and higher-level decision-making. Among different perceptual modalities, vision is arguably the most important one. It is therefore an essential building block of a cognitive robot. This book presents a snapshot of the wide variety of work in robot vision that is currently going on in different parts of the world.