Stony Brook University

Design and Analysis of Heterogeneous Sensors based Object Tracking Systems

A Dissertation Presented

by

Jinseok Lee

to

The Graduate School

in Partial Fulfillment of the

Requirements

for the Degree of

Doctor of Philosophy

in

Electrical Engineering

Stony Brook University

May 2009

Stony Brook University

The Graduate School

Jinseok Lee

We, the dissertation committee for the above candidate for the

Doctor of Philosophy degree,

hereby recommend acceptance of this dissertation.

Sangjin Hong, Advisor of Dissertation

Associate Professor, Department of Electrical and Computer Engineering

John Murray, Chairperson of Defense

Associate Professor, Department of Electrical and Computer Engineering

Monica Fernandez-Bugallo, Assistant Professor

Department of Electrical and Computer Engineering

Jeffrey Ge, Professor

Department of Mechanical Engineering

This dissertation is accepted by the Graduate School

Lawrence Martin

Dean of the Graduate School

Abstract of the Dissertation

Design and Analysis of Heterogeneous Sensors based Object Tracking Systems

by

Jinseok Lee

Doctor of Philosophy

in

Electrical Engineering

Stony Brook University

2009

In a surveillance system for monitoring objects, there is an increasing need for developing robust algorithm as well as dealing with intelligent interaction among different types of sensors and information. This dissertation describes a design and an analysis of object tracking methodology in heterogeneous sensor network.

Recently, Sequential Monte Carlo (SMC) techniques, or particle filters have received a great deal of attention due to desirable characteristics such as the ability to achieve multiple hypotheses and the relaxation of the Gaussian and linear modeling assumption. Such characteristics allow SMC techniques to be applied to a variety of areas including audio processing, RFID detection processing and computer vision. In this dissertation, the investigation of SMC technique is primarily formulated and

analyzed based on an acoustic sensor. Once the performance is analyzed, we address an inherent limitation of the technique with the single type of sensor, an acoustic sensor. In order to overcome the limitation, we propose a multiple types of sensors based cooperation method. In addition, since a visual sensor requires much higher computational resources than an acoustic sensor, we focus on a sensor cooperation method for minimized overall computation while the limitation is significantly alleviated. Note that the cooperation method assumes that visual sensors detect an object. In practice, however, visual sensors do not necessarily detect an object due to object overlapping, obstacle occlusion and background noise. Thus, we finally present the method for increasing an object detection through information collaboration.

In the first part, we consider the object tracking problem in three dimensional (3-D) space when the azimuth and the elevation of the object are available from the passive acoustic sensor. The particle filtering technique can be directly applied to estimate the 3-D location of the object, but we propose to decompose the 3-D particle filter into the three planes' particle filters which are individually designed for the 2-D bearings-only tracking problems. The 2-D bearing information is derived from the azimuth and the elevation of the object to be used for the 2-D particle filter. Two estimates of three planes' particle filters are selected based on the characterization of the acoustic sensor operation in noisy environment. The proposed approach is extended to multiple acoustic sensors and its robustness is analyzed. The Cramer-Rao Lower Bound of the proposed 2-D particle filter-based algorithm is derived and compared against the algorithm based on the direct 3-D particle filter.

In the second part, the object tracking by a single acoustic sensor based on the

particle filtering is extended for the multiple objects, and the corresponding inherent limitation is introduced. In order to overcome the limitation of the acoustic sensor for the simultaneous multiple object tracking, the support from the visual sensor with the objects' localization is considered. The cooperation from the visual sensor, however, should be minimized, as the visual sensor's object localization requires much higher computational resources than the acoustic sensor based estimation, and the visual sensor is usually not dedicated to the object tracking and deployed for other applications. The acoustic sensor mainly tracks multiple objects and the visual sensor supports the tracking task only when the acoustic sensor has a difficulty. Several techniques of the particle filtering are used for the multiple object tracking by the acoustic sensor and the limitations of the acoustic sensor are discussed to identify the need of the visual sensor cooperation. Performance of the triggering-based cooperation of the two visual sensors is evaluated and it is compared with a periodic cooperation in a real environment.

In the third part, we address enhancement of object detection with multiple visual sensors. The detection enhancement we introduce is to recover missed object detection given partially detected objects among multiple visual sensors. Once an object is detected by one or more visual sensors, the detected local object positions are transformed into a global object position. Based on a local and global information collaboration, any missed local object position is recovered by the global to local transformation. However, the collaboration may degrade the detection performance by incorrectly recovering the local object position, which is propagated from false object detection. Furthermore, local object positions corresponding to an identical

object are transformed into inequivalent global object positions due to detection uncertainty such as a shadow. We minimize the performance degradation by preventing from the propagation of the false object detection. In addition, we present an evaluation method for a final global object position. Finally, the proposed method is analyzed and evaluated with case studies.

In the last part, we summarize and highlight our proposed object tracking methodology in heterogeneous sensor network. In addition, ongoing and future research is presented. The future research includes face identification, robot navigation and other sensors combination based cooperation method. In the face identification issue, we study temporal and spatial face characteristics. In the robot navigation issue, we identify a limitation of the existing method, potential field method, and present a possible solutions. In the other sensors combination based cooperation issue, Radio Frequency Identification (RFID) and visual sensor combination is considered with data traffic analysis.

To my Family

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Tracking is generally defined as successive estimation of any unknown variable that is continuously evolving in the physical world [58]. The estimation process in tracking usually involves two indispensable components: the physical measurements provided by available sensors and the knowledge about the dynamics. Without sensor inputs, the unknown variable can only be guessed, or predicted, but can never be verified by physical evidence. Without information about dynamics, the unknown variable can only be derived from measurements. The measurements vary from each type of a sensor. In an acoustic tracking, the measurement is angle, intensity or time delay [1] [3] [10]. In a RFID tracking, the measurement is an identification number by a detecting RFID reader [72] [73]. Furthermore, in a visual tracking, the measurement is motion, color, feature or/and edge [59] [60] [61] [62].

Sequential Monte Carlo (SMC) methods are a set of flexible simulation-based methods for sampling from a sequence of probability distributions. These methods were originally introduced in the early 50's by physicists and have become very pop-

ular over the past few years in statistics and related fields. Hence, they are now extensively used to solve sequential Bayesian inference problems arising in signal processing, robotics and networks [5] [6] [7].

The SMC methods approximate the sequence of probability distributions of interest using a large set of random samples which is called particles. These particles are propagated over time using simple Importance Sampling (IS) and resampling mechanisms [17] [18] [19]. Asymptotically, i.e. as the number of particles goes to infinity, the convergence of these particle approximations towards the sequence of probability distributions can be ensured under very weak assumptions. However, for practical implementations, a finite and sometimes quite restricted number of particles has to be considered. Much research is therefore devoted to the design of efficient sampling strategies in order to sample particles in regions of high probability mass. Throughout this dissertation, an acoustic sensor based SMC technique is applied and evaluated with inherent limitation. Then, we use visual sensors for alleviating the limitation. In this chapter, we briefly explain about the background knowledge of SMC techniques as well as general tracking problem definition, and present general visual sensor based object tracking.

1.1 Tracking Problem Definition

To define the problem of tracking, consider the evolution of the state sequence $\mathbf{X}_n$

$$\mathbf{X}_n = f_{n-1}(\mathbf{X}_{n-1}) + \mathbf{Q}_{n-1}, \tag{1.1}$$

where f_n is a nonlinear, state transition function of the state $\mathbf{X}_n$, and $\mathbf{Q}_{n-1}$ is the non-Gaussian, process noise in the interval time-instant between n and $n-1$. The measurements of the evolving target state vector is expressed as

$$\mathbf{Z}_n \quad = \quad h_n(\mathbf{X}_n) + \mathbf{E}_n, \tag{1.2}$$

where h_n is a nonlinear and time-varying function of the target state, $\mathbf{E}_n$ is the measurement error which is independent identically distributed white noise process.

In order to estimate target state vector, dynamic prior probability density function (pdf) [8] is obtained as

$$p(\mathbf{X}_n|\mathbf{Z}_{1:n-1}) = \int p(\mathbf{X}_n|\mathbf{X}_{n-1})p(\mathbf{X}_{n-1}|\mathbf{Z}_{1:n-1})d\mathbf{X}_{n-1}, \tag{1.3}$$

where $\mathbf{Z}_{1:n}$ represents the sequence of measurements up to time instant n, and $p(\mathbf{X}_n|\mathbf{X}_{n-1})$ is the state transition density with Markov process of order one related to $f_n(\cdot)$ and $\mathbf{Q}_{n-1}$ in (1.1).

For the next time estimation based on Bayes' rule, posterior pdf involving prediction pdf is obtained [8] as

$$p(\mathbf{X}_n|\mathbf{Z}_{1:n}) = \frac{p(\mathbf{Z}_n|\mathbf{X}_n)p(\mathbf{X}_n|\mathbf{Z}_{1:n-1})}{\int p(\mathbf{Z}_n|\mathbf{X}_n)p(\mathbf{X}_n|\mathbf{Z}_{1:n-1})d\mathbf{X}_n}, \tag{1.4}$$

where $p(\mathbf{Z}_n|\mathbf{X}_n)$ is a likelihood function and the denominator is the normalizing constant.

The recursive propagation in (1.3) and (1.4) are only conceptual solution in the

sense that generally they cannot be determined analytically [19]. In other words, the implementation of the conceptual solution requires the storage of the entire pdf which is equivalent to an infinite dimensional vector. Since the analytic solution of in most practical situations (1.3) and (1.4) are intractable, suboptimal Bayesian algorithms approximate the solution.

1.2 Motivation of Particle Filter

Optimal finite-dimensional algorithms for recursive Bayesian state estimation are differently formulated according to assumptions. In a linear-Gaussian case, the functional recursion of (1.3) and (1.4) becomes the Kalman Filter. In addition, if the state space is discrete-valued with a finite number of states, the grid-based methods provide the optimal algorithm. [19]

However, the assumption of linear and Gaussian system are too strict and do not hold for most real world problems. As the system is non-linear or non-Gaussian, the closed form expressions are almost impossible to obtain. Hence, the approximation technique is required for real world system. In the approximation methods, the extended Kalman filter (EKF), the unscented Kalman filter (UKF) have been applied in a deterministic approach. However, the deterministic approaches turn out to be ineffective or too computationally demanding [7] [8]. Consequently, methods based on particle filter, which is simulation-based technology, has been widely applied in many tracking applications [16] [19].

1.3 Particle Filter

The sequential Monte Carlo (SMC) methods, referred to as particle filters, have received a lot of attention because they are particularly suitable for real-time estimation. Particle filters provide sequential procedures that use information from the previous time instant and current measurement to update the posterior distribution.

Sequential importance sampling (SIS) is widely applied to perform nonlinear filtering which is introduced in (1.3) and (1.4). The SIS algorithm determines the required posterior pdf by a set of random samples with associated weights [8]. The SIS algorithm forms the basis for most of the proposed particle filters such as sampling importance resampling particle filter (SIR-PF), auxiliary sampling importance resampling particle filter (ASIR-PF) and regularized particle filter (RPF). Under the assumption of which nonlinear functions f_{n-1} and h_n in (1.3) and (1.4) are known, SIR algorithm is derived by choosing the candidates of estimation referred as importance density as well as by performing the resampling step at every time-instant. The resampling step is required for eliminating the degeneracy problem in the particle filtering (i.e., the samples with small weight are becoming smaller as the estimation is recursively iterated) [17] [18]. The SIR filter method has the advantage that the importance weights are easily evaluated and the importance density can be easily sampled. Throughout the thesis, we apply SIR particle filter, especially in acoustic tracking.

1.4 Visual Sensor based Object Detection and Tracking

In a real-world problem, particle filter algorithms heavily depends on how accurate the underlying probabilistic model matches the real dynamic system [12] [40]. The model selection is typically based on physics, aimed at matching real measurements to standard analytical distributions. In addition, it is affected by mathematical tractability, computational complexity or heuristic considerations. Furthermore, based on an acoustic sensor, sound waves may be transmitted sporadically due to blocks. That is the reason why we collaborate multiple types of sensors, an acoustic sensor and a visual sensor.

Based on a visual sensor, the available measurement becomes more reliable and accurate. Figure 1-1 illustrates the basic surveillance system with motion detection, body tracking, face detection/tracking and localization modules [26] [45]. In addition to the modules, database is incorporated to the system for robust surveillance. Furthermore, the association method based on homographic lines and local-to-global information transformation is incorporated into the system even though the module and interaction is not explicit in Figure 1-1.

Figure 1-2 shows an example of object detection and tracking. In the example, body and head are detected and tracked with bounding boxes.

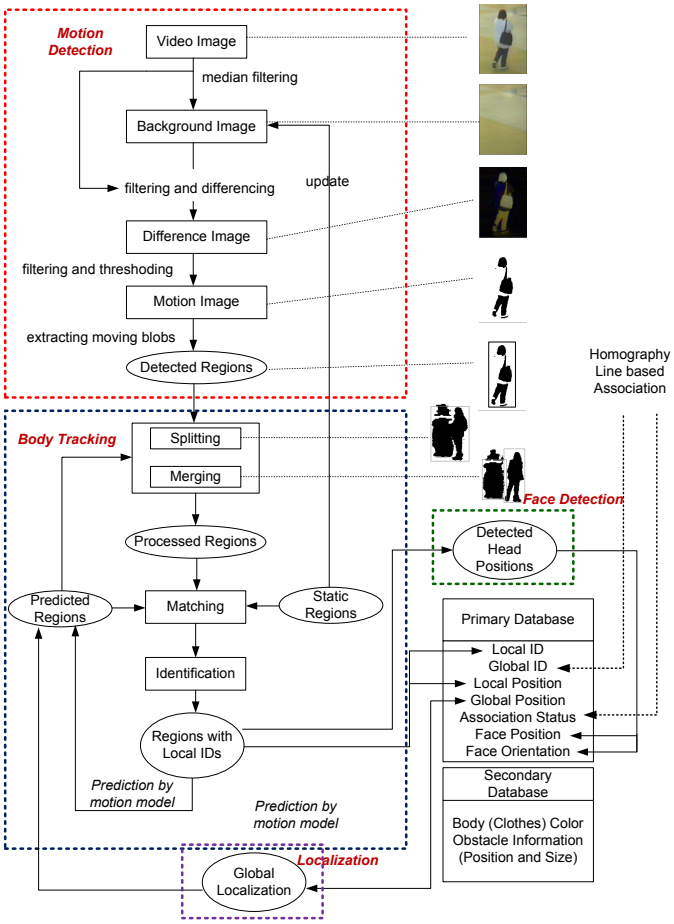


Figure 1-1: Simplified surveillance system with detection, tracking, face detection/tracking and localization modules.

(a) Frame #8 (b) Frame #9 (c) Frame #10

Figure 1-2: Two persons are moving apart and a bounding box for each human is drawn.

1.5 Contribution and Overview

This thesis is mainly concerned with solving problems in object tracking based on acoustic and visual sensors with the frame of particle filter.

Specifically, this thesis has following contributions:

- (Chapter 2) Acoustic sensor based 3 dimensional particle filtering formulation in object tracking methodology
- (Chapter 3) Visual sensor and acoustic sensor cooperation method for alleviating the limitation of the only acoustic sensor based tracking as well as minimizing overall computational resources
- (Chapter 4) Object Detection Enhancement using multiple visual sensors

Chapter 2

Object Tracking in 3-D Space with Passive Acoustic Sensors using Particle Filter

2.1 Introduction

Locating and tracking an object using passive sensors both indoor and outdoor have been great interests in numerous applications. For tracking an object with passive sensors, several approaches based on time-delay estimation (TDE) methods and beamforming methods have been proposed. The TDE method estimates location based on the time delay of arrival of signals at the receivers [1]. The beamforming method uses the frequency-averaged output power of a steered beamformer [2] [3]. The TDE method and beamforming method determine the current source location using the data obtained only at the current time. Each method transforms the acoustic data to

a spatial data so that the peak represents the source location in a deterministic way.

The estimation accuracy of these methods, however, is sensitive to the noise-corrupted signals. In order to overcome the drawback of these methods, a state-space driven approach based on particle filtering was proposed [4] [5]. The particle filtering is an emerging powerful tool for sequential signal processing, especially for nonlinear and non-Gaussian problems [6] [7] [8]. Tracking with particle filters for the source localization is formulated in [9], where the TDE and beamforming methods are revised for the new framework. In [9], sensors are positioned at specified locations with a constant height to estimate an object's trajectory in two dimensional (2-D) space. The extension to 3-D space from the revised TDE and beamforming methods is difficult and a large number of microphones are required to generate a new 2-D plane for the 3-D extension. In addition, mobility of the sensors cannot be supported due to their fixed positions. In order to overcome the mobility problem, Direction of Arrival (DOA) based bearings-only tracking has been widely used in many applications [10] [11] [12].

In this chapter, we analyze the tracking methods based on passive sensors for flexible and accurate 3-D tracking. Tracking in 3-D has been addressed by directly extending 2-D bearings-only tracking problem to 3-D problem [13] [14]. Instead of directly extending traditional particle filtering algorithms for bearings-only tracking in a 3-D space, we propose to decompose the 3-D particle filter into several simpler particle filters designed for 2-D bearings-only tracking problems. The decomposition and planes selection are based on the characterization of the acoustic sensor operation under noisy environment. We use the passive acoustic localizer model in [15], where

the two angle components (azimuth angle θ and elevation angle ϕ) between a sensor and an object are detected by the localizer. We also extend the proposed approach to multiple particle filter fusion for a robust performance. We compare the proposed approach with the directly extended bearings-only tracking method using Cramer-Rao Lower Bound.

2.1.1 Background and Problem Description

2.1.2 Noisy Measurement Characterization on Projected Planes

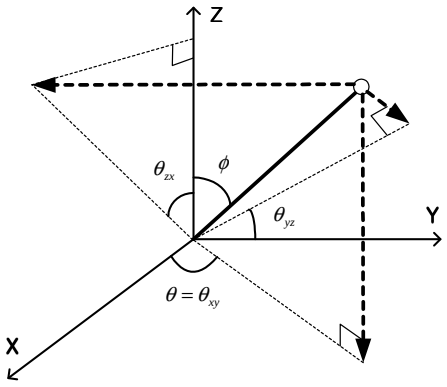


Figure 2-1: Conversion of the originally measured angles θ and ϕ to the projected angles θ_{xy}, θ_{yz} and θ_{zx}.

3-D localizer model and its implementation are described in [15], and it is based on the gradient flow to determine the DOA of the acoustic source. Figure 2-1 illustrates the simplified angle conversion process. Based on the two measured angles, azimuth θ and elevation ϕ, ($0 \leq \theta < 2\pi$, $0 \leq \phi < \pi$), three projected angles onto two dimensional (2-D) planes are derived; θ_{xy}, θ_{yz} and θ_{zx}. Each of these three angles can be used for a 2-D tracking using particle filter [16]. For example, θ_{xy} is used in xy-plane, θ_{yz} and

θ_{zx} are used in yz-plane and zx-plane, respectively. The projected angles are derived and defined as

$$\theta_{xy} = \theta, \quad \theta_{yz} = \arctan\left(\frac{|\sec\theta|}{\tan\theta\tan\phi}\right) + \beta, \quad \theta_{zx} = \arctan\left(\frac{\tan\phi}{|\sec\theta|}\right) + \gamma, \tag{2.1}$$

where

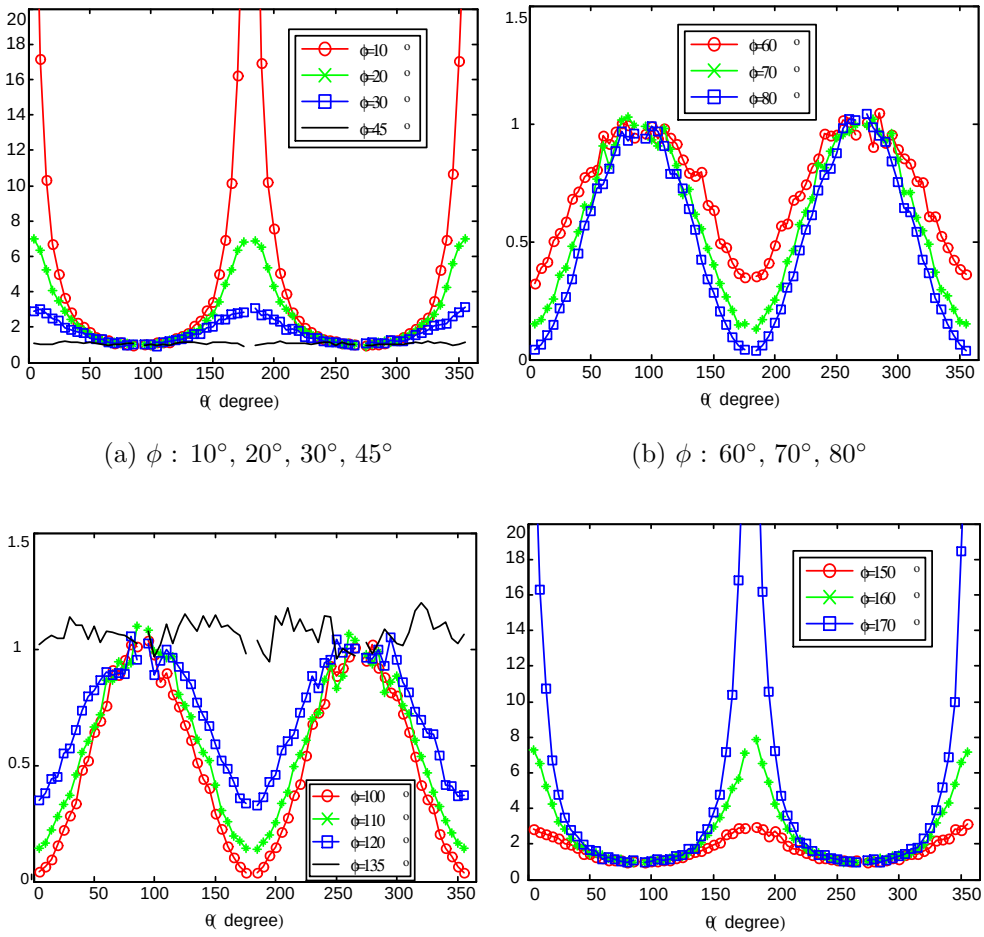
$$\beta = \begin{cases} 0, & \text{for } y \geq 0, z \geq 0 \\ \pi, & \text{for } y < 0, \\ 2\pi, & \text{for } y \geq 0, z < 0, \end{cases} \quad \gamma = \begin{cases} 0, & \text{for } z \geq 0, x \geq 0 \\ \pi, & \text{for } x < 0, \\ 2\pi, & \text{for } z \geq 0, x < 0, \end{cases} \quad \text{and } \sec\theta = \frac{1}{\cos\theta}. \tag{2.2}$$

We assume that each of measurement error of original angles θ and ϕ is an independent and identically distributed random sequence respectively and the two random sequences are independent. Also, we assume that the measurement errors are zero-mean with the same variance of σ^2. Then, the noisy measurements of θ and ϕ with the same error variance of σ^2 are reflected to the projected plane angles θ_{xy}, θ_{yz} and θ_{zx} with their own variances σ^2_{xy}, σ^2_{yz}, and σ^2_{zx}. Define the projected plane angles as

$$\theta_{xy,n} = \bar{\theta}_{xy,n} + e^{xy}_n, \quad \theta_{yz,n} = \bar{\theta}_{yz,n} + e^{yz}_n, \quad \theta_{zx,n} = \bar{\theta}_{zx,n} + e^{zx}_n, \tag{2.3}$$

where $\bar{\theta}_{\mathtt{P},n}$ is the projected true angle, $e^{\mathtt{P}}_n$ is the angle error with the variance $\sigma^2_{\mathtt{P}}$ in $\mathtt{P}$-plane at time instant n respectively, and $\mathtt{P} \in \{xy, yz, zx\}$. Note that the original measurement error variance, σ^2, is differently projected to σ^2_{xy}, σ^2_{yz} and σ^2_{zx}.

Projected angles from the original measurements θ and ϕ are derived in (2.1),



(a) ϕ : 10°, 20°, 30°, 45°

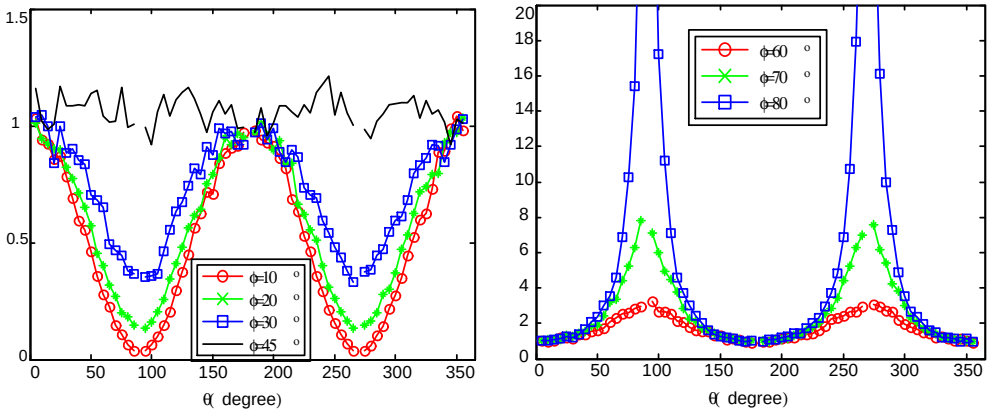
(b) ϕ : 60°, 70°, 80°

(c) ϕ : 100°, 110°, 120°, 135°

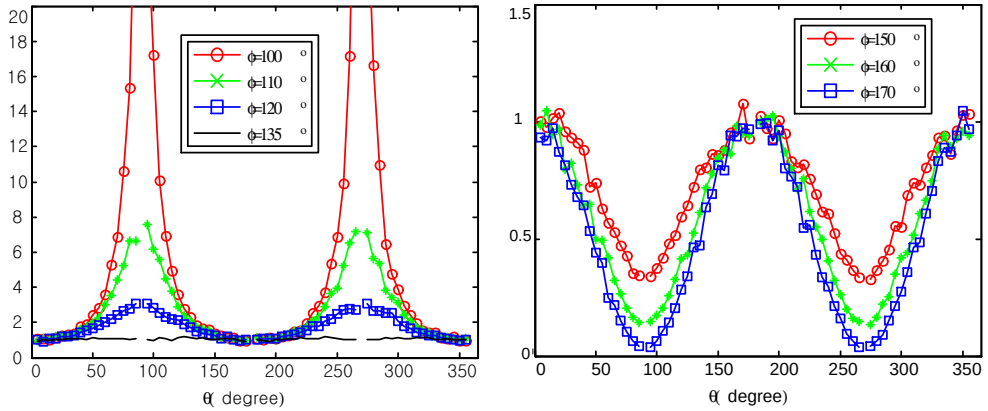
(d) ϕ : 150°, 160°, 170°

Figure 2-2: Angle variances σ_{yz} in a projected yz-plane according to θ and ϕ. The originally measured angle variances are 1. (x-axis: angle θ (degree), y-axis: variance)

but it is difficult to derive the closed-form expression for their variances from the variances of the original measurement errors – it requires the variance of products and the variance of nonlinear functions. Results from the Monte-Carlo simulation in Figure 2-2 and Figure 2-3 show the projected angles' variances when the original measurements' variances are the same by one. Note that the projected measurement in xy-plane, θ_{xy} is the same as the original θ; thus, σ^2_{xy} is the same as σ^2. The projected variances in yz- and zx-planes are functions of θ and ϕ. In yz-plane, the elevation



(a) ϕ : 10°, 20°, 30°, 45° (b) ϕ : 60°, 70°, 80°



(c) ϕ : 100°, 110°, 120°, 135° (d) ϕ : 150°, 160°, 170°

Figure 2-3: Angle variances σ_{zx} in a projected z-x plane according to θ and ϕ. The originally measured angle variances are 1 (x-axis: angle θ (degree), y-axis: variance).

angles ϕ between 45° and 135° are projected with a smaller variance than the original measurement variance of one. In addition, as the azimuth angle θ approaches to 0° or 180°, the variance decreases further. For zx-plane, the other ranges of ϕ and θ are projected with a smaller variance than the original measurements' variance.

2.1.3 Problem Formulation for 3-D Space Estimation

Consider an object's state vector $\mathbf{X}_n$, with discrete time instant $n \in \{1, 2, \cdots\}$, evolving according to

$$\mathbf{X}_n = f_{n-1}(\mathbf{X}_{n-1}) + \mathbf{Q}_{n-1}, \tag{2.4}$$

where f_{n-1} is a nonlinear dynamic transition function on state vector $\mathbf{X}_{n-1}$ and $\mathbf{Q}_{n-1}$ is a noise process (not-necessarily Gaussian) sampled at time instant $n-1$. The measurements of the object state vector is expressed as

$$\mathbf{Z}_n = h_n(\mathbf{X}_n) + \mathbf{E}_n, \tag{2.5}$$

where h_n is a nonlinear and time-varying observation function of state vector $\mathbf{X}_n$ and $\mathbf{E}_n$ is the measurement error referred to as a measurement noise sequence which is independent identically distributed (IID) noise process. Then, the prediction probability density function (pdf) is obtained as

$$p(\mathbf{X}_n|\mathbf{Z}_{1:n-1}) = \int p(\mathbf{X}_n|\mathbf{X}_{n-1})\ p(\mathbf{X}_{n-1}|\mathbf{Z}_{1:n-1})\ \mathrm{d}\mathbf{X}_{n-1}, \tag{2.6}$$

where $\mathbf{Z}_{1:n}$ represents the sequence of measurements up to time instant n, and $p(\mathbf{X}_n|\mathbf{X}_{n-1})$ is the state transition density with Markov process of order one related to $f_n(\cdot)$ and $\mathbf{Q}_{n-1}$ in (5.35) [19]. Note that $p(\mathbf{X}_{n-1}|\mathbf{Z}_{1:n-1})$ is recursively obtained from previous time instants.

From the Bayes' rule, the estimation at the next time instant can be done as

follow. The posterior pdf is obtained using the prediction pdf as

$$p(\mathbf{X}_n|\mathbf{Z}_{1:n}) = \frac{p(\mathbf{Z}_n|\mathbf{X}_n)\ p(\mathbf{X}_n|\mathbf{Z}_{1:n-1})}{\int p(\mathbf{Z}_n|\mathbf{X}_n)\ p(\mathbf{X}_n|\mathbf{Z}_{1:n-1})\ \mathrm{d}\mathbf{X}_n}, \tag{2.7}$$

where $p(\mathbf{Z}_n|\mathbf{X}_n)$ is the likelihood or measurement density in (5.36) related to the measurement model $h_n(\cdot)$ and the noise process $\mathbf{E}_n$, and the denominator is the normalizing constant. Note that the measurement $\mathbf{Z}_n$ is used to modify the prior density in (5.37) to obtain the current posterior density in (5.38) [19].

In this chapter, $\theta_{xy,n}$ and $\mathbf{Z}_n(xy)$ are interchangeably used as the projected angle measurement in xy-plane. Similarly, $\theta_{yz,n}$, $\mathbf{Z}_n(yz)$, $\theta_{zx,n}$, $\mathbf{Z}_n(zx)$ are for yz-plane and zx-plane, respectively. State vectors of an object in 3-D space ($\mathbf{X}_n$) and in 2-D planes, ($\mathbf{X}_n(xy)$, $\mathbf{X}_n(yz)$, $\mathbf{X}_n(zx)$) are defined as

$$\mathbf{X}_n = \begin{pmatrix} x_n \\ V_n^x \\ y_n \\ V_n^y \\ z_n \\ V_n^z \end{pmatrix}, \qquad \mathbf{X}_n(xy) = \begin{pmatrix} x_n(xy) \\ V_n^x(xy) \\ y_n(xy) \\ V_n^y(xy) \end{pmatrix},$$

$$\mathbf{X}_n(yz) = \begin{pmatrix} y_n(yz) \\ V_n^y(yz) \\ z_n(yz) \\ V_n^z(yz) \end{pmatrix}, \qquad \mathbf{X}_n(zx) = \begin{pmatrix} z_n(zx) \\ V_n^z(zx) \\ x_n(zx) \\ V_n^x(zx) \end{pmatrix}, \tag{2.8}$$

where $\{x_n, y_n, z_n\}$ and $\{V_n^x, V_n^y, V_n^z\}$ are the true source location and the velocity in 3-D Cartesian coordinates at time instant n. $\{x_n(xy), y_n(xy)\}$ and $\{V_n^x(xy), V_n^y(xy)\}$ are the projected true source location and velocity on xy-plane at time instant n; the same notation is applied for yz- and zx-planes. Note that $x_n(xy)$ and $x_n(zx)$ are estimated separately and x_n is the finally fused value based on $x_n(xy)$ and $x_n(zx)$; the rest of components are applied by the same way. The three posterior pdf involving prediction probability density functions are given as

$$p(\mathbf{X}_n(xy)|\mathbf{Z}_{1:n}(xy)) = \frac{p(\mathbf{Z}_n(xy)|\mathbf{X}_n(xy))\ p(\mathbf{X}_n(xy)|\mathbf{Z}_{1:n-1}(xy))}{\int p(\mathbf{Z}_n(xy)|\mathbf{X}_n(xy))\ p(\mathbf{X}_n(xy)|\mathbf{Z}_{1:n-1}(xy))\ \mathrm{d}\mathbf{X}_n(xy)}, \quad (2.9)$$

$$p(\mathbf{X}_n(yz)|\mathbf{Z}_{1:n}(yz)) = \frac{p(\mathbf{Z}_n(yz)|\mathbf{X}_n(yz))\ p(\mathbf{X}_n(yz)|\mathbf{Z}_{1:n-1}(yz))}{\int p(\mathbf{Z}_n(yz)|\mathbf{X}_n(yz))\ p(\mathbf{X}_n(yz)|\mathbf{Z}_{1:n-1}(yz))\ \mathrm{d}\mathbf{X}_n(yz)}, \quad (2.10)$$

$$p(\mathbf{X}_n(zx)|\mathbf{Z}_{1:n}(zx)) = \frac{p(\mathbf{Z}_n(zx)|\mathbf{X}_n(zx))\ p(\mathbf{X}_n(zx)|\mathbf{Z}_{1:n-1}(zx))}{\int p(\mathbf{Z}_n(zx)|\mathbf{X}_n(zx))\ p(\mathbf{X}_n(zx)|\mathbf{Z}_{1:n-1}(zx))\ \mathrm{d}\mathbf{X}_n(zx)}. \quad (2.11)$$

Three 2-D estimates from the posterior pdfs given by equations (2.9), (2.10) and (2.11) can be used to estimate a single object's 3-D state vector. However, equations (2.9), (2.10) and (2.11) are only for the conceptual purpose, and in general they cannot be computed analytically except in special cases such as the linear Gaussian state space model. Instead of using those equations, for a nonlinear system, the particle filter can approximate the posterior pdf using a cloud of particles, and a sequential importance sampling (SIS) can be applied to perform the nonlinear filtering [8]. The particle filtering further derives to the sequential importance resampling (SIR) algorithm which chooses the candidates of importance density and performs the resampling at every time instant [18]. In this chapter, we use the SIR particle

filter which has a generic particle filtering algorithm for an object tracking.

2.1.4 Dynamic Model and Observation Likelihood Function

Several dynamic models have been proposed to estimate the time-varying location and velocity. For the bearings-only tracking, three types of models are presented [12]. In 2-D xy-plane, the constant velocity (CV) model, the clockwise coordinated turn (CT) model, and the anti-clockwise coordinated turn (ACT) model are expressed by state transition matrices $\mathbf{F}_n^{(1)}$, $\mathbf{F}_n^{(2)}$ and $\mathbf{F}_n^{(3)}$ respectively as

$$\mathbf{F}_n^{(1)} = \begin{pmatrix} 1 & T_s & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{2.12}$$

and

$$\mathbf{F}_n^{(d)} = \begin{pmatrix} 1 & \sin(\Re_n^{(d)} T_s)/\Re_n^{(d)} & 0 & -(1-\cos(\Re_n^{(d)} T_s))/\Re_n^{(d)} \\ 0 & (1-\cos(\Re_n^{(d)} T_s))/\Re_n^{(d)} & 1 & \sin(\Re_n^{(d)} T_s)/\Re_n^{(d)} \\ 0 & \cos(\Re_n^{(d)} T_s) & 0 & -\sin(\Re_n^{(d)} T_s) \\ 0 & \sin(\Re_n^{(d)} T_s) & 0 & \cos(\Re_n^{(d)} T_s) \end{pmatrix}, \tag{2.13}$$

where T_s is the sampling period, d = 2,3 and $\Re_n^{(d)}$ is the mode-conditioned turning rate expressed as following;

$$\Re_n^{(2)} = \frac{\alpha}{\sqrt{(V_n^x)^2 + (V_n^y)^2}}, \text{ and } \Re_n^{(3)} = \frac{-\alpha}{\sqrt{(V_n^x)^2 + (V_n^y)^2}}, \tag{2.14}$$

where α is a constant for the rotated angle degree. In addition, Constant Acceleration (CA) model in xy-plane is expressed as follows,

$$\mathbf{F}_n^{(4)} = \begin{pmatrix} 1 & \left(A_x T_s^2/2V_{n-1}^x\right) + T_s & 0 & 0 \\ 0 & \left(A_x T_s/V_{n-1}^x + 1\right) & 0 & 0 \\ 0 & 0 & 1 & \left(A_y T_s^2/2V_{n-1}^y\right) + T_s \\ 0 & 0 & 0 & \left(A_y T_s/V_{n-1}^y\right) + 1 \end{pmatrix} \tag{2.15}$$

where A_x and A_y denote accelerations in xy-plane for x- and y-directions, respectively. For yz- and zx-planes, V^x and V^y in (2.14), and A_x and A_y in (2.15) are replaced according to the object state directional components. Furthermore, the CA model becomes the CV model when the values of A_x and A_y are zeros.

The SIR particle filter operates as follow [18]. After a dynamic model propagates the sets of M particles for $\mathbf{X}_{n-1}^{(1:M)}(xy)$, $\mathbf{X}_{n-1}^{(1:M)}(yz)$ and $\mathbf{X}_{n-1}^{(1:M)}(zx)$, new sets of particles $\mathbf{X}_n^{(1:M)}(xy)$, $\mathbf{X}_n^{(1:M)}(yz)$, and $\mathbf{X}_n^{(1:M)}(zx)$ are generated. Then, the observation likelihood functions

$$p\left(\mathbf{Z}_n(xy) \,\middle|\, \mathbf{X}_n^{(1:M)}(xy)\right),\ p\left(\mathbf{Z}_n(yz) \,\middle|\, \mathbf{X}_n^{(1:M)}(yz)\right),\ \text{and } p\left(\mathbf{Z}_n(zx) \,\middle|\, \mathbf{X}_n^{(1:M)}(zx)\right) \tag{2.16}$$

calculate weights of the generated particles and estimate $\mathbf{X}_n(xy)$, $\mathbf{X}_n(yz)$ and $\mathbf{X}_n(zx)$ respectively, through the resampling processes.

2.2 Projected Planes Selection for Object Tracking in 3-D Space

2.2.1 Projected Planes Selection (PPS) Method

Planes selection and particles generation

Instead of using the particle filter formulation with the direct 3-D state, the approach in this chapter is to use two of three possible 2-D particle filter formulations in order to estimate the 3-D state information. In the PPS method, we choose two planes with the smallest variance according to Figure 2-2 and Figure 2-3. Note that xy-plane is always chosen because the projected variance in xy-plane is the second best plane with the same variance as the originally measured azimuth angle θ. The other yz- or zx-plane is selected based on the measured angle. For example, when ϕ is measured between 45^o and 135^o, yz-plane is chosen. Otherwise, zx-plane is chosen.

Once the two planes are selected, the two 2-D particle filters estimate states separately. Figure 2-4 illustrates an example where xy- and yz-planes are chosen and the selected 2-D particle filters estimate the 3-D state vector (i.e., the projected measurement variance in yz-plane is less than the variance in zx-plane, according to the originally measured θ and ϕ). While the particle filters in the chosen planes estimate the state vectors, the particle filter in the other remained plane is waiting for

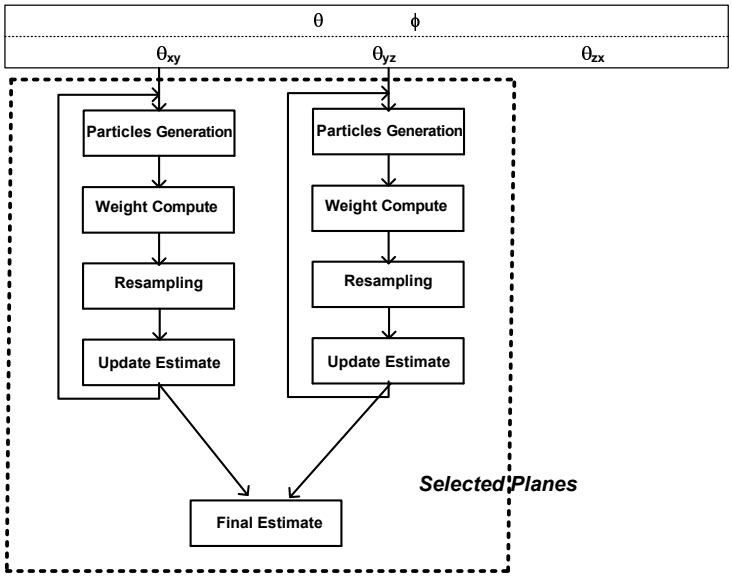


Figure 2-4: Illustration of Projection Planes Selection (PPS) method which chooses xy- and yz-planes while the other zx-plane is waiting for the plane selection.

the selection. When the measured angles become close to the range where the projected measurement variance in the remained plane becomes less than the originally measured variance, the selected plane is switched.

There is always one redundant component that appears in both planes (i.e., y-component appears in xy-and yz-planes). As two particle filters are estimating the states separately, the redundant directional state from two particle filters may differ. For example, as discussed in (2.8), the intermediate 2-D object state vectors are given as $(x_n(xy), V_n^x(xy), y_n(xy), V_n^y(xy))^T$ from the xy-plane particle filter and $(y_n(yz), V_n^y(yz), z_n(yz), V_n^z(yz))^T$ from the yz-plane particle filter. Both $y_n(xy)$ and $y_n(yz)$ represent y directional position information, but the two values are different. Therefore, a combining method should be considered in order to get one final 3-D object state vector $\mathbf{X}_n$.

Redundancy consideration in combining method

There are two ways to combine the two estimates of y-direction's state vectors when xy- and yz- planes are selected; the planes weighted combining and the equal weight combining.

In the planes weighted combining method, the two estimates are weighted according to the unnormalized particles' weight-sum of each plane's particle filter. This method is derived from the multiple particle filtering method [20], and extended to combine into a final value with respect to the redundant state. Since a particle represents a point mass of the probability density, the unnormalized particles weight-sum can be used in evaluating how the expected state is close to the true state [16] [20] [21]. The final 3-D object state vector $\mathbf{X}_n$ with the planes weighted combining method is obtained by

$$\mathbf{X}_n = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \mathbf{X}_n(x|xyz) + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \mathbf{X}_n(y|xyz) + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \mathbf{X}_n(z|xyz), \quad (2.17)$$

where $\mathbf{X}_n(x|xyz)$, $\mathbf{X}_n(y|xyz)$ and $\mathbf{X}_n(z|xyz)$ are final 3-D estimated vectors with respect to each directional component representing $[x_n, V_n^x]^T$, $[y_n, V_n^y]^T$ and $[z_n, V_n^z]^T$,

respectively. When xy- and yz-planes are selected

$$\begin{aligned}
\mathbf{X}_n(x|xyz) &= \mathbf{X}_n(x|xy), && (2.18)\\
\mathbf{X}_n(y|xyz) &= \frac{\mathbf{X}_n(y|xy)\sum_{i=1}^{M} w_n^{(i)}(xy) + \mathbf{X}_n(y|yz)\sum_{i=1}^{M} w_n^{(i)}(yz)}{\sum_{i=1}^{M} w_n^{(i)}(xy) + \sum_{i=1}^{M} w_n^{(i)}(yz)}, && (2.19)\\
\mathbf{X}_n(z|xyz) &= \mathbf{X}_n(z|yz), && (2.20)
\end{aligned}$$

where $\mathbf{X}_n(x|xy)$ and $\mathbf{X}_n(y|xy)$ represent the x and y directional 2-D state vectors in xy-plane. $\mathbf{X}_n(y|yz)$ and $\mathbf{X}_n(z|yz)$ represent the y and z directional 2-D state vectors in yz-plane. $w_n^{(i)}(xy)$ and $w_n^{(i)}(yz)$ are the i-th particle's weight of the particle filter for xy- and yz-plane at time instant n, and M represents the the number of particles for each particle filter. Thus, the redundant y directional states are combined as in (2.19), where the weighting factors are $\sum_{i=1}^{M} w_n^{(i)}(xy)$ to xy-plane and $\sum_{i=1}^{M} w_n^{(i)}(yz)$ to yz-plane.

For the equal weight combining method, as it simply takes an average value, the redundant component y in (2.19) is replaced by

$$\mathbf{X}_n(y|xyz) = \frac{\mathbf{X}_n(y|xy) + \mathbf{X}_n(y|yz)}{2}. \qquad (2.21)$$

The Algorithm 1 summarizes the PPS with the planes weighted combining method.

Algorithm 1: Projected Planes Selection (PPS) with Planes Weighted Combining Method

Given $\mathbf{Z}_n$, calculate $\mathbf{Z}_n(p)$ based on (2.1) and (2.2), where $p \in \{xy, yz, zx\}$.
Find the plane with a smaller variance of measurements between yz and zx based on Figure 2-2 and 2-3.
Independently estimate an object state vector $\mathbf{X}_n(p_1)$ with $\mathbf{Z}_n(p_1)$ and $\mathbf{X}_n(p_2)$ with $\mathbf{Z}_n(p_2)$.

Draw $\mathbf{X}_n^{(1:M)}(p_1) \sim p(\mathbf{X}_n(p_1)|\mathbf{X}_{n-1}^{(1:M)}(p_1))$ and $\mathbf{X}_n^{(1:M)}(p_2) \sim p(\mathbf{X}_n(p_2)|\mathbf{X}_{n-1}^{(1:M)}(p_2))$

Calculate $w_n^{(1:M)}(p_1) = p(\mathbf{Z}_n(p_1)|\mathbf{X}_n^{(1:M)}(p_1))$ and $w_n^{(1:M)}(p_2) = p(\mathbf{Z}_n(p_2)|\mathbf{X}_n^{(1:M)}(p_2))$

Calculate total weights: $\sum_{i=1}^{M} w_n^{(i)}(p_1)$ and $\sum_{i=1}^{M} w_n^{(i)}(p_2)$

Planes Weighted Combining with Redundancy for $\mathbf{X}_n(x|xyz)$, $\mathbf{X}_n(y|xyz)$ and $\mathbf{X}_n(z|xyz)$

if $p_1 = yz$ **then**

$\mathbf{X}_n(x|xyz) = \mathbf{X}_n(x|xy)$.

$\mathbf{X}_n(y|xyz) = \frac{\mathbf{X}_n(y|xy)\sum_{i=1}^{M} w_n^{(i)}(xy) + \mathbf{X}_n(y|yz)\sum_{i=1}^{M} w_n^{(i)}(yz)}{\sum_{i=1}^{M} w_n^{(i)}(xy) + \sum_{i=1}^{M} w_n^{(i)}(yz)}$.

$\mathbf{X}_n(z|xyz) = \mathbf{X}_n(z|yz)$.

else if $p_1 = zx$ **then**

$\mathbf{X}_n(x|xyz) = \frac{\mathbf{X}_n(x|xy)\sum_{i=1}^{M} w_n^{(i)}(xy) + \mathbf{X}_n(x|zx)\sum_{i=1}^{M} w_n^{(i)}(zx)}{\sum_{i=1}^{M} w_n^{(i)}(xy) + \sum_{i=1}^{M} w_n^{(i)}(zx)}$.

$\mathbf{X}_n(y|xyz) = \mathbf{X}_n(y|xy)$.

$\mathbf{X}_n(z|xyz) = \mathbf{X}_n(z|zx)$.

end

2.2.2 Discussion

Planes Weighted Combining Versus Equal Weight Combining

It has been assumed that the nonlinear dynamic transition function f_n is known as the state transition matrix $\mathbf{F}_n$ – as the particle filter is a model-based approach. If the dynamic model f_n changes in the middle of the tracking, the estimation from the particle filter can diverge. The divergence means that a predicted state and a true state continuously become more distant due to unmatched model of a particle filter. Also, if the state of the unmatched model lasts longer, the estimation may not

recover even after recovering the model. The planes weighted combining method can discard the estimation from the plane with negligible unnormalized particles weight-sum based on the likelihood function $p\left(\mathbf{Z}_n \left| \mathbf{X}_n^{(1:M)} \right.\right)$, and thus prevents the estimation from deviation.

The equal weight combining and the planes weighted combining methods have similar tracking performances if all selected plane-particle filters show good tracking performances. However, when one of two particle filters' tracking performance deteriorates, the planes weighted combining method shows a better performance. The scenario to be investigated is that an object is moving in the range of ϕ being between original measurements are between 50.47^o and 82.14^o as well as in the range of θ being between 36.35^o and 85.60^o. More specifically, a single sensor is placed in an origin ($0m$, $0m$, $0m$), and an initial position of the object is ($10m$, $130m$, $18m$) with an initial velocity of ($1m/s$, -$0.75m/s$, $0.75m/s$). The sensor is measuring θ and ϕ with the interval of 1 second for 165 seconds, and the variances of the measurements are both 3. The observed object is moving in CV at the x-direction and in CT at the y and z directions with $0.01m/s^2$. Since the ϕ is measured in the range between 50.47^o and 82.14^o, xy- and yz-planes are selected. In addition, the initial object state is given. TABLE 2.1 shows the comparison of RMSE between the two combining methods when all selected plane-particle filters hold correct dynamic models. On the other hand, TABLE 2.2 shows the comparison between the two combining methods when yz-plane particle filter has incorrect dynamic models: the dynamic model is temporarily manipulated with CV instead of CT during 50 seconds. 1,000 particles are used for generating the results.

RMSE	Equal Weight Combining	Weighted Combining
x component	1.8267	1.7258
y component	0.5673	0.5378
z component	1.6134	1.6235
average	1.3358	1.2957

Table 2.1: RMSE of the equal weight combining method versus the weighted combining method when all plane-particle filters have good tracking performance (100 times simulation).

RMSE	Equal Weight Combining	Weighted Combining
x component	1.7958	1.6459
y component	17.3250	0.6243
z component	1.5783	1.6431
average	6.8997	1.3044

Table 2.2: RMSE of the equal weight combining method versus the weighted combining method when yz-plane particle filter has poor tracking performance (100 times simulation).

The tracking performance is also shown in Figure 2-5 and Figure 2-6, where the particle filter in yz-plane results in deviated estimation. Since xy- and yz-planes are selected, y direction's state estimates are combined. Figure 2-5 shows the example of tracking deviation in yz-plane due to the unmatched model or a particle filter's performance degradation. Figure 2-6 shows a final estimation after applying two combining methods. Especially in Figure 2-6(b), it is shown that the planes weighted combining method maintains the object tracking by considering the contribution of unnormalized particles' weight-sums from different planes.

PPS Versus Direct 3-D Method

The 3-D object state model directly uses two original measurements and a cone shape likelihood function for assigning 3-D distributed particle weights [22]. The direct

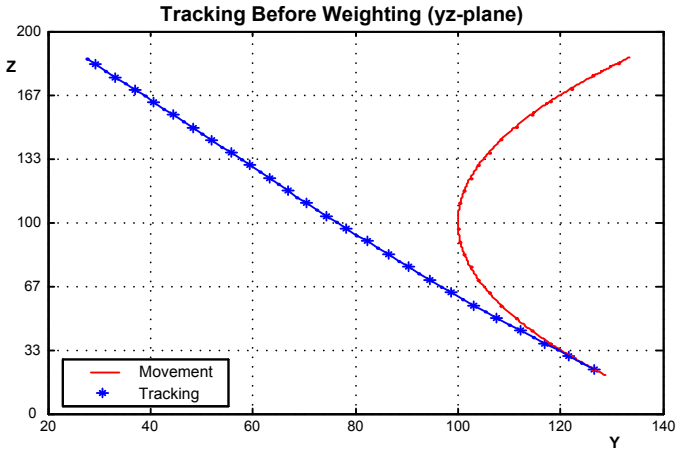


Figure 2-5: Poor tracking performance in yz-plane without combining methods. (Number of particles : 1,000)

3-D Method uses the two original measurements with σ^2 while the PPS method uses two projected measurements with σ^2_{xy} and $\min(\sigma^2_{yz}, \sigma^2_{zx})$. Figure 2-7 shows the unnormalized particles weight-sums corresponding to the selected yz-plane and the direct 3-D model. It is shown that the selected plane is less sensitive to measurement noise than the direct 3-D model; thus, the unnormalized particles weight-sums of PPS method is larger than those of the direct 3-D Method. In addition, the direct 3-D Method cannot obtain a redundancy, and thus there is no opportunity to avoid the performance degradation when a particle filter has a poor performance. The performances are compared according to the Cramer-Rao Lower bound (CRLB) in Chapter 2.3.

2.2.3 Extended PPS Method with Multiple Sensors

The tracking trajectory deviation due to an unexpected change of an object is shown to be mitigated by applying the planes weighted combining method. In addition,

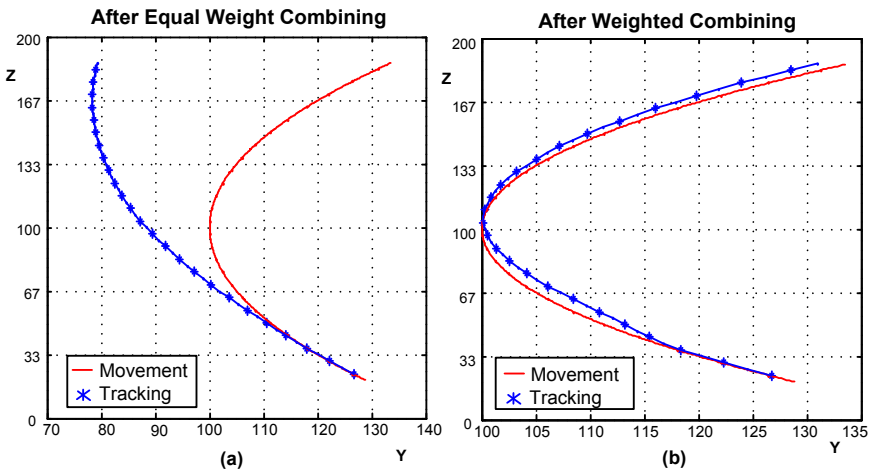


Figure 2-6: Modified tracking performance with combining methods (Number of particles : 1,000) (a) Equal weight combining method (b) Weighted combining method.

the PPS method can also be extended to multiple sensor environment by considering multiple particle filter fusion. Multiple sensor based particle filtering has been introduced for a robust tracking, especially when some of measurements are severely corrupted [12] [19]. In this subchapter, we present the extended PPS method in multiple sensor environment.

Global Coordinate Transformation

Denote k, $k = 0{,}1\ ,\ 2,\ \cdots,\ K-1$, the index of sensor when there are K acoustic sensors. Also, define the location of k-th sensor as $\left(x_s^k, y_s^k, z_s^k\right)$. As illustrated in Figure 2-8, each sensor has its own coordinate system with x and y directional unit vectors as $\mathbf{u}_x^k$ and $\mathbf{u}_y^k$ in xy-plane; the same notation and illustration are applicable for yz- and zx-planes. Each data in the different coordinate system should be converted to the global coordinate, especially when the data obtained from multiple sensors are collected and associated. Assume that the global coordinate is corresponding to the

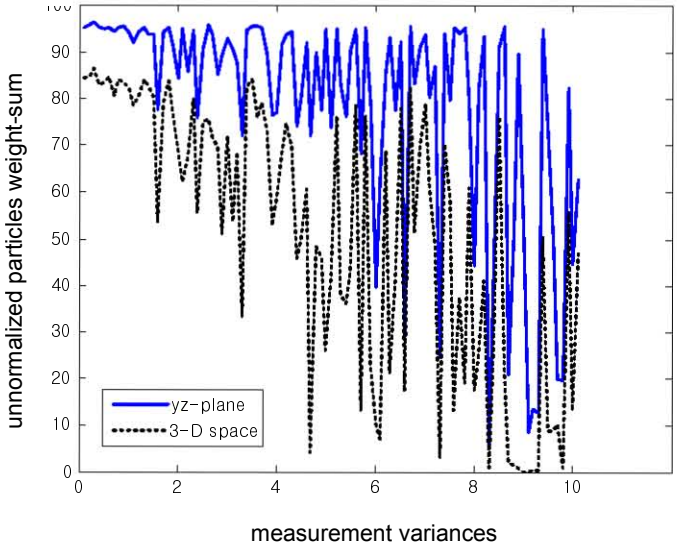


Figure 2-7: Comparison between the selected yz- planes and 3-D space: unnormalized particles weight-sums according to the variances of original measurements (The number of particles: 100).

coordinate of one senor called the primary sensor and for example, sensor 0 is the primary sensor as illustrated in Figure 2-8. The primary sensor is placed at the origin as $x_s^0 = y_s^0 = 0$, and $x_s^0 \leq x_s^k$ and $y_s^0 \leq y_s^k$, for k = 1,2,3.

Given an object state vector $\mathbf{X}_n(xy)$ in the global coordinate, each sensor differently expresses the objects state vector as $\mathbf{X}_n^k(xy)$ in its own coordinate, where k represents the sensor index. The data conversion from $\mathbf{X}_n^k(xy)$ to $\mathbf{X}_n(xy)$ is done by a multiplication of conversion matrix $\mathbf{D}^k(xy)$ and an addition of shift matrix $\mathbf{S}^k(xy)$ as

$$\mathbf{X}_n(xy) = \mathbf{D}^k(xy)\mathbf{X}_n^k(xy) + \mathbf{S}^k(xy), \tag{2.22}$$

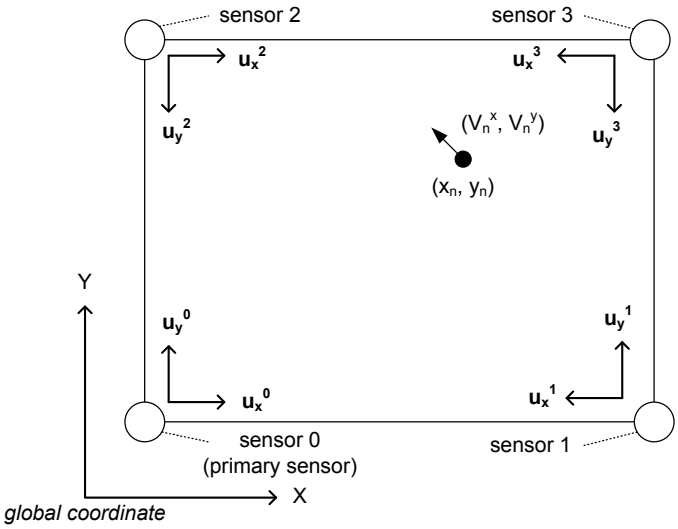


Figure 2-8: Each sensor has its own coordinate, and the primary sensor coordinate is the global coordinate.

where $\mathbf{D}^k(xy)$ is a 4×4 matrix and $\mathbf{S}^k(xy)$ is a 4×1 vector as

$$\mathbf{D}^k(xy) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \mathbf{u}_x^k \cdot \mathbf{u}_x^0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \mathbf{u}_y^k \cdot \mathbf{u}_y^0 \end{pmatrix} \text{ and } \mathbf{S}^k(xy) = \begin{pmatrix} s_x^k \\ 0 \\ s_y^k \\ 0 \end{pmatrix}, \tag{2.23}$$

where

$$s_x^k = \begin{cases} -x_s^k, & \text{for } \mathbf{u}_x^k \cdot \mathbf{u}_x^0 = 1, \\ x_s^k - x_n^k(xy), & \text{for } \mathbf{u}_x^k \cdot \mathbf{u}_x^0 = -1, \end{cases} \quad s_y^k = \begin{cases} -y_s^k, & \text{for } \mathbf{u}_y^k \cdot \mathbf{u}_y^0 = 1, \\ y_s^k - y_n^k(xy), & \text{for } \mathbf{u}_y^k \cdot \mathbf{u}_y^0 = -1. \end{cases} \tag{2.24}$$

In (2.23), $\mathbf{u}_x^k \cdot \mathbf{u}_x^0$ determines the polarity of x directional velocity component of the state vector, and $\mathbf{u}_y^k \cdot \mathbf{u}_y^0$ determines the polarity of y directional velocity component of the state vector. In (2.23), s_x^k and s_y^k are x and y directional shifted distances

according to the relative positions between senor 0 and sensor k. They also depend on the values of $\mathbf{u}_x^k \cdot \mathbf{u}_x^0$ and $\mathbf{u}_y^k \cdot \mathbf{u}_y^0$ as in (2.24).

Extended PPS Method

The extended PPS method is to select two projected measurements with the lowest variances and estimate the state vectors independently. Denote all the projected measurements at time instant n as $\mathbf{Z}_n^k(\mathtt{P})$, where k, $k \in \{0, 1, 2, \cdots, K-1\}$, is the sensor index and $\mathtt{P}$, $\mathtt{P} \in \{xy, yz, zx\}$, represents the plane. Also, denote the two selected measurements as $\mathbf{Z}_n^{k_1}(\mathtt{P}_1)$ and $\mathbf{Z}_n^{k_2}(\mathtt{P}_2)$, where $\mathbf{Z}_n^{k_1}(\mathtt{P}_1)$ is with the lowest variance corresponding to the projected measurement to $\mathtt{P}_1$-plane from sensor k_1, $\mathbf{Z}_n^{k_2}(\mathtt{P}_2)$ is with the next lowest variance corresponding to the projected measurement to $\mathtt{P}_2$-plane from sensor k_2, and $\mathtt{P}_1 \neq \mathtt{P}_2$ (i.e., xy- and yz- planes, xy- and zx- planes or yz- and zx- planes). First, the measurement $\mathbf{Z}_n^{k_1}(\mathtt{P}_1)$ is selected by finding the lowest variance among $(\sigma_n^k(\mathtt{P}))^2$ corresponding to all projected measurements $\mathbf{Z}_n^k(\mathtt{P})$, where $(\sigma_n^k(\mathtt{P}))^2$ is obtained based on the range of the original measurement ϕ and θ according to the variances of the projected angles shown in Figure 2-2 and Figure 2-3. If $\mathtt{P}_1$ is zx-plane, $\mathbf{Z}_n^{k_2}(\mathtt{P}_2)$ is selected by finding the lowest variance among $(\sigma_n^k(yz))^2$ and $(\sigma_n^k(xy))^2$ corresponding to projected measurements $\mathbf{Z}_n^k(yz)$ and $\mathbf{Z}_n^k(xy)$. In other words, $\mathbf{Z}_n^{k_2}(\mathtt{P}_2)$ should be selected among $\mathtt{P}_2$, such that $\mathtt{P}_2 \neq \mathtt{P}_1$.

After measurements $\mathbf{Z}_n^{k_1}(\mathtt{P}_1)$ and $\mathbf{Z}_n^{k_2}(\mathtt{P}_2)$ are selected, their own 2-D particle filters estimate the object state vectors independently. Once the estimated state vectors are obtained, they should be converted to the global coordinate with respect to a primary sensor as in (2.22). As in the PPS method, there is one redundant directional

component, and the sensor-weighted combining method can be considered in evaluating weighting values between the two selected sensors. The unnormalized particles' wight-sums $W_n^{k_1}$ and $W_n^{k_2}$ obtained from each selected plane from different sensors are used for the combining and they are evaluated as

$$W_n^{k_1} = \sum_{i=1}^{M} w_n^{k_1,(i)}(\mathtt{P}_1), \text{ and } W_n^{k_2} = \sum_{i=1}^{M} w_n^{k_2,(i)}(\mathtt{P}_2), \tag{2.25}$$

where $w_n^{k_1,(i)}(\mathtt{P}_1)$ represents the i^{th} particle weight in $\mathtt{P}_1$-plane of sensor k_1, and $w_n^{k_2,(i)}(\mathtt{P}_2)$ represents the i^{th} particle weight in $\mathtt{P}_2$-plane of sensor k_2. Similarly to the example in (2.19), when xy- and yz-planes are selected from sensor k_1 and k_2, respectively, the redundant y-direction components are combined as

$$\mathbf{X}_n(y|xyz) = \frac{\mathbf{X}_n^{k_1}(y|xy)W_n^{k_1} + \mathbf{X}_n^{k_2}(y|yz)W_n^{k_2}}{W_n^{k_1} + W_n^{k_2}}, \tag{2.26}$$

and for the non-redundant directional components,

$$\mathbf{X}_n(x|xyz) = \mathbf{X}_n^{k_1}(x|xy), \text{ and } \mathbf{X}_n(z|xyz) = \mathbf{X}_n^{k_2}(z|yz). \tag{2.27}$$

Finally, (2.17) is used to obtain the final 3-D state vectors.

2.3 Cramer-Rao Lower Bound Derivation and Performance Analysis

The Cramer-Rao Lower Bound (CRLB) has been widely used as a reference in evaluating an estimator by representing the minimum covariance of the estimated states that an unbiased estimator can achieve. For the object tracking problem with bearings-only measurements, the CRLB is investigated in [23], and the similar approaches are taken in this chapter. As in [23], we assume that the process noise $\mathbf{Q}_n$ is zero and the dynamic models are fixed and known; otherwise, the derivation is intractable. The covariance matrix of the state estimate $\hat{\mathbf{X}}_n$ is given as follow

$$\mathbf{C}_n = E\left[\left(\hat{\mathbf{X}}_n - \mathbf{X}_n\right)\left(\hat{\mathbf{X}}_n - \mathbf{X}_n\right)^T\right] \geq \mathbf{J}_n^{-1}, \tag{2.28}$$

where $\mathbf{J}_n$ is the information matrix, and it is defined as

$$\mathbf{J}_n = E\left\{\left[\nabla_{\mathbf{X}_n} \log p(\mathbf{X}_n|\mathbf{Z}_n)\right]\left[\nabla_{\mathbf{X}_n} \log p(\mathbf{X}_n|\mathbf{Z}_n)\right]^T\right\}, \tag{2.29}$$

where $\nabla_{\mathbf{X}_n}$ denotes the gradient operator with respect to the state vector $\mathbf{X}_n$ and $p(\mathbf{X}_n|\mathbf{Z}_n)$ is the conditional pdf of state $\mathbf{X}_n$ given the observation $\mathbf{Z}_n$. Note that the inequality of the square matrix in (2.28) means that matrix $\mathbf{C}_n - \mathbf{J}_n^{-1}$ is positive definite. The CRLB's of the components in the state vector $\mathbf{X}_n$ is the lower bound of its variance and it is the diagonal elements of the inverse matrix of $\mathbf{J}_n$ [24].

We do not directly obtain the information matrix as in (2.29), but it is derived

recursively as follow. In the absence of the process noise, the evolution of the state vector is deterministic and it is given as [19] [25]

$$\mathbf{J}_{n+1} = \left[\mathbf{F}_n^{-1}\right]^T \mathbf{J}_n \mathbf{F}_n^{-1} + \mathbf{H}_{n+1}^T \mathbf{R}_{n+1}^{-1} \mathbf{H}_{n+1}, \tag{2.30}$$

where $\mathbf{F}_n$ is the state transition matrix that represents CV or CA as shown in (2.13) and (2.15), $\mathbf{R}_{n+1}$ is the covariance matrix of the bearing measurements and $\mathbf{H}_n$ is the gradient component of a measurement function h_n. $\mathbf{H}_n$ is given as follow and it is referred to as the Jacobian of h_n,

$$\mathbf{H}_n = \left(\nabla_{\mathbf{X}_n} h_n^T(\mathbf{X}_n)\right)^T. \tag{2.31}$$

In the following subchapters, the CRLB's for the PPS method are compared against the direct 3-D Method. The dynamic model of interest is assumed to be CV in a x-axis, CA with A_y and A_z in y and z-axis.

2.3.1 CRLB Derivation based on the PPS Method

In the PPS method, two information matrices in (2.30) are generated for each selected plane. For a clear notation, we put the plane type $\mathtt{P}$ as $\mathbf{J}_n^{\mathtt{P}}$, which represents $\mathbf{J}_n^{xy}$, $\mathbf{J}_n^{yz}$ or $\mathbf{J}_n^{zx}$. Similarly, the transition matrix, measurement variance and Jacobian of h_n are also denoted as $\mathbf{F}_n^{\mathtt{P}}$, $\mathbf{R}_n^{\mathtt{P}}$ and $\mathbf{H}_n^{\mathtt{P}}$, respectively for $\mathtt{P} \in \{xy, yz, zx\}$. From (2.12)

and (2.15), transition matrices $\mathbf{F}_n^{\mathtt{P}}$'s are derived as

$$\mathbf{F}_n^{xy} = \begin{pmatrix} 1 & T_s & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & A_y T_s^2/2V_{n-1}^y + T_s \\ 0 & 0 & 0 & A_y T_s/V_{n-1}^y + 1 \end{pmatrix}, \quad \mathbf{F}_n^{zx} = \begin{pmatrix} 1 & A_z T_s^2/2V_{n-1}^z + T_s & 0 & 0 \\ 0 & A_z T_s/V_{n-1}^z + 1 & 0 & 0 \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{2.32}$$

and

$$\mathbf{F}_n^{yz} = \begin{pmatrix} 1 & A_y T_s^2/2V_{n-1}^y + T_s & 0 & 0 \\ 0 & A_y T_s/V_{n-1}^y + 1 & 0 & 0 \\ 0 & 0 & 1 & A_z T_s^2/2V_{n-1}^z + T_s \\ 0 & 0 & 0 & A_z T_s/V_{n-1}^z + 1 \end{pmatrix}. \tag{2.33}$$

In the PPS method, the covariance matrix of measurement, $\mathbf{R}_n^{\mathtt{P}}$ becomes σ_{xy}^2, σ_{yz}^2 or σ_{zx}^2 which is the variance of a single (projected) bearing measurement in the projected plane xy, yx or zx-plane respectively. The performance of the PPS method is mainly enhanced by taking only the measurement with a smaller variance. According to Figure 2-2 and 2-3, the raw bearings, θ and ϕ are projected onto the three planes with the different angle variances according to the object's position.

For Jacobians in xy-plane $\mathbf{H}_{n+1}^{xy}$ is derived from

$$h_{n+1}^T\left(\mathbf{X}_{n+1}(xy)\right) = \theta_{xy}\left(\mathbf{X}_{n+1}(xy)\right) = \arctan\left(\frac{y_{n+1}}{x_{n+1}}\right) \tag{2.34}$$

and

$$\frac{\partial}{\partial x_{n+1}} \arctan\left(\frac{y_{n+1}}{x_{n+1}}\right) = \frac{-y_{n+1}}{x_{n+1}^2 + y_{n+1}^2}, \quad \frac{\partial}{\partial y_{n+1}} \arctan\left(\frac{y_{n+1}}{x_{n+1}}\right) = \frac{x_{n+1}}{x_{n+1}^2 + y_{n+1}^2},$$

$$\frac{\partial}{\partial V_{n+1}^x} \arctan\left(\frac{y_{n+1}}{x_{n+1}}\right) = \frac{\partial}{\partial V_{n+1}^y} \arctan\left(\frac{y_{n+1}}{x_{n+1}}\right) = 0. \tag{2.35}$$

Then,

$$\mathbf{H}_{n+1}^{xy} = \left(\nabla_{\mathbf{x}_{n+1}(xy)} h_{n+1}^T(\mathbf{X}_{n+1}(xy))\right)^T = \left(\frac{-y_{n+1}}{x_{n+1}^2 + y_{n+1}^2}, 0, \frac{x_{n+1}}{x_{n+1}^2 + y_{n+1}^2}, 0 \right), \tag{2.36}$$

and by the same way, Jacobians for yz- and zx-planes are derived as follow

$$\mathbf{H}_{n+1}^{yz} = \left(\frac{-z_{n+1}}{y_{n+1}^2 + z_{n+1}^2}, 0, \frac{y_{n+1}}{y_{n+1}^2 + z_{n+1}^2}, 0 \right), \tag{2.37}$$

and

$$\mathbf{H}_{n+1}^{zx} = \left(\frac{-x_{n+1}}{x_{n+1}^2 + z_{n+1}^2}, 0, \frac{z_{n+1}}{x_{n+1}^2 + z_{n+1}^2}, 0 \right). \tag{2.38}$$

For the PPS method with a single sensor, the information matrix $\mathbf{J}_n$ given in (2.30) can be recursively obtained using equations from (2.32) to (2.38) except the initial condition. We can assume that $\mathbf{J}_0$ is the zero matrix – no information at all at the beginning of the estimation.

2.3.2 CRLB Derivation based on the Direct 3-D Method

In the direct 3-D method, the information matrix $\mathbf{J}_n$ is expressed as a 6×6 matrix, and the lower bound is directly obtained from (2.30) with the extension of 2-D state vector based matrices.

The state transition matrix is expressed as

$$\mathbf{F}_n = \begin{pmatrix} 1 & T_s & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & A_y T_s^2/2V_{n-1}^y + T_s & 0 & 0 \\ 0 & 0 & 0 & A_y T_s/V_{n-1}^y + 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & A_z T_s^2/2V_{n-1}^z + T_s \\ 0 & 0 & 0 & 0 & 0 & A_z T_s/V_{n-1}^z + 1 \end{pmatrix}. \tag{2.39}$$

Measured bearings vector $[\theta, \phi]^T$ is given with variances σ_θ^2 and σ_ϕ^2, and it can be noted that the two bearings' tracking are simply extended to multiple sensors tracking. For the 3-D state vector estimation, only a single sensor detects bearings physically, but the bearings measurement should be interpreted such that two different sensors detect each angle independently. The measurement error covariance $\mathbf{R}_n$ and the Jacobian $\mathbf{H}_{n+1}$ are expressed as in the multiple sensors' case as follow,

$$\mathbf{R}_n = \begin{pmatrix} \sigma_\theta^2 & 0 \\ 0 & \sigma_\phi^2 \end{pmatrix}, \tag{2.40}$$

and

$$
\begin{aligned}
\mathbf{H}_{n+1} &= \left(\nabla_{\mathbf{x}_{n+1}} \left[\; h^{\theta}_{n+1}(\mathbf{X}_{n+1})\; h^{\phi}_{n+1}(\mathbf{X}_{n+1}) \;\right]\right)^{T} \\
&= \begin{pmatrix}
\dfrac{\partial h^{\theta}}{\partial x_{n+1}} & \dfrac{\partial h^{\theta}}{\partial V^{x}_{n+1}} & \dfrac{\partial h^{\theta}}{\partial y_{n+1}} & \dfrac{\partial h^{\theta}}{\partial V^{y}_{n+1}} & \dfrac{\partial h^{\theta}}{\partial z_{n+1}} & \dfrac{\partial h^{\theta}}{\partial V^{z}_{n+1}} \\
\dfrac{\partial h^{\phi}}{\partial x_{n+1}} & \dfrac{\partial h^{\phi}}{\partial V^{x}_{n+1}} & \dfrac{\partial h^{\phi}}{\partial y_{n+1}} & \dfrac{\partial h^{\phi}}{\partial V^{y}_{n+1}} & \dfrac{\partial h^{\phi}}{\partial z_{n+1}} & \dfrac{\partial h^{\phi}}{\partial V^{z}_{n+1}}
\end{pmatrix} \\
&= \begin{pmatrix}
\dfrac{-y_{n+1}}{x^2_{n+1}+y^2_{n+1}} & \dfrac{x_{n+1}z_{n+1}}{(x^2_{n+1}+y^2_{n+1}+z^2_{n+1})\sqrt{x^2_{n+1}+y^2_{n+1}}} \\
0 & 0 \\
\dfrac{x_{n+1}}{x^2_{n+1}+y^2_{n+1}} & \dfrac{y_{n+1}z_{n+1}}{(x^2_{n+1}+y^2_{n+1}+z^2_{n+1})\sqrt{x^2_{n+1}+y^2_{n+1}}} \\
0 & 0 \\
0 & \dfrac{-\sqrt{x^2_{n+1}+y^2_{n+1}}}{(x^2_{n+1}+y^2_{n+1}+z^2_{n+1})} \\
0 & 0
\end{pmatrix}^{T},
\end{aligned}
\tag{2.41}
$$

where h^{θ}_{n} and h^{ϕ}_{n} are measurement functions of bearings θ and ϕ, respectively. Also, we can assume that $\mathbf{J}_0$ is the zero matrix – no information at all at the beginning of the estimation.

2.4 Analysis and Simulation

In this Chapter, the PPS and the direct 3D methods are compared with their simulation results and CRLB's. As the proposed method selects the smallest measurement variance, the covariance $\mathbf{R}_n$ plays an important role for the lower bound. The minimum covariances obtained from the PPS method minimizes the lower bound – the

PPS method is to flexibly choose planes with the smallest variances. Several scenarios are considered for the performance comparisons. Scenario 1 and 2 show the single sensor based plane selection according to ϕ. Scenario 3 shows the changes of the plane selection from xy- and yz-planes to xy- and zx-planes according to ϕ. Scenario 4 shows the multiple sensors based planes and sensors selection according to θ and ϕ.

2.4.1 Scenario 1

In this scenario, an object is moving in the range of ϕ being between 45.36^o and 76.74^o as well as in the range of θ being between 45.00^o and 49.04^o. More specifically, a single sensor is placed in the origin ($0m$, $0m$, $0m$), and the initial position of the object is ($3m$, $3m$, $1m$) with initial velocity of ($1m/s$, $1m/s$, $1m/s$). The sensor is measuring θ and ϕ with the interval of 0.1 second and the variances of the measurements are both 3. The observed object is moving in CV at the x-direction, in CA at the y and z directions, with $0.1m/s^2$ and $0.5m/s^2$, respectively. Since the ϕ is measured in the range between 45.36^o and 76.74^o, xy- and yz-planes are selected. In addition, the initial object state is given.

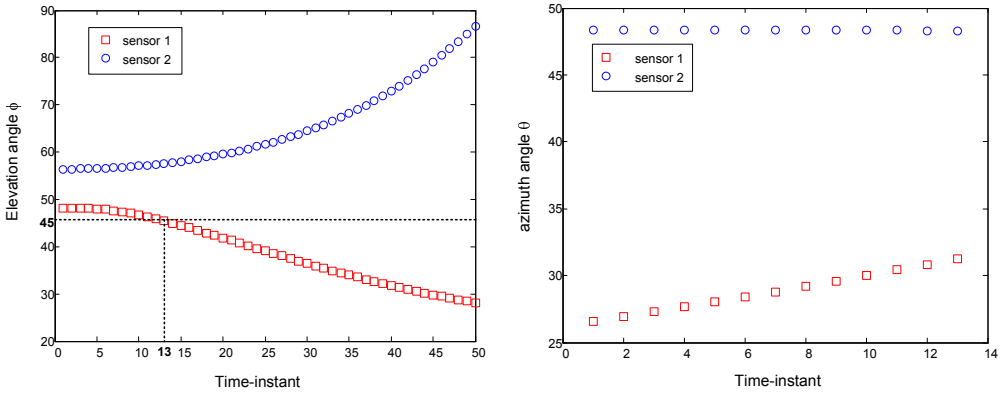
2.4.2 Scenario 2

In this scenario, an object is moving in the range of ϕ being between 25.24^o and 36.26^o as well as in the range of θ being between 45.00^o and 50.28^o. Similar to scenario 1, a single sensor is placed at the origin ($0m$, $0m$, $0m$) with the same initial object velocity and movement. Initial position of the object is ($1m$, $1m$, $3m$). Since the ϕ is in the

range between 25.24^o and 36.26^o, xy- and zx-planes are selected. Also, the initial object state is given.

2.4.3 Scenario 3

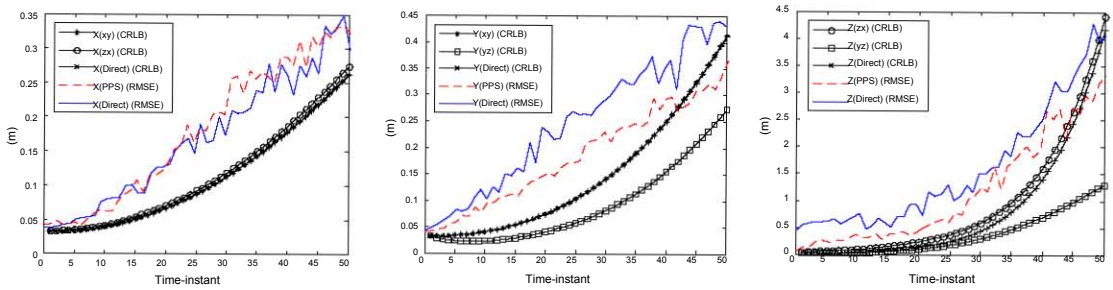
In this scenario, an object is moving in the range of ϕ being between 28.07^o and 48.24^o crossing 45^o. More specifically, a single sensor is placed at the origin ($0m$, $0m$, $0m$), and the initial position of the object is ($2m$, $1m$, $2m$) with initial velocity of ($0.3m/s$, $0.3m/s$, $0.3m/s$). Similar to previous scenarios, the observed object is moving in CV at the x-direction, in CA at the y and z directions, with $0.1m/s^2$ and $0.5m/s^2$, respectively. Since ϕ of the first 13 time instants is measured between 48.24^o and 45.42^o, xy- and yz-planes are selected. In the last 37 time instants, xy- and zx-planes are selected since ϕ is measured between 28.07^o and 44.96^o.



(a) Elevation angles ϕ in the view of two multiple sensors

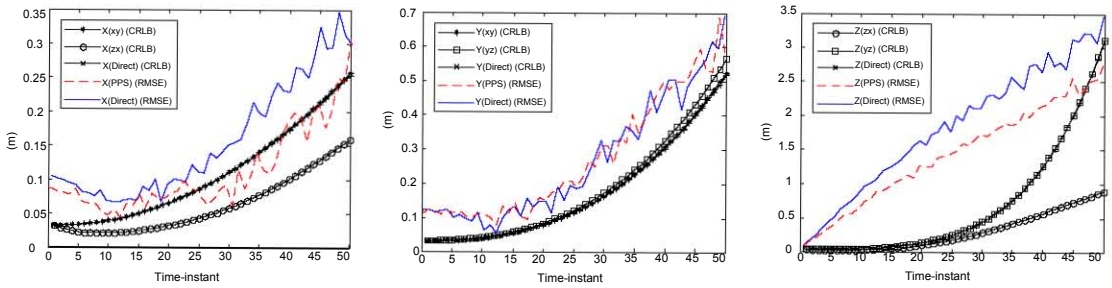
(b) Azimuth angles θ in the view of two multiple sensors

Figure 2-9: Elevation angles ϕ and azimuth angles θ in the view of two multiple sensors



(a) CRLB X (b) CRLB Y (c) CRLB Z

Figure 2-10: Scenario 1: Selected xy- and yz- planes based on PPS shows better performance.



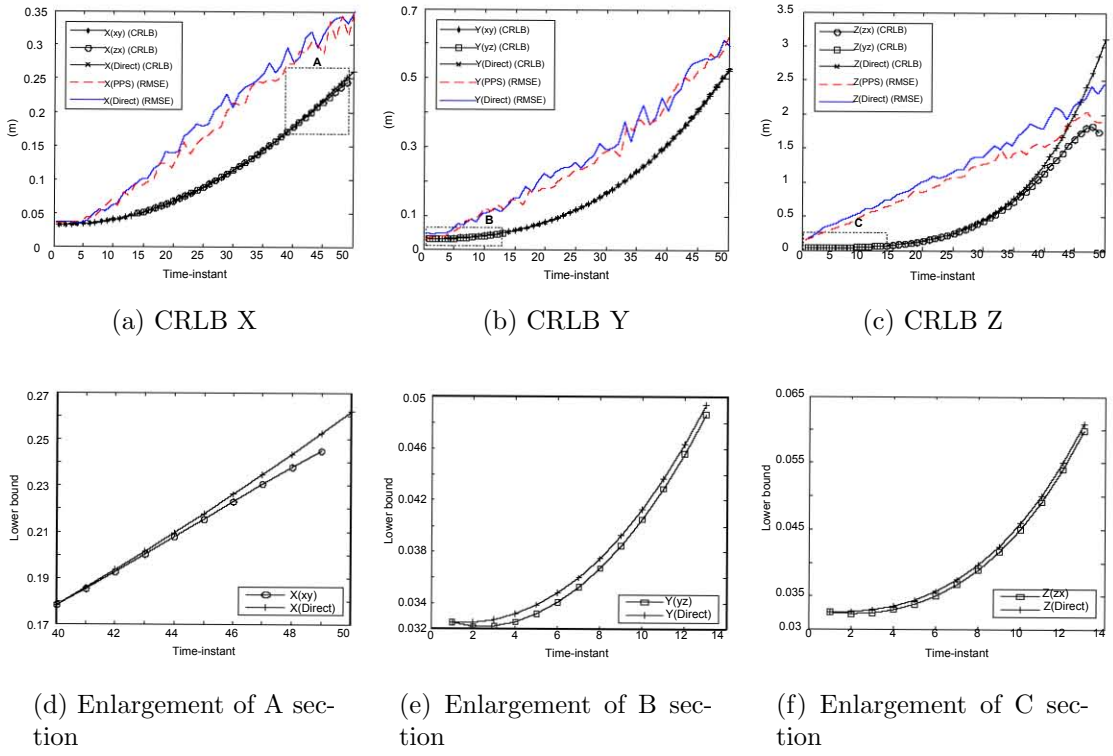
(a) CRLB X (b) CRLB Y (c) CRLB Z

Figure 2-11: Scenario 2: Selected xy- and zx- planes based on PPS shows better performance

2.4.4 Scenario 4

In this scenario, an object is moving as in scenario 3. The sensors, sensor 1 and 2 are placed at ($0m$, $0m$, $0m$) and ($10m$, $10m$, $10m$) respectively. The measured elevation angles ϕ and azimuth angles θ are different with respect to each sensor position as shown in Figure 2-9. During the first 13 time instants, the projected measurement with the lowest variance is selected with yz-plane from sensor 1. In addition, since the measurement with the second lowest variance is with yz-plane from sensor 2, xy-plane from any of two sensors is selected. After the time instant 13, zx-plane from

sensor 1 and yz-plane from sensor 2 are selected.



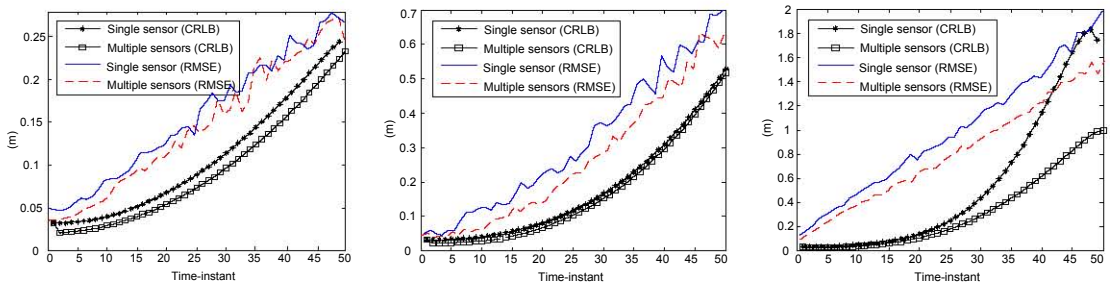
(a) CRLB X (b) CRLB Y (c) CRLB Z

(d) Enlargement of A section (e) Enlargement of B section (f) Enlargement of C section

Figure 2-12: Scenario 3: Since ϕ of the first 13 time instants is measured between 48.24^o and 45.42^o, xy- and yz-planes are selected. In the last 37 time instants, xy- and zx-planes are selected since ϕ is measured between 28.07^o and 44.96^o. For the performance comparison between PPS and direct 3D method, the certain section in CRLB is enlarged (A, B and C)

2.4.5 Result

Figure 2-10 and 2-11 represent the lower bound and RMSE in each direction based on the scenario 1 and 2, respectively. In Figure 2-10, the selection of yz- plane with xy- plane, in Figure 2-11, the selection of zx- plane with xy- plane show the good performance which proves the PPS method is a good estimator. Note that all boundaries are presented for the comparison of other planes selection. In addition,



(a) CRLB X (b) CRLB Y (c) CRLB Z

Figure 2-13: Scenario 4: Multiple sensor and single sensor based estimation with PPS are compared.

single sensor and multiple sensor based estimations are compared in Figure 2-12 and 2-13 which have the same scenario except for the number of sensors. In particular, the multiple sensor based estimation is using the scheduling method finding the best two planes among the multiple sensors. Since the multiple sensors support broader choices for planes selection, the performance is shown to be better comparing single sensor based estimation.

2.5 Conclusions

We have proposed an object tracking algorithm in 3-D space with a passive acoustic sensor. Particle filtering technique used in the 2-D bearings-only tracking problem has been applied to the 3-D space. 3-D space is decomposed into 2-D planes, and by exploiting the fact that the noisy measurements of the acoustic sensor differ on the projected planes, we have proved the effectiveness of the plane selection based on the characteristics. We have shown that the particle filtering with the proposed plane selection is more flexible than the direct 3-D method where the proposed method can

be easily extended to multiple sensor particle filtering. We have also analyzed the performance of the proposed method using the Cramer-Rao Lower Bound (CRLB) and the theoretical lower bound and the simulation results are compared to the direct 3-D method. We have shown that the proposed method outperforms the direct 3-D method.

Chapter 3

Acoustic Sensor Based Multiple Object Tracking with Visual Information Association

3.1 Introduction

Tracking multiple objects has been a great interest to numerous surveillance-required areas applied in diverse fields such as a military, a factory, a hospital and a mining [27] [19]. Among a variety of sensors deployed in a surveillance system, an acoustic sensor is widely used since it allows easy and quick deployment with a less computational complexity as well as a broad sampling range [15] [28]. Acoustic sensor based object tracking is widely studied with several approaches. A time delay estimation method aims at measuring the time delays of arrival signals at receivers [29]. A beamforming method uses a frequency-averaged output power of a steered beamformer [30]. A

bearings-only tracking method aims at estimating position, velocity and possibly some extra features by measuring bearings [31].

Despite the easy deployment of the acoustic sensor, there are several difficult issues when one acoustic sensor tracks multiple objects. Multiple objects and multiple measurements can be incorrectly associated when an acoustic sensor receives the measurements with negligibly small difference [32]. In addition, the number of measurements is varying when the objects do not transmit sound-wave, new objects come into an acoustic sensing range or objects move out an acoustic sensing range. The varying number of measurements gives inconsistent measurement sequences to the acoustic sensor based estimator [33]. Furthermore, when measurements are severely corrupted with the noise or the dynamic models are incorrect, the estimation performance is degraded with a deviated estimation more severely for the multiple object tracking.

In order to overcome the limitations of the acoustic sensor based estimation for multiple objects' tracking, the visual sensor based estimation is combined [34] [35] [36] [37]. In [34], the visual sensor mainly tracks the objects, and the acoustic sensor partially supports the estimation when the tracked objects are occluded. This method is experimentally shown in a video conferencing environment, where persons are seated and speak in a small space. In [35], the acoustic-visual combining method is presented with the iterative decoding algorithm from the theory of turbo codes and factor graphs. This method computes both the likelihood values from the acoustic sensor and the visual sensor, and one of the two data with a higher likelihood is selected for a more accurate estimation. In [36] and [37], two data from acoustic and

visual sensors are simultaneously combined. In [36], a way of jointly processing different sources of information is presented using cooperative Hidden Markov Models (HMMs) with appearance models, whereas in [37], an implicated interaction of lip movements synchronized with acoustic samples is proposed. Our interest is to minimize the resources from visual sensor since the visual sensor based object localization requires much higher computational complexity [38] [39], and the visual sensor is assumed to be deployed for other purpose so the visual sensor cannot dedicate its operation to support one acoustic sensor. We take the approach where the acoustic sensor mainly tracks the objects and the visual sensor cooperation is performed when the acoustic sensor has a difficulty. The timing of the cooperation is determined from the triggering timing by the acoustic sensor.

The acoustic sensor based estimation is performed with bearings-only tracking developed by the sequential Monte Carlo methods known as the particle filter. In the fields of wireless communications, navigation systems, sonar, and robotics applications, the particle filtering is adopted as an emerging powerful tool for solving non-linear and non-Gaussian problems [6] [7] [8] [40]. The particle filters are generally used for an estimation and/or a detection of dynamic system parameters or states in real-time application. While the particle filter with an acoustic sensor tracks multiple objects, the visual sensor detects the objects and localizes their positions when the acoustic sensor triggers for the visual sensor cooperation.

3.2 Background

3.2.1 Object tracking with an acoustic sensor with the multi-model and multi-measurement particle filtering

The acoustic sensor's object tracking is performed with bearings-only measurements. A bearings-only tracking is to estimate object positions and velocities with a sequence of noisy bearing measurements [31] [12]. For an object in tracking, its state at a discrete time k, $k \in \{1, 2, \cdots\}$ is described by

$$\mathbf{x}(k) = \mathbf{F}^{(m(k))}\, \mathbf{x}(k-1) + \mathbf{w}(k-1), \tag{3.1}$$

$$z(k) = H\left(\mathbf{x}(k)\right) + v(k), \tag{3.2}$$

where $\mathbf{x}(k)$ denotes the state vector of the object as $[x(k)\ y(k)\ V^x(k)\ V^y(k)]^T$ and $z(k)$ is the corresponding bearing measurement for the object. $[x(k), y(k)]$ is the 2-dimensional location of the object at time k and $[V^x(k)\ V^y(k)]$ is the x- and y-directional velocity of the object at time k. $H(\mathbf{x})$ is the bearing measurement function for state vector $\mathbf{x}$ as $H\left(\mathbf{x}(k)\right) = \arctan\left(\frac{y(k)}{x(k)}\right)$. The noise random process $\mathbf{w}(k-1)$ and measurement noise $v(k)$ are modeled as zero-mean independent Gaussian. $\mathbf{F}^{(m)}$ is the 4×4 state-transition matrix for model m, $m \in \{1, 2, \cdots, J\}$, where J is the number of the hypothesized models [27] [41], and $m(k)$ represents the model index at time k for the object in tracking. For the object of interest, the model switching is governed by a finite-state Markov chain according to the switching probabilities Prob $[m(k) = v | m(k-1) = u]$ of switching from model u to v, $u, v \in$

$\{1, 2, \cdots, J\}$. Note that this switching probabilities are not needed in the following estimation. As there are multiple measurements, let $\mathbf{z}(k)$ denote a set of measurements as $\left\{z^1(k), z^2(k), \cdots, z^{N(\mathbf{z}(k))}(k)\right\}$, where $z^i(k)$ is the i-th measurement and $N(\mathbf{z}(k))$ is the number of bearing measurements at time k. Also define $\mathbf{z}^i(1:k)$ as the set of measurements up to and including time k as $\{z^i(1), z^i(2), \cdots, z^i(k)\}$, where $i = 1, 2, \cdots, N(\mathbf{z}(k))$. Note that as the unlabeled measurements are received by an acoustic sensor, it is not known which measurement index is corresponding to the object of interest.

The goal of the object tracking is to estimate the state of the object $\mathbf{x}(k)$ and the probability that the object's model index is m at time k for the given history of observations. More specifically, based on the particle filtering,

- Conditional probability density function (pdf) of the object's state $\mathbf{x}(k)$ at time k given the history of observation up to time k; $p\left(\mathbf{x}(k)|\mathbf{z}(1:k)\right)$
- Conditional expected state when the model index is m at time k; $\bar{\mathbf{x}}_m(k)$
- Unconditional probability that the object's model index is m at time k; $\mu_m(k)$

where $m \in \{1, 2, \cdots, J\}$ and $\sum_{m=1}^{J} \mu_m(k) = 1$. Conditional expected means and the probabilities are not directly used for the object tracking but it is used to trigger the visual sensor association. As we use the particle filtering technique for the state estimation, the conditional pdf is estimated with many particles in the state space where each particle is of equal conditional probability density through the sequential importance resampling (SIR) algorithm [18]. L, $L >> 1$, particles are updated for every new observation, and the estimation is done as follow. L resampled particles are

given and they represent the conditional pdf, $p\left(\mathbf{x}(k-1)|\mathbf{z}(1:k-1)\right)$. Then, there is a set of new $N(\mathbf{z}(k))$ measurements $\left\{z^1(k), z^2(k), \cdots, z^{N(\mathbf{z}(k))}(k)\right\}$. From these measurements and the given L particles, we want to obtain

- L resampled particles representing $p\left(\mathbf{x}(k)|\mathbf{z}(1:k)\right)$
- Conditional mean vector, $\bar{\mathbf{x}}_m(k)$ and the unconditional probabilities of the object's model, $\mu_m(k)$, where $m \in \{1, 2, \cdots, J\}$, then eventually the mean vector estimate $\bar{\mathbf{x}}(k)$ as the weighted sum.

The state estimation is done by the interactive multiple model particle filter (IMM-PF) framework [42]. The IMM estimator is a state estimation algorithm for a system represented by Markovian switching model with multiple model indices. In the particle filtering stage at time k, $L \times J$ particles $\hat{\mathbf{x}}_m^{(l)}(k)$, for $l \in \{1, 2, \cdots, L\}$ and $m \in \{1, 2, \cdots, J\}$, are drawn from the previous a posteriori density function $p\left(\mathbf{x}(k-1)|\mathbf{z}(1:k-1)\right)$ for each model m as follow.

$$\hat{\mathbf{x}}_m^{(l)}(k) = \mathbf{F}^{(m)}\tilde{\mathbf{x}}^{(l)}(k-1) + \mathbf{n}_m^{(l)}(k) \text{ for } l \in \{1, 2, \cdots, L\} \text{ and } m \in \{1, 2, \cdots, J\} \quad (3.3)$$

where $\tilde{\mathbf{x}}^{(l)}(k-1)$ is the resampled particles at time $k-1$ and $\mathbf{n}_m^{(l)}(k)$'s are identically distributed independent Gaussian zero-mean noise. The predicted bearing measurements to particles $\hat{\mathbf{x}}_m^{(l)}(k)$'s are obtained as

$$\hat{z}_m^{(l)}(k|k-1) = H(\hat{\mathbf{x}}_m^{(l)}(k)) = \arctan\left(\frac{\hat{y}_m^{(l)}(k)}{\hat{x}_m^{(l)}(k)}\right), \quad (3.4)$$

for $l \in \{1, 2, \cdots, L\}$ and $m \in \{1, 2, \cdots, J\}$, where $\left(\hat{y}_m^{(l)}(k), \hat{x}_m^{(l)}(k)\right)$ is the l^{th} particle's 2-dimensional position of $\hat{\mathbf{x}}_m^{(l)}(k)$ with model m. Note that there are $L \times J$ predicted measurements for the object of interest. These $L \times J$ predicted measurements lead to the weight evaluation from the set of actual measurements $\mathbf{z}(k)$ as

$$\bar{w}_m^{i,(l)}(k) = d\left(z^i(k) - \hat{z}_m^{(l)}(k|k-1)\right), \tag{3.5}$$

for $i \in \{1, 2, \cdots, N(\mathbf{z}(k))\}$, $l \in \{1, 2, \cdots, L\}$ and $m \in \{1, 2, \cdots, J\}$, where $d(\cdot)$ is the particle weight evaluation function from the Gaussian probability density function [8] [43]. Since each particle $\hat{\mathbf{x}}_m^{(l)}(k)$ is assigned with $N(\mathbf{z}(k))$ weights, there are $L \times J \times N(\mathbf{z}(k))$ weights. $\bar{w}_m^{i,(l)}(k)$ denotes the (unnormalized) weight of the l^{th} particle in model m for given measurement $z^i(k)$. These $L \times J \times N(\mathbf{z}(k))$ weights are normalized as follow,

$$w_m^{i,(l)}(k) = \frac{\bar{w}_m^{i,(l)}(k)}{\sum_{i'=1}^{N(\mathbf{z}(k))} \sum_{m'=1}^{J} \sum_{l'=1}^{L} \bar{w}_{m'}^{i',(l')}(k)}, \tag{3.6}$$

for $i \in \{1, 2, \cdots, N(\mathbf{z}(k))\}$, $l \in \{1, 2, \cdots, L\}$ and $m \in \{1, 2, \cdots, J\}$. The SIR algorithm is used to obtain $\tilde{\mathbf{x}}^{(l)}(k)$'s, $l \in \{1, 2, \cdots, L\}$ with the equal conditional probability density from $\hat{\mathbf{x}}_m^{(l)}(k)$ particles with $w_m^{i,(l)}(k)$ weights. Note that there are $L \times J$ particles and each particle has $N(\mathbf{z}(k))$ weight values. However, in order to apply the SIR algorithm, each particle has to have only one weight. Each particle is identically copied $N(\mathbf{z}(k))$ times to have the same number of weights, then the SIR algorithm is applied as in Figrue 3-1, where $L \times J \times N(\mathbf{z}(k))$ particles are transformed to L resampled particles. Each circle in Figrue 3-1 illustrates the weight of the particle. The

resampled particles are assigned with an equal weight of $1/L$. The particles distribution with the resampled particles $\tilde{\mathbf{x}}^{(l)}(k)$ with each corresponding weight value $1/L$ represents the conditional pdf of $p(\mathbf{x}(k)|\mathbf{z}(1:k))$. The resampled particles, $\tilde{\mathbf{x}}^{(l)}(k)$, are used for generating particles $\hat{\mathbf{x}}_m^{(l)}(k+1)$ as in (3.3) for time $k+1$.

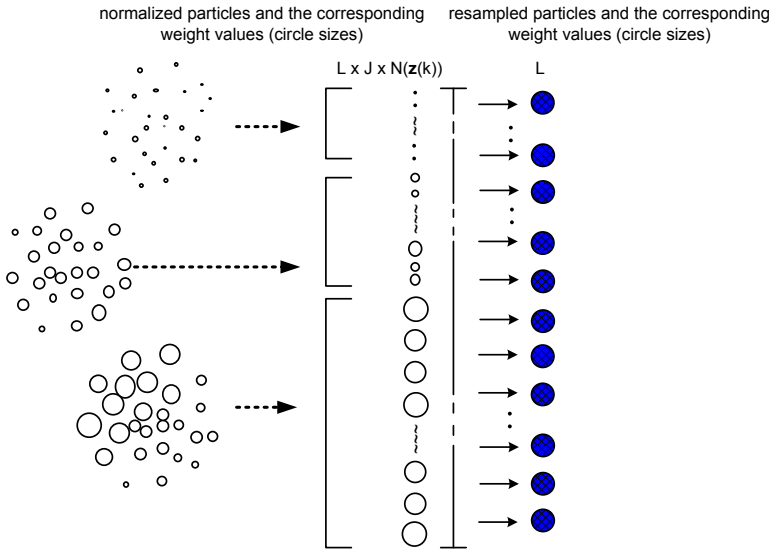


Figure 3-1: Resampling of $L \times J \times N(\mathbf{z}(k))$ particles to L particles

In order to estimate the final estimated state vector denoted as $\bar{\mathbf{x}}(k)$, the joint probability density association (JPDA) method is used which makes use of all $L \times J \times N(\mathbf{z}(k))$ particles. $\bar{\mathbf{x}}(k)$ can also be obtained from the resampled L particles, but using the original $L \times J \times N(\mathbf{z}(k))$ particles can give a better mean estimate of the state. The JPDA technique uses a weighted average of all the measurements falling inside an object track's validation region to update the object state [44]. In addition, the weighted average of all possible J models is also applied for estimating $\bar{\mathbf{x}}(k)$. First, $\bar{\mathbf{x}}_m^i(k)$'s, $i \in \{1, 2, \cdots, N(\mathbf{z}(k))\}$, the conditional means of the state given

each measurement $z^i(k)$ over the particles set, $\hat{\mathbf{x}}_m^{(l)}(k)$'s of model m is obtained as

$$\bar{\mathbf{x}}_m^i(k) = \sum_{l=1}^{L} \hat{\mathbf{x}}_m^{(l)}(k) \cdot w_m^{i,(l)}(k). \tag{3.7}$$

Then, $\bar{\mathbf{x}}_m(k)$'s, $m \in \{1, 2, \cdots, J\}$, the conditional means of the state for model m is obtained as

$$\bar{\mathbf{x}}_m(k) = \sum_{i=1}^{N(\mathbf{z}(k))} \bar{\mathbf{x}}_m^i(k) \cdot \mu_m^i(k), \tag{3.8}$$

where $\mu_m^i(k)$ represents the probability that the model index is m given the measurement $z^i(k)$, and it is obtained as

$$\mu_m^i(k) = \frac{\sum_{l=1}^{L} w_m^{i,(l)}(k)}{\sum_{m=1}^{J}(\sum_{l=1}^{L} w_m^{i,(l)}(k))}. \tag{3.9}$$

Finally, the mean state vector estimate $\bar{\mathbf{x}}(k)$ is obtained as

$$\bar{\mathbf{x}}(k) = \sum_{m=1}^{J} \bar{\mathbf{x}}_m(k) \cdot \mu_m(k), \tag{3.10}$$

where $\mu_m(k)$ is the probability that the object's model index is m, and it is obtained as

$$\mu_m(k) = \frac{\sum_{i=1}^{N(\mathbf{z}(k))} \mu_m^i(k)}{\sum_{m=1}^{J}(\sum_{i=1}^{N(\mathbf{z}(k))} \mu_m^i(k))}. \tag{3.11}$$

3.2.2 Object tracking with a visual sensor

In our application, once an acoustic sensor triggers for visual sensor cooperation, the visual sensor performs the object localization and supports the acoustic sensor

with the localized position. The visual sensor localizes the object positions with the parallel projection model which supports zooming, panning and tilting of the visual sensor [45] [16], and simplifies the computational complexity in determining the object positions with automatically focusing on the objects. As the visual sensor cooperation is triggered, a pair of visual sensors simultaneously detect, identify and localize the multiple objects as shown in Figrue 3-2. The objects are detected with motion analysis and color information as shown in [47] [48]. We assume that the viewable range of the visual sensors and the measurable range of the acoustic sensor are overlapped so that the visual sensors support the localized positions of the objects moving within the measurable range of acoustic sensors.

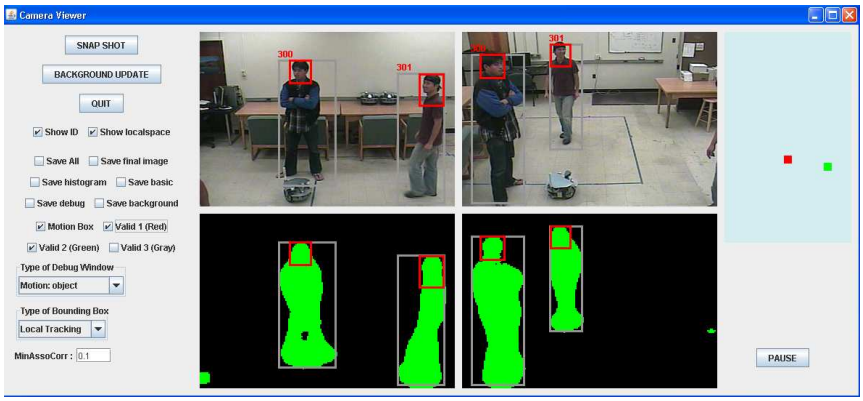


Figure 3-2: Visual sensors based tracking demo: as visual sensors cooperation is triggered, two visual sensors simultaneously detect, identify and localize multiple objects.

Let $(x^{[v]}(k), y^{[v]}(k))$ denote the visually localized position of the triggered object at time k. Then, if the cooperation is performed at time k, the final estimated state vector $\bar{\mathbf{x}}(k) = [\bar{x}(k)\ \bar{y}(k)\ \bar{V}^x(k)\ \bar{V}^y(k)]^T$ of the object of interest is replaced by $[x^{[v]}(k)\ y^{[v]}(k)\ \bar{V}^x(k)\ \bar{y}^y(k)]$. Figure 3-3 illustrates the simplified acoustic sensor

based IMM-PF data flow incorporated with the visual sensor cooperation where the triggering conditions can be from measurements and/or estimated results with the particle filtering.

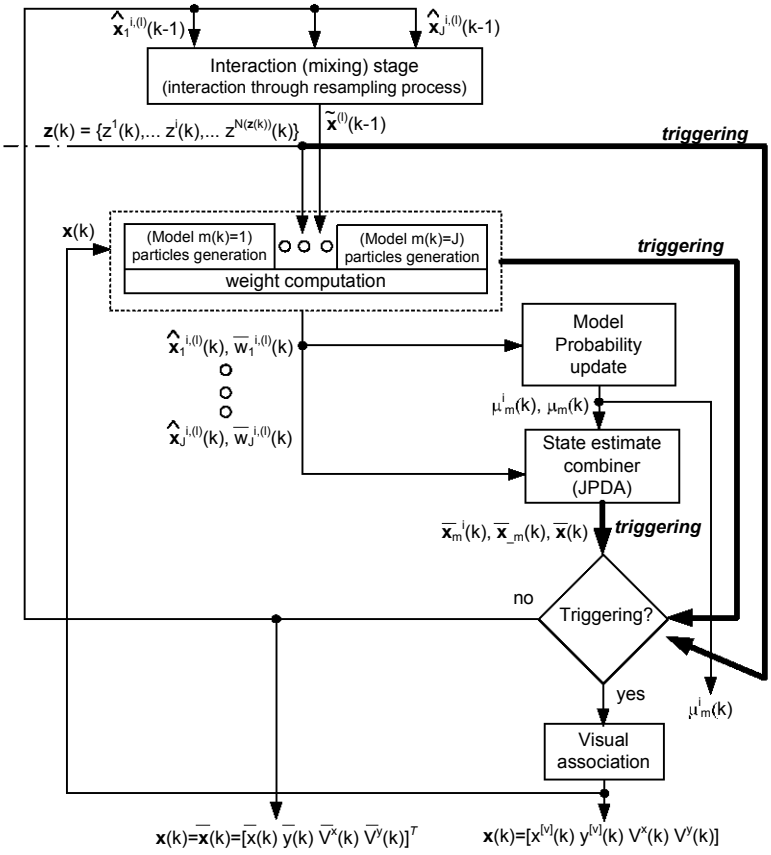


Figure 3-3: IMM-PF data flow incorporated with the visual sensor cooperation

3.3 Effect of Visual Sensor Cooperation

In this Chapter, the triggering conditions of the visual sensor cooperation are discussed. As a simple case, unconditional periodic triggering is discussed in Chapter 3.3.1, and we show that additional triggering conditions are needed unless the cooperation period is sufficiently small. The acoustic sensor based estimation can have

difficulties from two different perspectives – the system dynamics and the estimation performance. These two issues can be considered as two different triggering conditions. Firstly, due to the system dynamics, the number of tracked objects and the number of measurements in the acoustic sensor can be different. If so, the acoustic sensor cannot track multiple objects correctly, and the support from the visual sensor is needed. There can be several cases for the system dynamics and they are discussed in Chapter 3.3.2. The performance degradation of the object tracking by the acoustic sensor, in our application the particle filter's performance, can be overcome by the support from the visual sensor even when the number of tracked objects and the number of measurements are the same. In this case, the performance of the estimation can be a condition for the triggering and they are discussed in Chapter 3.3.3. Performance improvement by having the two triggering conditions is presented by the simulation in Chapter 3.3.4.

3.3.1 Periodic Visual Sensor Cooperation

Suppose that the visual sensor periodically localizes the object positions and supports the acoustic sensor based estimation every visual sampling time T_v. In order to verify the effect of the periodic visual sensor cooperation, the tracking environment with three objects and an acoustic sensor are used as follow.

- Objects O^1, O^2 and O^3 are initially positioned at $(50m, 30m)$, $(35m, 50m)$ and $(45m, 45m)$, respectively. Trajectories of the three objects are shown in Figrue 3-4(a).

- Each object trajectory is sampled by 200 acoustic bearing data.

- Three models are considered – constant velocity $\mathbf{F}^{(1)}$, clockwise coordinated turn $\mathbf{F}^{(2)}$ and anticlockwise coordinated turn $\mathbf{F}^{(3)}$ with manoeuvre rotation acceleration $0.01m/s^2$ [49]. They are

$$\mathbf{F}^{(1)} = \begin{pmatrix} 1 & 0 & T_s & 0 \\ 0 & 1 & 0 & T_s \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \ \mathbf{F}^{(p)} = \begin{pmatrix} 1 & 0 & \frac{\sin(\Re_k^{(p)} T_s)}{\Re_k^{(p)}} & -\frac{(1-\cos(\Re_k^{(p)} T_s))}{\Re_k^{(p)}} \\ 0 & 1 & \frac{(1-\cos(\Re_k^{(p)} T_s))}{\Re_k^{(p)}} & \frac{\sin(\Re_k^{(p)} T_s)}{\Re_k^{(p)}} \\ 0 & 0 & \cos(\Re_k^{(p)} T_s) & -\sin(\Re_k^{(p)} T_s) \\ 0 & 0 & \sin(\Re_k^{(p)} T_s) & \cos(\Re_k^{(p)} T_s) \end{pmatrix}, \tag{3.12}$$

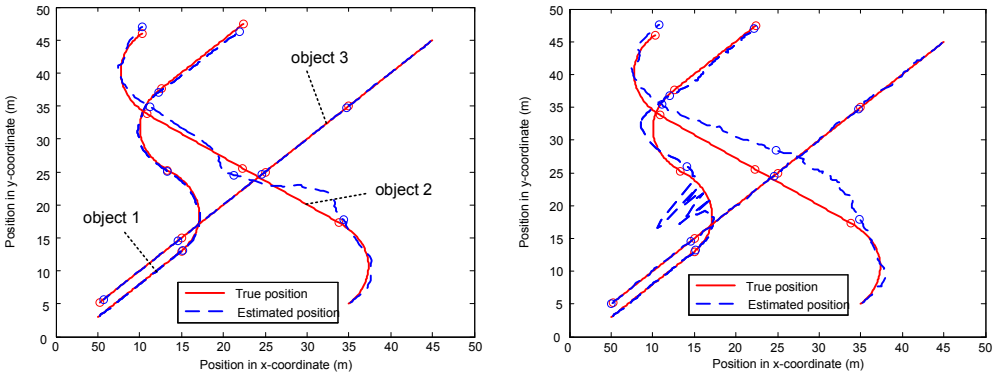
where p = 2,3 and $\Re_k^{(p)}$ is the model-dependent turning rates expressed as

$$\begin{aligned} \Re_k^{(2)} &= \frac{\alpha}{\sqrt{\left(V^x(k-1)\right)^2 + \left(V^y(k-1)\right)^2}}, \\ \Re_k^{(3)} &= \frac{-\alpha}{\sqrt{\left(V^x(k-1)\right)^2 + \left(V^y(k-1)\right)^2}}, \end{aligned} \tag{3.13}$$

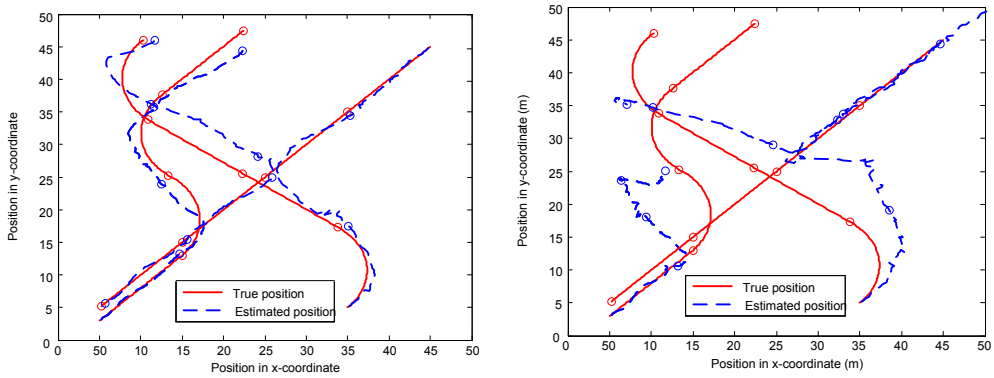
with α being the factor determining the rotation degree as $1m/s^2$.

- Measurement noise variance σ^2 varies from 0.0 to 5.0.

Figrue 3-4 shows the performance of the acoustic sensor based estimation for various noise variances, where 500 particles are used for each object in each model. As the noise variance increases, the estimation has a higher Root-Mean-Square (RMS) position error. Especially with the noise variance of 5.0, the RMS position error of each object is 3.98, 9.80 and 1.08, respectively. Under the same condition with



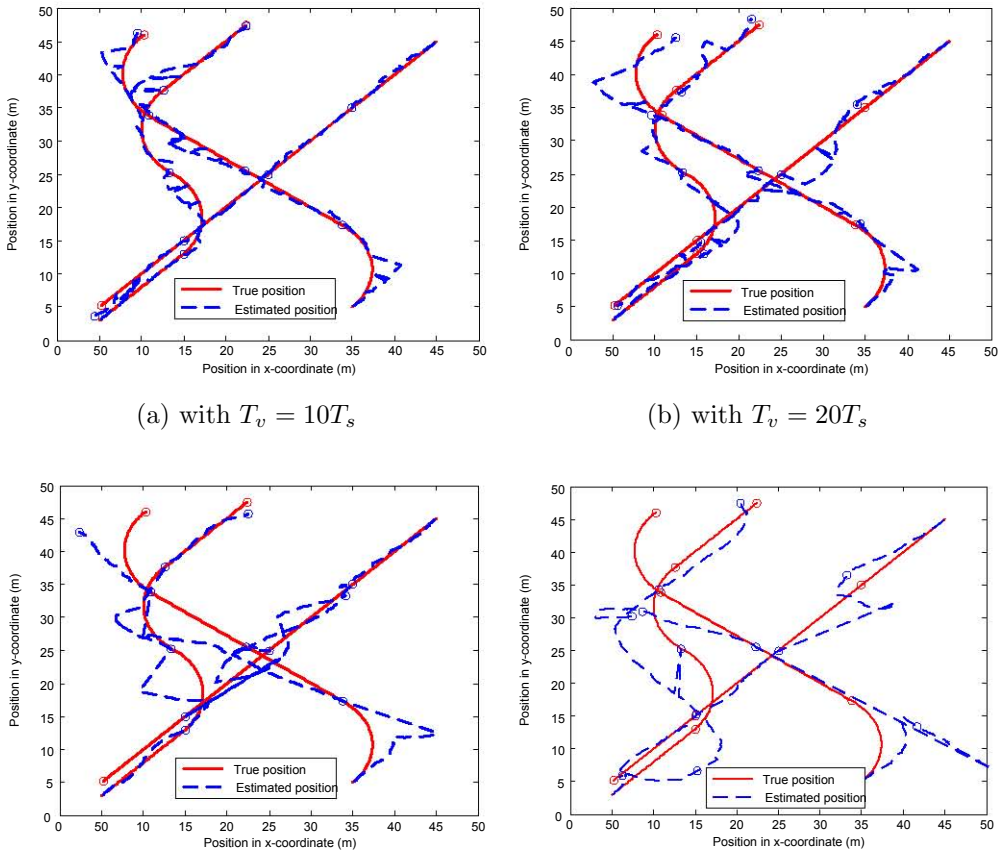
(a) with $\sigma^2 = 0$ (b) with $\sigma^2 = 0.5$



(c) with $\sigma^2 = 1.5$ (d) with $\sigma^2 = 5.0$

Figure 3-4: The estimation with an acoustic sensor only is shown according to different measurement noise variance σ^2: 0, 0.5, 1.5, 5.0.

the noise variance of 5.0 in Figrue3-4(d), the visual sensor periodically supports the acoustic sensor based estimation by updating the localized object position for each object. The effect of the periodic visual sensor supports with different sampling time T_v is shown in Figrue 3-5 and 3-6. In Figrue 3-5, the estimated trajectories are shown for different visual sensor's sampling times T_v: $10T_s$, $20T_s$, $50T_s$ and $100T_s$. Figrue 3-6 shows the average RMS position errors with visual sensor's sampling time T_v from $1T_s$ to $100T_s$ through 1,000 time trials respectively.



(a) with $T_v = 10T_s$ (b) with $T_v = 20T_s$

(c) with $T_v = 50T_s$ (d) with $T_v = 100T_s$

Figure 3-5: Visual sensor cooperation performance is shown according to periodic cooperation with T_v: $10T_s$, $30T_s$, $50T_s$, $100T_s$ based on the result with measurement variance 5 in Figrue 3-4(d) (500 particles are used in the simulation).

From the results shown in Figrue 3-5 and 3-6, it is difficult to find an optimal visual sensor's sampling time T_v. It can only be seen that the estimated object position becomes more accurate as the visual sensor's sampling time T_v is close to the acoustic sampling time T_s. Even when the acoustic sensor estimates an object's position close to the true position, the visual sensor may unnecessarily support the acoustic sensor through the periodic cooperation. In order to efficiently use the precious visual sensor cooperation, it has be triggered only when the cooperation is necessary.

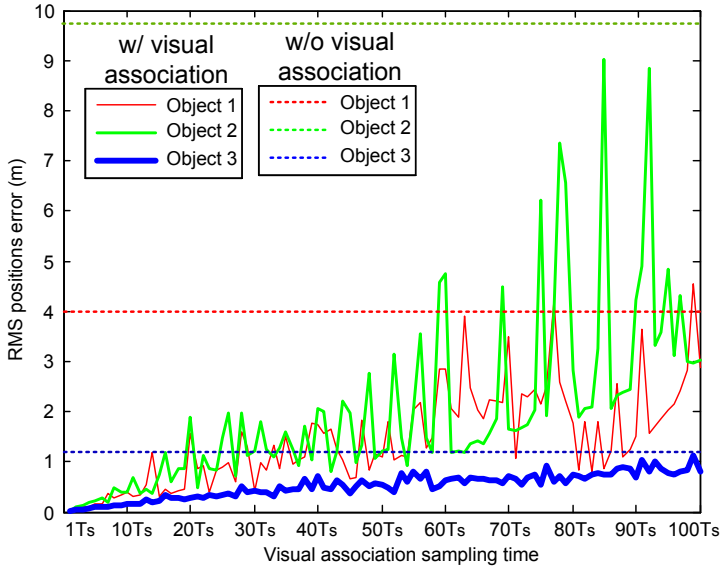


Figure 3-6: RMS position error is shown according to periodic cooperation with T_v: $1T_s$ to $100T_s$. (500 particles are used in the simulation)

Furthermore, the periodic cooperation does not efficiently support the acoustic sensor based estimation against deviated estimation, measurement resolution problem and a varying number of objects. These issues are discussed in the following subchapters.

3.3.2 Triggering based on System Dynamics

An acoustic sensor can have a difficulty in measuring multiple measurements when their difference is negligibly small – the acoustic sensor has a limited resolution of $\Delta z_{\text{critical}}$ [15] [50]. The bearing measurement difference of two objects less than $\Delta z_{\text{critical}}$ can cause an acoustic sensor to recognize only one sound wave by merging the incoming sound wave. Let $I(k)$ denote the number of objects estimated by the acoustic sensor at time k. Then, if the acoustic sensor cannot differentiate the objects, the number of measurements at time k, $N(\mathbf{z}(k))$ and the number of estimated

objects at time $k-1$ become unequal as

$$N(\mathbf{z}(k)) \neq I(k-1). \tag{3.14}$$

The visual sensor cooperation should be triggered in case of (3.14). Once the visual sensor supports the acoustic sensor based estimator with the visually localized positions at time k, the number of estimated objects $I(k)$ is updated and verified with $N(\mathbf{z}(k+1))$ for time $k+1$.

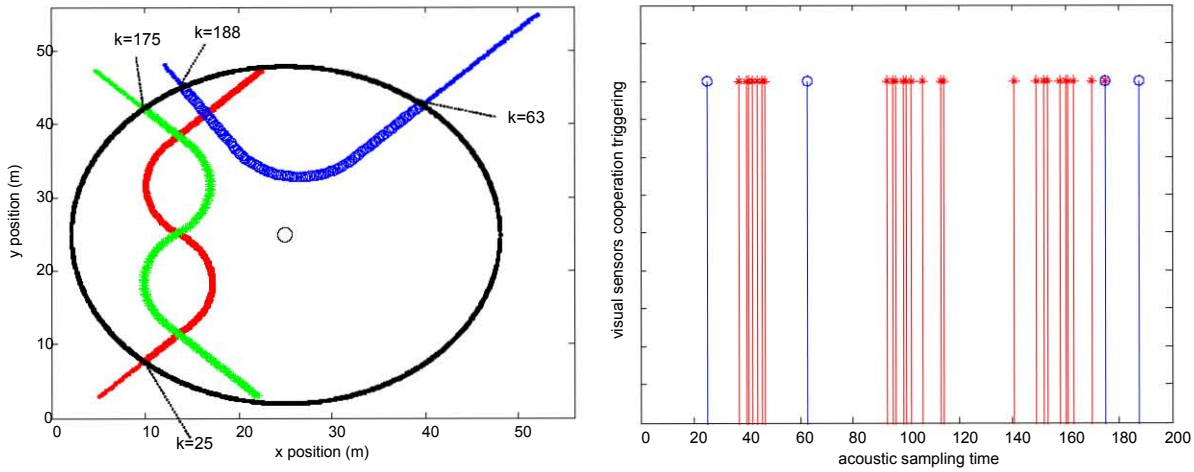
Together with the measurement resolution problem, an acoustic sensor also has a difficulty in estimating the state with a varying number of objects/measurements positioned within the measurable range of the acoustic sensor. The number of measurements is varying when the objects do not transmit sound-wave, new objects come into an acoustic sensing range or objects move out an acoustic sensing range. Then, similarly to the measurement resolution problem, the number of measurements at time k and the number of estimated objects at time $k-1$ become unequal as in (3.14). More specifically in the varying number of objects/measurements, if $N(\mathbf{z}(k)) < I(k-1)$, objects move out of acoustic sensing range, or/and an acoustic sensor does not receive bearing measurements from objects at time k. On the other hand, if $N(\mathbf{z}(k)) > I(k-1)$, new objects are moving into the acoustic sensing range at time k. That is, the varying number of objects/measurements also can be triggered for the visual sensor cooperation with the same condition of (3.14). After the visual sensor cooperation, $I(k)$ is updated and verified with $N(\mathbf{z}(k+1))$ for time $k+1$.

Consider the environment shown in Figrue 3-7(a) where the acoustic sensor is posi-

tioned at $(25m, 25m)$ while the bearing sources are sampled every 1 second during 200 second period with the noise variance of 3. Object 1 and object 2 starts from $(5m, 3m)$ and $(22m, 3m)$ with initial velocities of $(0.2m/s, 0.2m/s)$ and $(-0.2m/s, 0.2m/s)$, respectively. Two objects are with model $\mathbf{F}^{(1)}$ except at time $k = 51T_s$, $101T_s$ and $151T_s$. Their models at those times are $\mathbf{F}^{(2)}$ or $\mathbf{F}^{(3)}$ defined in equation (3.12), and the resulting trajectories are shown in Figrue 3-7(a). Object O^1 is moving into the acoustic sensing range at time $25T_s$ and object O^2 is moving out at time $175T_s$. The new object O^3, is moving in the acoustic sensing range at time $63T_s$ and moving out at time $188T_s$. Object O^3 is initially with model $\mathbf{F}^{(1)}$, and it changes to $\mathbf{F}^{(2)}$ and returns to $\mathbf{F}^{(1)}$ at time $101T_s$ and $151T_s$, respectively. Figrue 3-7(b) shows the triggering timings based on the system dynamics including the measurement resolution problem and the varying number of objects. For better understanding, 'o' is marked when the triggering timing is caused by the varying number of objects while '$*$' is marked when it is caused by the measurement resolution problem.

3.3.3 Triggering based on Estimation Performance

The triggering based on (3.14) cannot trigger the visual sensor association for a simultaneous varying number of objects or measurements. Figrue 3-8 shows several examples. Given the three objects in Figrue 3-8 (a), Figrue 3-8 (b) shows that a new object, O^4 moves into the acoustic sensing range while O^3 bearing measurement is not received by an acoustic sensor. In this case, the number of objects $I(k-1)$ and the number of measurements $N(\mathbf{z}(k))$ are the same even though the number of objects



(a) The 23-meter radius acoustic sensing range and object trajectories

(b) Triggering timings where 'o' is by the varying number of objects and '$*$' is by the measurements resolution problem.

Figure 3-7: The triggering timings based on system dynamics

is varying and the association of a visual sensor is needed. Similarly, the condition in equation (3.14) does not trigger an association either for the case in Figrue 3-8 (c), where the new object, O^4 moves into the acoustic sensing range while O^3 moves out the acoustic sensing range. Figrue 3-8 (d), (e) and (f) also illustrate similar cases, where the association is not triggered despite the need. The cases in Figrue 3-8 (b) through 3-8 (f) should trigger the visual sensor association by considering the estimation performance at the particle filtering state.

The triggering based on the estimation performance is to find the triggering timing with the deviated estimation at the particle filtering stage while the triggering based on the system dynamics is to find the triggering timing with the inconsistency between $I(k)$ and $N(\mathbf{z}(k))$. The deviated estimation is caused by the cases in (3-8) or an incorrect interaction between the measurement and the predicted particles. It is non-

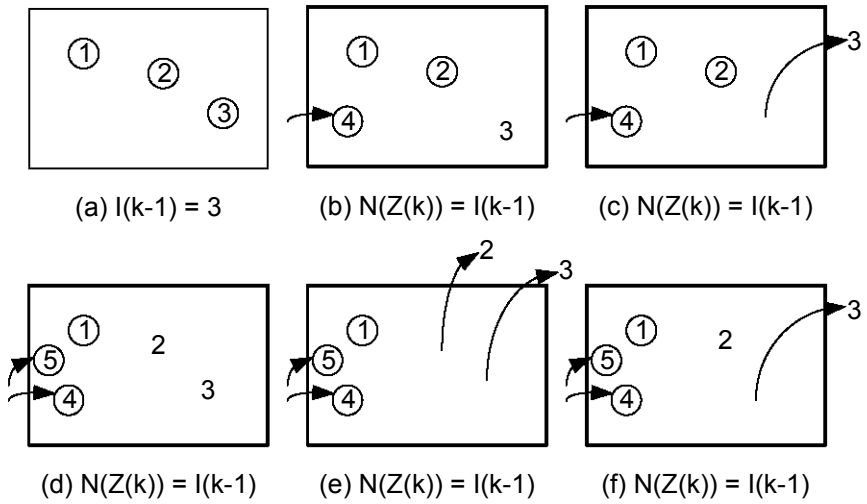


Figure 3-8: Examples when the triggering based on (3.14) does not work.

trivial to evaluate how the estimated position is deviated from a true object position because an acoustic sensor receives only the bearing measurements, and the triggering should be based on the difference between the angle from the estimated position and the bearing measurement. Let $(\bar{x}(k), \bar{y}(k))$ be the estimated position of an object and a bearing measurement $z^i(k),\ i \in \{1, 2, \cdots, N(\mathbf{z}(k))\}$ with noise variance σ^2 are given as illustrated in Figrue 3-9. Assuming that the bearing measurement $z^i(k)$ follows the Gaussian distribution, its range between $z^i(k)-2\sigma$ and $z^i(k)+2\sigma$ contains 95% (2σ confidence) of the true bearing. Then, the estimated position $(\bar{x}(k), \bar{y}(k))$ is considered as a deviation if the following condition is satisfied,

$$\arctan\left(\frac{\bar{y}(k)}{\bar{x}(k)}\right) < z^i(k)-2\sigma \text{ or } \arctan\left(\frac{\bar{y}(k)}{\bar{x}(k)}\right) > z^i(k)+2\sigma, \quad \forall i,\ i \in \{1, 2, \cdots, N(\mathbf{z}(k))\}. \tag{3.15}$$

This means that if no bearing measurement falls within $\pm\ 2\sigma$ of the estimated angle, the visual sensor is triggered for the cooperation. Note that the measurement variance

σ^2 is known from the acoustic sensor's performance characteristics.

The 95 % confidence true bearing range plays an important role to evaluate the deviated estimation, especially for estimating multiple object states with multiple models; $I(k) > 1$ and $J > 1$. Consider the estimation with multiple objects and two models. Figrue 3-10 illustrates a deviated estimation example with simplified sequential steps from particles generation to object state estimation. In Figrue 3-10(a), two-model based particles $\hat{\mathbf{x}}_1^{(1:L)}(k)$ and $\hat{\mathbf{x}}_2^{(1:L)}(k)$ are generated for an object, and the unlabeled measurements $z^1(k)$ and $z^2(k)$ are updated. Suppose that measurement $z^1(k)$ is obtained from the object of interest while measurement $z^2(k)$ is obtained from another object. Suppose also that $\hat{\mathbf{x}}_1^{(1:L)}(k)$ is generated close to $z^1(k)$ and $\hat{\mathbf{x}}_2^{(1:L)}(k)$ is generated close to $z^2(k)$. Then, in Figrue 3-10(b), particles' weights for model 1 given measurement $z^1(k)$, $\bar{w}_1^{1,(1:L)}$ and particles' weights for model 2 given the measurement $z^2(k)$, $\bar{w}_2^{2,(1:L)}$ are evenly dominating for the particles $\hat{\mathbf{x}}_1^{(1:L)}(k)$ and $\hat{\mathbf{x}}_2^{(1:L)}(k)$, respectively. According to the weights, the estimated object state $\bar{\mathbf{x}}(k)$ is obtained with the average of each model based particles information. Finally, the bearing of the estimated position $\arctan\left(\frac{\bar{y}(k)}{\bar{x}(k)}\right)$ strays off from the 95 % confidence true bearing range of $z^1(k)$ as illustrated in Figrue 3-10(c).

However, the 95 % confidence true bearing range as in (3.15) does not necessarily trigger the visual sensor cooperation. Figrue 3-11 illustrates another deviated estimation example, where the visual sensor cooperation cannot be triggered with the 95 % confidence true bearing range from the condition in (3.15). Similarly to the example in Figrue 3-10, suppose that measurement $z^1(k)$ is obtained from the object of interest while measurement $z^2(k)$ is obtained from another object. In Figrue 3-

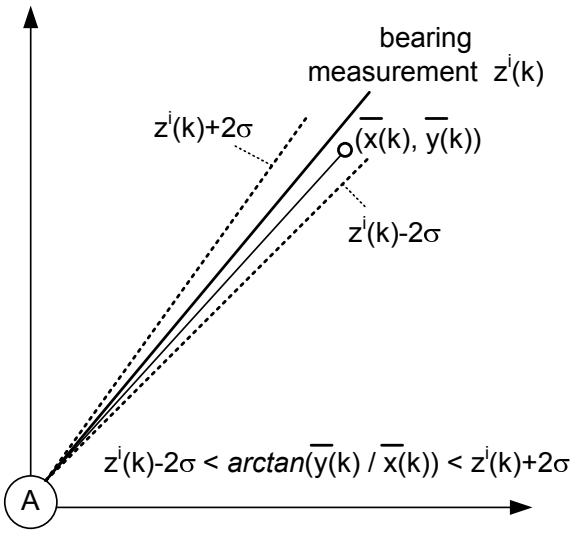
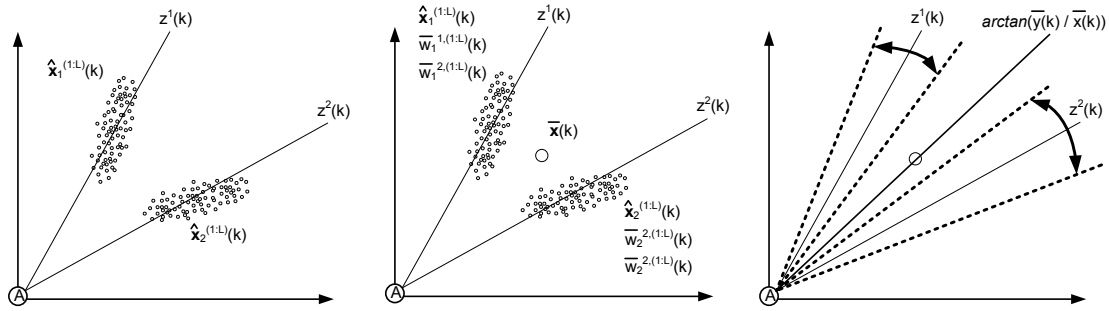


Figure 3-9: 95 % confidence true bearing ranges based triggering method



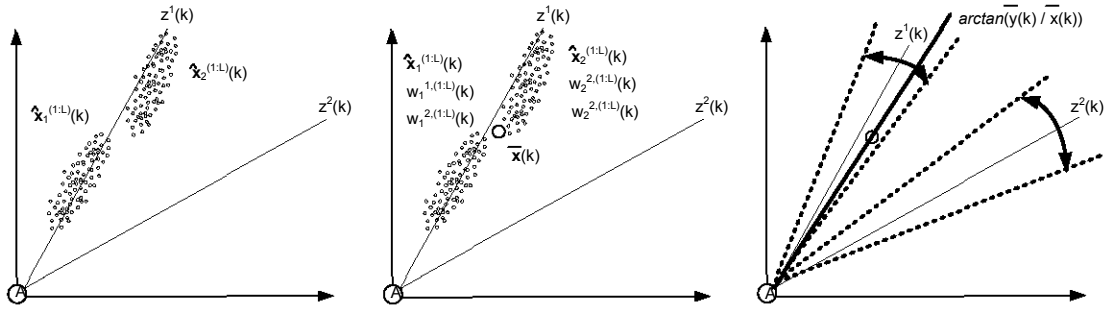
(a) Particles generation and unlabeled measurements update

(b) Particles weights evaluation and object state estimation

(c) 95 % confidence true bearing ranges and the bearing of the estimated position

Figure 3-10: Deviated estimation example with multiple models and multiple measurements

11(a), two-model based particles $\hat{\mathbf{x}}_1^{(1:L)}(k)$ and $\hat{\mathbf{x}}_2^{(1:L)}(k)$ for the object of interest are generated and both of them are at the angle close to $z^1(k)$. Then, in Figrue 3-11(b), particles' weights for model 1 given measurement $z^1(k)$ and particles' weights for the model 2 given the measurement $z^1(k)$, $\bar{w}_1^{1,(1:L)}$ and $\bar{w}_2^{1,(1:L)}$ are evenly dominating for the particles $\hat{\mathbf{x}}_1^{(1:L)}(k)$ and $\hat{\mathbf{x}}_2^{(1:L)}(k)$, respectively. According to the weights, the esti-



(a) Particles weights evaluation and object state estimation

(b) Particles weights evaluation and object state estimation

(c) 95 % confidence true bearing ranges and the bearing of the estimated position

Figure 3-11: Deviated estimation example where the triggering condition in (3.15) is not enough

mated object state $\bar{\mathbf{x}}(k)$ is obtained with the average of each model based particles. As illustrated in Figrue 3-11(c), even though the estimated object state $\bar{\mathbf{x}}(k)$ is deviated by the two models, the bearing of the estimated position $\arctan\left(\frac{\bar{y}(k)}{\bar{x}(k)}\right)$ does not trigger the visual sensor cooperation from the condition in (3.15).

In order to overcome the limitation of the triggering with the 95 % confidence true bearing range in (3.15), we consider an additional triggering condition based on predicted particles distribution. The particle distribution can be expressed with an ellipse representing the region, which contains 95% (2σ confidence) of the particles assuming that they are Gaussian distributed [51] in two dimension. Denote the 95% confidence ellipse of $\hat{\mathbf{x}}_j^{(1:L)}(k)$ as $\mathbf{D}_j(k)$, where j, $j \in \{1, 2, \cdots, M\}$, represents the model index. Figrue 3-12 illustrates the 95% confidence particles ellipses $\mathbf{D}_1(k)$ and $\mathbf{D}_2(k)$ corresponding to $\hat{\mathbf{x}}_1^{(1:L)}(k)$ and $\hat{\mathbf{x}}_2^{(1:L)}(k)$ in the deviated estimation example in Figrue 3-11. If the estimated position $(\bar{x}(k), \bar{y}(k))$ is obtained outside the 95%

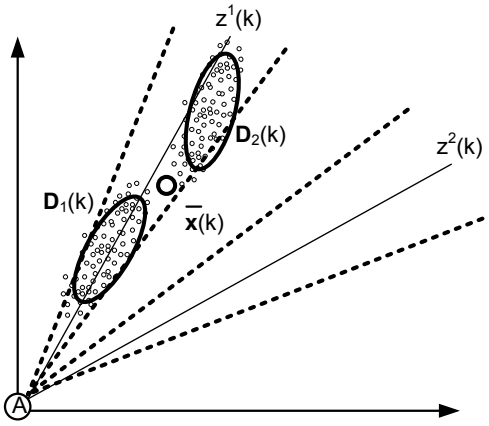
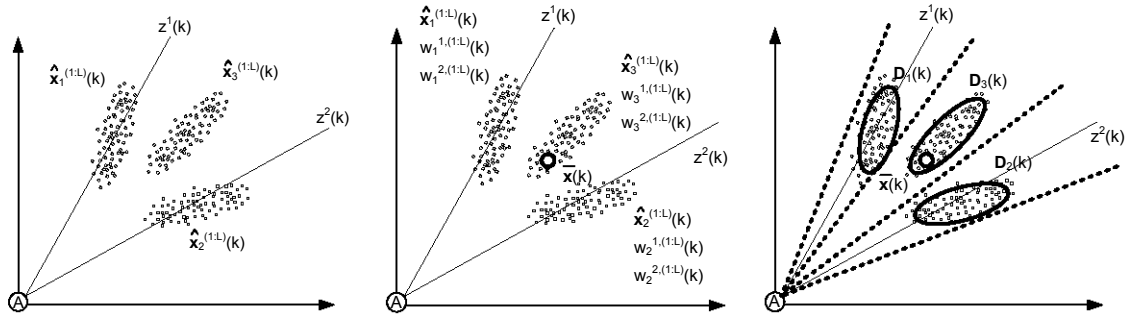


Figure 3-12: Particle distribution containing 95% (2σ confidence) of the particles assuming they are Gaussian distributed

confidence predicted particles ellipse as in Figrue 3-11, then it is considered as a deviation. In a general form, the estimated position $(\bar{x}(k), \bar{y}(k))$ is considered as a deviation with the condition as

$$(\bar{x}(k), \bar{y}(k)) \notin \mathbf{D}_j(k), \quad \forall j \in \{1, 2, \cdots, M\}. \tag{3.16}$$

Even though the 95% confidence particles ellipses in the condition (3.16) is to overcome the limitation of the triggering with the 95 % confidence true bearing range in the condition (3.15), these two conditions should be used together – at least one condition indicates a deviation then the association should be triggered. Figrue 3-13 illustrates another deviated example, where the visual sensor cooperation are triggered not by (3.16) but from (3.15). Also, suppose that measurement $z^1(k)$ is obtained from the object of interest while measurement $z^2(k)$ is obtained from another object. In Figrue 3-13(a), three-model based particles $\hat{\mathbf{x}}_1^{(1:L)}(k)$, $\hat{\mathbf{x}}_2^{(1:L)}(k)$ and $\hat{\mathbf{x}}_3^{(1:L)}(k)$ for the object of interest are generated as $\hat{\mathbf{x}}_1^{(1:L)}(k)$ are generated close to $z^1(k)$ and $\hat{\mathbf{x}}_2^{(1:L)}(k)$



(a) Particles weights evaluation and object state estimation

(b) Particles weights evaluation and object state estimation

(c) 95 % confidence true bearing ranges and the bearing of the estimated position

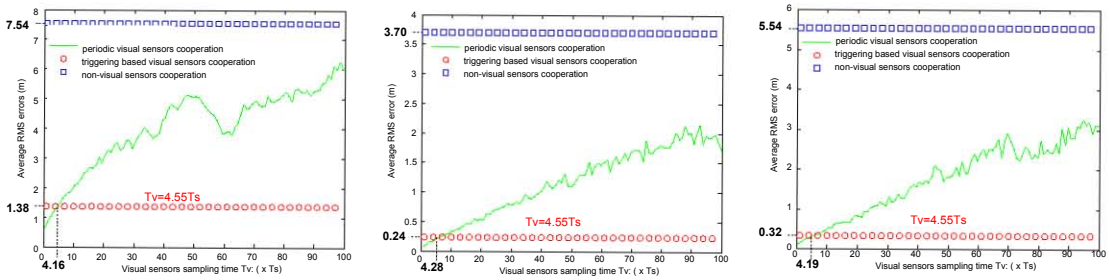
Figure 3-13: Deviated estimation example where both conditions (3.15) and (3.16) should be considered for the visual sensor association.

are generated close to $z^2(k)$. Then, as illustrated in Figrue 3-13(b), each model based particles' weights $\bar{w}_1^{1,(1:L)}$, $\bar{w}_1^{2,(1:L)}$, $\bar{w}_2^{1,(1:L)}$, $\bar{w}_2^{2,(1:L)}$, $\bar{w}_3^{1,(1:L)}$ and $\bar{w}_3^{2,(1:L)}$ are evaluated corresponding to the unlabeled measurements $z^1(k)$, $z^2(k)$ and $z^3(k)$, where $\bar{w}_1^{1,(1:L)}$ and $\bar{w}_2^{2,(1:L)}$ are evenly dominating for the particles $\hat{\mathbf{x}}_1^{(1:L)}(k)$ and $\hat{\mathbf{x}}_2^{(1:L)}(k)$, respectively. Finally, the estimated object state $\bar{\mathbf{x}}(k)$ is obtained with the particles information averaged over model 1 and 2, which is close not to $\hat{\mathbf{x}}_1^{(1:L)}(k)$ but to $\hat{\mathbf{x}}_2^{(1:L)}(k)$. In this case, the estimated position $(\bar{x}(k), \bar{y}(k))$ is satisfied with (3.16), but the bearing of the estimated position $\arctan\left(\frac{\bar{y}(k)}{\bar{x}(k)}\right)$ is not satisfied with (3.15) as illustrated in Figrue 3-13(c). Thus, the 95% confidence particles ellipses in (3.16) and the 95 % confidence true bearing range in (3.15) should be considered together.

3.3.4 Performance Evaluation

In this subchapter, the performance of the triggering-based visual sensor cooperation is evaluated with the comparison to the performance of the periodic visual sensor cooperation as well as no visual sensor cooperation. For the performance evaluation, the environment described in Figrue 3-7(a) is considered with 200 acoustic sampling times with 100 trials. Figrue 3-14 shows the average RMS position errors corresponding to triggering based visual sensor cooperation, periodic visual sensor cooperation and no visual sensor cooperation. As shown in Figrue 3-14(a), the average RMS position errors of object O^1 is 1.38 based on the triggering based visual sensor cooperation and 7.54 without the visual sensor cooperation. Also, the average RMS position errors with the periodic visual sensor cooperation are shown according to different visual sensor's sampling time T_v: $1T_s$ to $100T_s$. In the triggering based visual sensor cooperation, the average visual sensor's sampling time T_v is approximately $4.55T_s$. In the periodic visual sensor cooperation, on the other hand, the visual sensor's sampling time T_v corresponding to the average RMS position error 1.38 is approximately $4.16T_s$. It shows that the triggering based visual sensor cooperation requires less visual sensor resources than the periodic visual sensor cooperation for the same tracking performance. Similarly, Figrue 3-14(b) and 3-14(c) show the same pattern for the objects O^2 and O^3. Furthermore, in the periodic visual sensor cooperation, the visual sensor's sampling time T_v corresponding to the average RMS position error 0.64 is approximately $4.21T_s$.

Table 3.1(a) summarizes the average RMS position errors with the triggering



(a) Average RMS position errors of object O^1

(b) Average RMS position errors of object O^2

(c) Average RMS position errors of object O^3

Figure 3-14: Average RMS position errors with three cooperation approaches. For the periodic visual sensor cooperation, the period varies from $1T_s$ to $100T_s$

based visual sensor cooperation and the no visual sensor cooperation. Table 3.1(b) summarizes the average triggered visual sensor's sampling time and the periodic visual sampling time corresponding to the performance level as same as the RMS position error in the triggering sensor based cooperation.

In practice, the visual sensor cooperation period is unknown since the triggering mechanism is dependent on system dynamics and estimation performance. In any environment, the triggering visual sensors cooperation adapts the cooperation period while periodic visual sensors cooperation may waste resources. In addition, under the cooperation period restriction due to network delay and image processing, the triggering mechanism may support the cooperation to the objects with the highest priority since it recognizes critical ones.

	no visual sensor cooperation	triggering based visual sensors cooperation
Object 1	7.54	1.38
Object 2	3.70	0.24
Object 3	5.54	0.32
Total average	5.59	0.64

(a) Average RMS position errors

	periodic visual sensors cooperation	triggering based visual sensors cooperation
Object 1	$4.16T_s$	$4.55T_s$
Object 2	$4.28T_s$	
Object 3	$4.19T_s$	
Total average	$4.21T_s$	

(b) Equivalent visual sensor cooperation period

Table 3.1: Performances of the triggering based visual sensor cooperation, the periodic visual sensors cooperation and the no visual sensor cooperation

3.4 Simulation and Analysis

3.4.1 Simulation Setup

The visual sensor cooperation with the acoustic sensor based estimation is simulated in an indoor environment with size $14.63m \times 8.23m$ illustrated in Figrue 3-15. Object O^1 starts with initial velocity (0m/s, -0.3m/s) from position $(2.9m, 4.5m)$, object O^2 starts with initial velocity $(0.3m/s, -0.1m/s)$ from position $(4.8m, 3.5m)$ and object O^3 starts with initial velocity $(0m/s, 0m/s)$ from position (5.1m,8.2m). Two acoustic sensors A_1 and A_2 are deployed on the ceiling positioned at $(3.2m, 1.9m)$ and $(7.6m, 6.8m)$ each with 100 emulated samples per second. Each acoustic sensor receives the acoustic samples with variance 3 during 19 seconds, and tracks the objects independently. Three visual sensors V_1, V_2 and V_3 are placed at positions $(1.9m, 6.3m)$, $(13.4m, 5.0m)$ and $(5.5m, 0.3m)$ each with 6 samples per second.

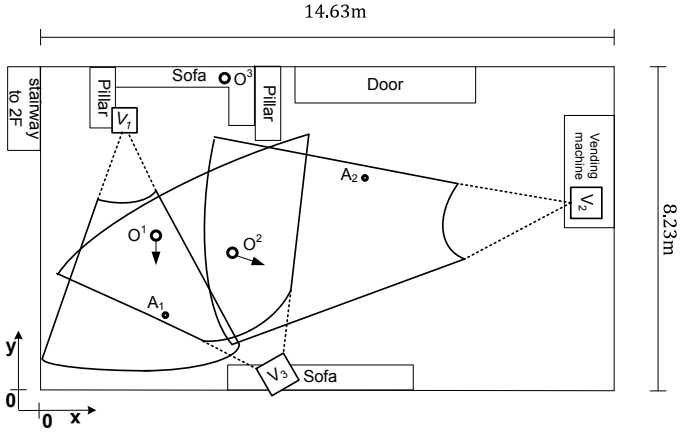


Figure 3-15: The visual sensor cooperation with an acoustic sensor based estimation is simulated in an indoor environment with size $14.63m \times 8.23m$.

3.4.2 Objects Dynamic Characteristics with Acoustic Sensing Range

Figrue 3-16 shows the three objects movement by switching three dynamic models: the constant velocity with $\mathbf{F}^{(1)}$ (CV), the clockwise coordinated turn with $\mathbf{F}^{(2)}$ (CT) and the anticlockwise coordinated turn with $\mathbf{F}^{(3)}$ (ACT) in (3.12).

Object O^1 starts with the CV model for 3.6 seconds. Between 3.6s and 7.1s, the object moves with the ACT model with $\alpha = 0.15m/s^2$. Between 7.1s and 13.5s, the object moves with the CV model. Between 13.5s and 15.0s, the object moves with ACT model with $\alpha = 0.20m/s^2$. Finally, between 15.0s and 19.0s, the object moves with the CV model. Object O^2 starts with the CV model for 4.5s. Between 4.5s and 6.1s, the object moves with the ACT model with $\alpha = 0.45m/s^2$. Between 6.1s and 7.5s, the object moves with the CV model. Between 7.5s and 8.5s, the object moves with the ACT model with $\alpha = 0.30m/s^2$. Between 8.5s and 9.5s, the object moves with the CT model with $\alpha = 0.30m/s^2$. Between 9.5s and 13.0s, the object moves

with CV the model. Between 13.0s and 16.0s, the object moves with the CT model with $\alpha = 0.25m/s^2$. Finally, between 16.0s and 19.0s, the object moves with the CV model. Object O^3 initially does not move without transmitting sound wave for 13.0 seconds, and starts to move with the CV model between 13.0s and 19.0s.

In addition, given the acoustic sensors A_1 and A_2 shown in Figrue 3-16, if the measurement is received by only acoustic sensor A_1, a circle is marked ('o'). If the measurement is received by only acoustic sensor A_2, a square is marked ('□'). If the measurement is received by both sensors A_1 and A_2, a diamond is marked ('◇'). If the measurement is not received by any of two sensors, a star is marked ('*').

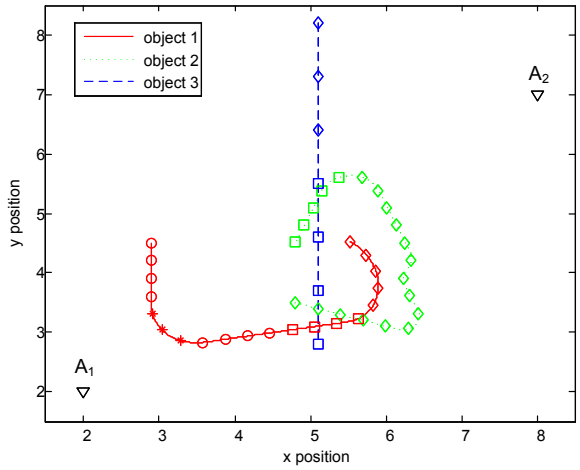
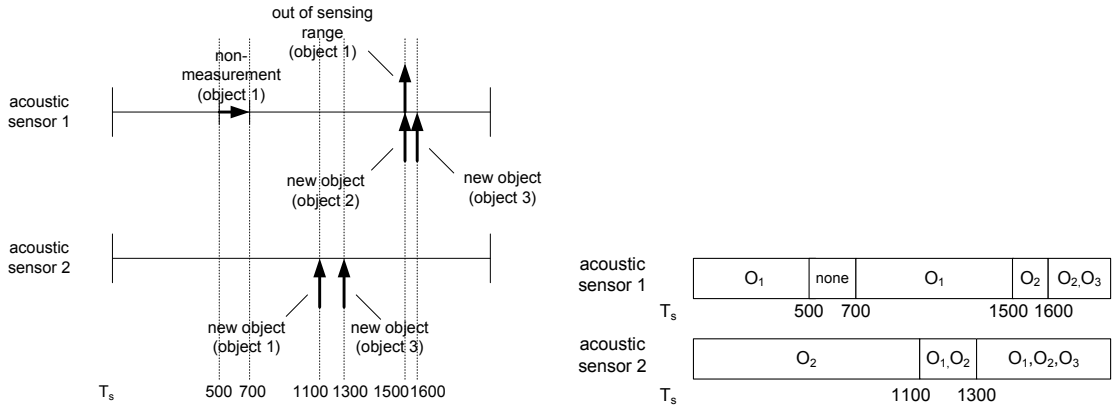


Figure 3-16: Three objects trajectories and the positions of the two acoustic sensors

Figrue 3-17(a) arranges non-measurement, new object appearance and movement out of sensing range with respect to each sensor. Sensor A_1 initially receives one measurement from object O^1 and does not receive measurements between 5.0s and 7.0s. At time 15.0s, sensor A_1 starts to receive new measurement from object O^2, but starts to miss the measurement from object O^1 since object O^1 moves out the

sensing range. At time 16.0s, sensor A_1 starts to receive another new measurement from object O^3. Sensor A_2 initially receives one measurement from object O^2. At time 11s, sensor A_2 starts to receive new measurement from object O^1. At time 13.0s, sensor A_1 starts to receive another new measurement from object O^3. Figrue 3-17(b) shows that the measured objects from each sensor A_1 and A_2.



(a) Non-measurement, new object appearance and movement out of sensing range with respect to each sensor

(b) Measured objects from each sensor A_1 and A_2

Figure 3-17: Measured objects over time

3.4.3 Visual Sensor Cooperation with Triggering Timing Analysis

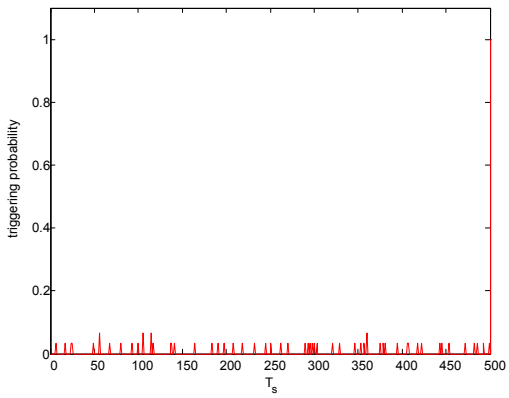
For the triggering timing analysis, the objects movement scenario is simulated 100 times, and the triggering timing is represented as the triggering probabilities. In addition, the triggering probabilities are compared with the two cases. The one is that the visual sensor supports the localized positions to acoustic sensor estimator when they are triggered. The other is that the visual sensor does not support the

localized positions to the acoustic sensor based estimation.

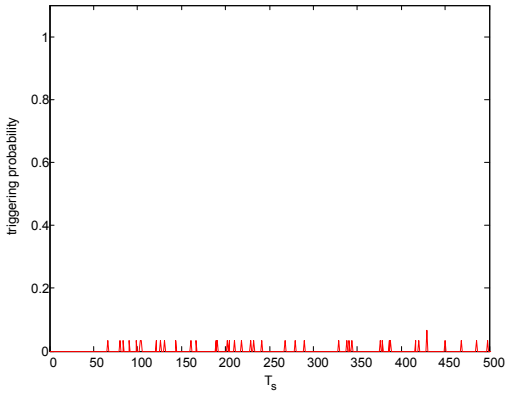
From time $1T_s$ to $500T_s$, acoustic sensor A_1 receives measurements from object O^1, and acoustic sensor A_2 receives measurements from object O^2. Since each sensor estimates different objects' state, it is considered as the single object estimation with a single sensor. Then, the triggering timing is obtained from the estimation performance only. Figrue 3-18(a) and 3-18(b) show the triggering probabilities with the visual sensor cooperation in sensors A_1 and A_2, respectively between $1T_s$ and $500T_s$. For comparison, Figrue 3-18(c) and 3-18(d) show the triggering probabilities without visual sensor cooperation in sensors A_1 and A_2, respectively.

From time $501T_s$ to $700T_s$, object O^1 does not transmit sound wave. Due to the non-measurement, the acoustic sensor A_1 triggers visual sensor cooperation: the number of objects and the number of measurements are different. Figrue 3-19 continually shows the triggering probabilities of the two sensors between $1T_s$ and $1{,}100T_s$ through 100 times trial. Figrue 3-19(a) and 3-19(b) show the triggering probabilities with visual sensor cooperation in sensors A_1 and A_2, respectively. Also, Figrue 3-19(c) and 3-19(d) show the triggering probabilities without the visual sensor cooperation in sensors A_1 and A_2, respectively.

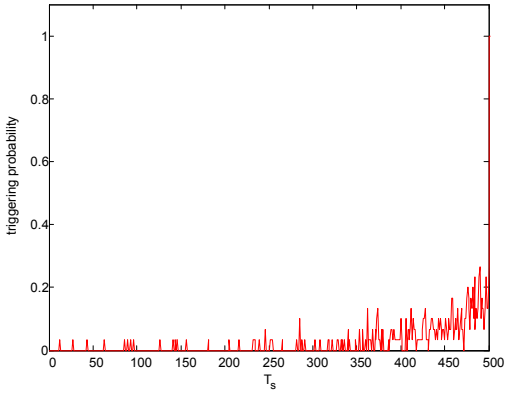
At time $1{,}101T_s$, acoustic sensor A_2 receives additional new measurement from object O^2. At time $1{,}300T_s$, acoustic sensor A_2 receives additional new measurement from object O^3. At time $1{,}500T_s$, acoustic sensor A_1 receives additional new measurement from object O^2, but the measurement from object O^1 is not received simultaneously. At time $1{,}600T_s$, acoustic sensor A_1 receives additional new measurement from object O^3. Figrue 3-20 shows the triggering probabilities of the two



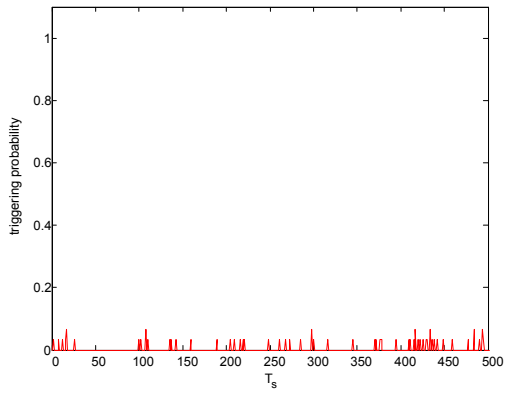
(a) Triggering probabilities with visual sensor cooperation in the sensor A_1



(b) Triggering probabilities with visual sensor cooperation in the sensor A_2



(c) Triggering probabilities without visual sensor cooperation in the sensor A_1

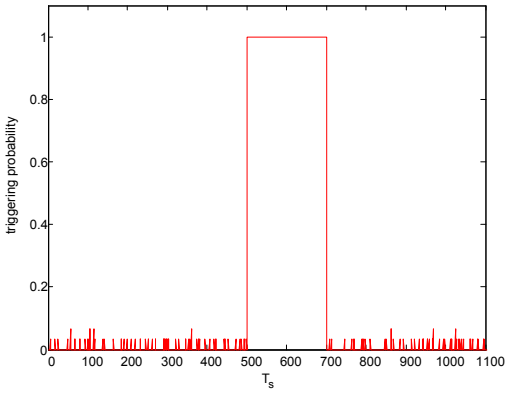


(d) Triggering probabilities without visual sensor cooperation in the sensor A_2

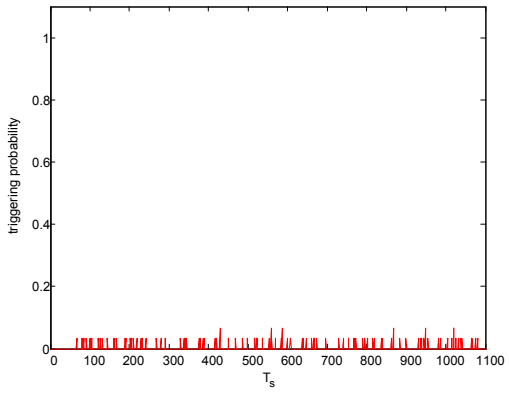
Figure 3-18: Triggering probabilities of the two sensors between $1T_s$ and $500T_s$

sensors between $1{,}100T_s$ and $1{,}900T_s$. Figrue 3-20(a) and 3-20(b) show triggering probabilities with the visual sensor cooperation in sensors A_1 and A_2, respectively. Also, for the comparison, Figrue 3-20(c) and 3-20(d) show the triggering probabilities without the visual sensor cooperation in sensors A_1 and A_2, respectively,

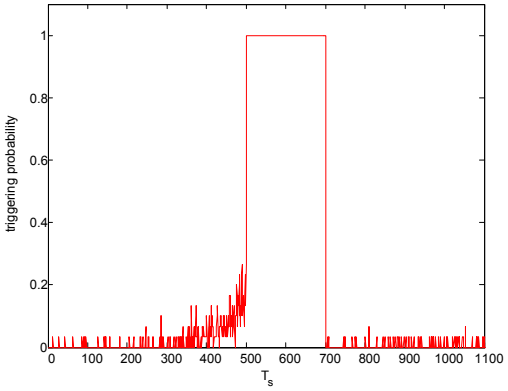
Finally, Figrue 3-21 shows the estimated final position of the three objects in each sensor. Figrue 3-21(a) shows the final estimated position with acoustic sensor A_1, and Figrue 3-21(b) shows the final estimated position with acoustic sensor A_2.



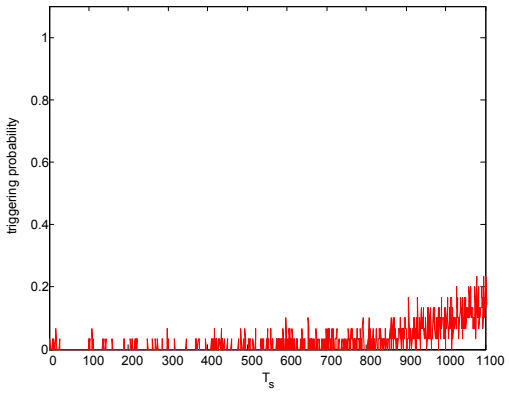
(a) Triggering probabilities with visual sensor cooperation in the sensor A_1



(b) Triggering probabilities with visual sensor cooperation in the sensor A_2



(c) Triggering probabilities without visual sensor cooperation in the sensor A_1

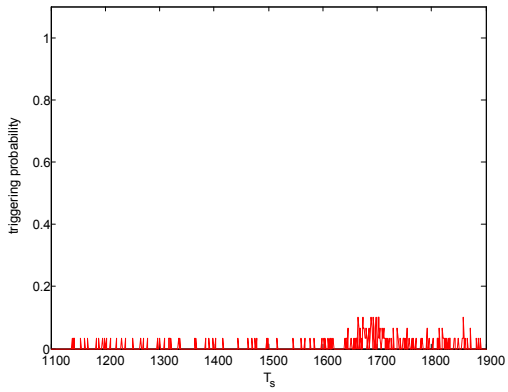


(d) Triggering probabilities without visual sensor cooperation in the sensor A_2

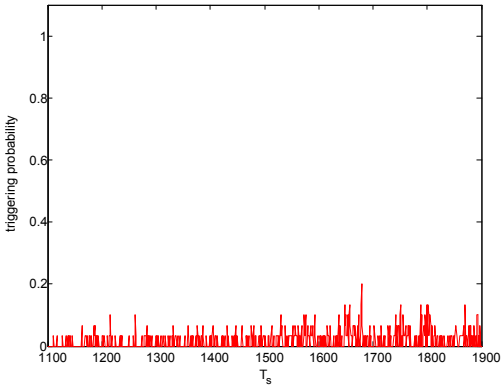
Figure 3-19: Triggering probabilities of two sensors time between $501T_s$ and $1100T_s$

3.5 Conclusion and Remarks

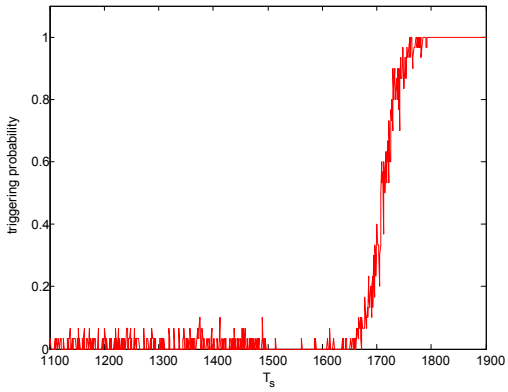
In this chapter, the acoustic-visual sensor cooperation method for multiple object tracking was presented. Since the visual sensor based object localization requires much higher computational complexity than acoustic sensor based estimation, the minimized visual sensor cooperation is adopted throughout this chapter. The visual sensor cooperation method was proposed based on the analysis of the limitation in



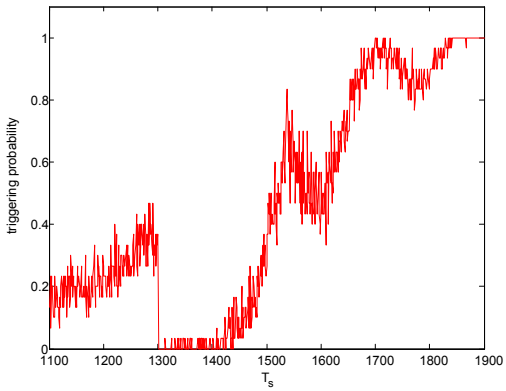
(a) Triggering probabilities with visual sensor cooperation in the sensor A_1



(b) Triggering probabilities with visual sensor cooperation in the sensor A_2



(c) Triggering probabilities without visual sensor cooperation in the sensor A_1

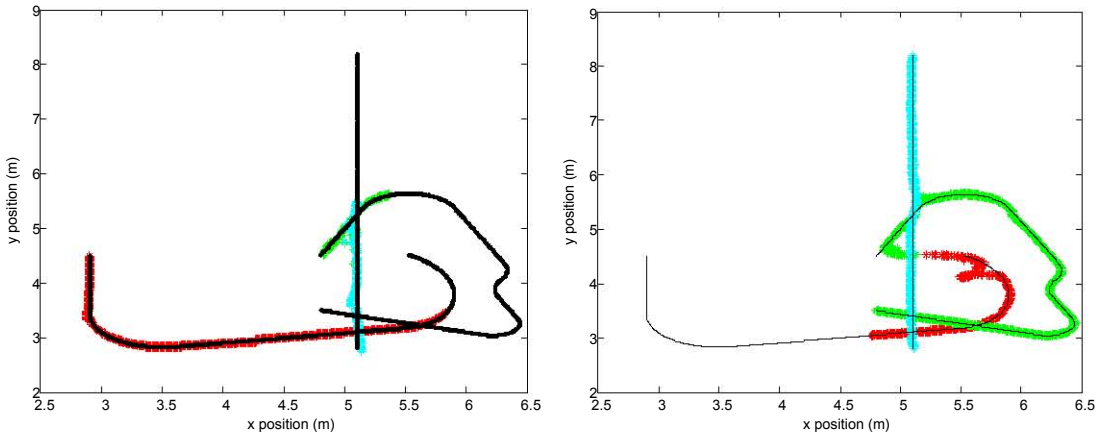


(d) Triggering probabilities without visual sensor cooperation in the sensor A_2

Figure 3-20: Triggering probabilities of two sensors time between $1101T_s$ and $1900T_s$

the acoustic sensor based estimation. In order to alleviate the limitation estimation, the visual sensor is triggered for the cooperation. For comparison, the proposed acoustic-visual sensor cooperation method was evaluated with a periodic visual sensor cooperation method and the no cooperation. Finally, the cooperation method was verified in a real environment.

In the future work, we extend the cooperation method in a large scale environment. Since an acoustic sensor has a limited coverage as well as a capacity measuring



(a) Final estimated position for acoustic sensor A_1

(b) Final estimated position for acoustic sensor A_2

Figure 3-21: Final estimated position for acoustic sensors A_1 and A_2

the limited of sound wave, it is required to deploy multiple acoustic sensors. We investigate the effects of interaction among acoustic sensors. In addition, we analyze the effect of visual sensors cooperation delay time since visual sensors and acoustic sensors receive measurements with different sampling rates.

Chapter 4

Local and Global Collaboration for Object Detection Enhancement with Information Redundancy

4.1 Introduction

Visual sensor based surveillance system has been great interest to diverse fields, and many researchers have made every effort to enhance the performance of object detection, tracking and localization [52] [53] [54]. Among the components, object detection by a visual sensor is not only a critical part to evaluate an overall surveillance system, but also a challenging problem. Difficulties in object detection arise due to abrupt object motion, varying lighting condition, changing appearance patterns of both an object and a background, non-rigid object structures, object-to-object occlusions and object-to-background occlusions. More specifically, [53] proposes a face detection

method for color images in the presence of varying lighting conditions and complex backgrounds. However, the method has an assumption that a human face of interest should be viewed by a visual sensor. [54] presents a comprehensive survey on object detection based on object motion and behaviors, and addresses an occlusion handling. It suggests that the most promising practical method for addressing occlusion is to utilize multiple visual sensors. The advantage of the multiple visual sensors is that when an object is detected by one or more visual sensors, any missed local object position is recovered based on a local and global collaboration. That is, detected local object positions are transformed into a global object position, and it aids in recovering any missed local object position.

Throughout this chapter, we identify the limitations of the collaboration, and propose to find the solution of the limitations. The collaboration may degrade the detection performance by propagating false object detection. For instance, when some of visual sensors detect a false object, the false local object position is propagated to the other non-detecting visual sensors with a falsely recovered local object position. Furthermore, the detected local object position has uncertainty even though it represents a true identical object. For instance, in an outdoor environment, a change detection algorithm may detect an object together with a shadow [55]. Then, the local object positions corresponding to an identical object are transformed into inequivalent global object positions. The inequivalent global object positions may be recognized as multiple objects. Our objective is to handle the inequivalent global object positions transformed by local object positions corresponding to an identical object. Furthermore, we minimize the performance degradation by preventing from

the propagation of the false detection.

4.2 Problem Description and Formulation

4.2.1 Application Model

Figure 4-1 shows an application model for an object detection enhancement, where multiple visual sensors share a viewable range. Once any of visual sensors detects an object, the detected local information is transformed into a global coordinates, and it is re-transformed into local information for all visual sensors; thus, any missing detection is recovered. Let denote the j-th visual sensor as V^j, where j =1, 2,..., J

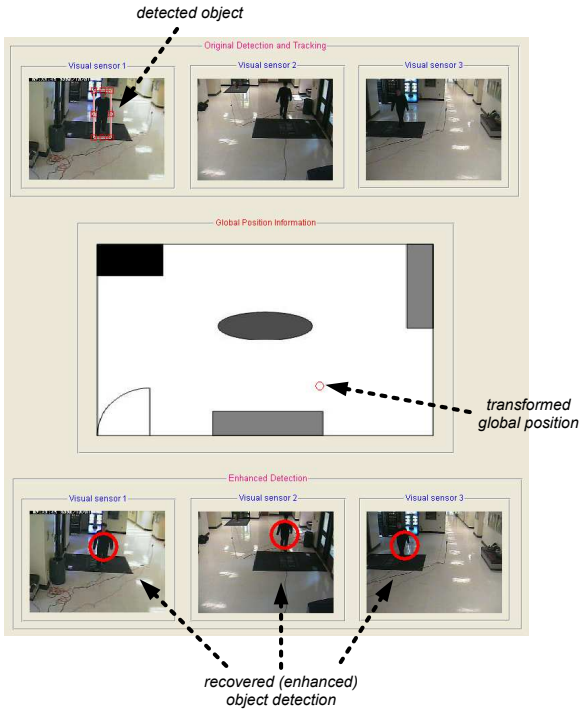


Figure 4-1: Application model for an object detection enhancement

for the number of visual sensors J. Once an object is detected by V^j, let denote the

detected local object position viewed by V^j as $\mathbf{l}^j(n)$ or $(l_x^j(n), l_y^j(n))$. The local object position $\mathbf{l}^j(n)$ is transformed into a global object position in a global coordinates denoted as $\mathbf{g}^j(n)$ or $(g_x^j(n), g_y^j(n))$. For the local to global transformation, we use a perspective model widely used in a surveillance system [57]. The global object position $\mathbf{g}^j(n)$ can be re-transformed into the local object position, and it is denoted as $\tilde{\mathbf{l}}^j(n)$ or $(\tilde{l}_x^j(n), \tilde{l}_y^j(n))$. By the local-global-local transformation using multiple visual sensors, any of missed local object position $\mathbf{l}^j(n)$ is recovered by $\tilde{\mathbf{l}}^j(n)$ as long as at least one of visual sensors detects an object. Throughout this chapter, we restrict our attention to single object detection.

4.2.2 Problem Formulation

Let us consider a binary hypothesis testing problem at time n with the following two hypothesis:

$$H_0(n) \quad : \quad \text{an object is absent in a surveillance area}$$

$$H_1(n) \quad : \quad \text{an object is present in a surveillance area}$$

The a-priori probabilities of the two hypotheses are denoted by $P(H_0(n)) = P_0(n)$ and $P(H_1(n)) = P_1(n)$. From the perspective of each visual sensor, each local object status such as detection or non-detection is classified as binary representation as

$$E^j(n) = E_1 \quad \text{when an object is detected by } V^j,$$

$$E^j(n) = E_0 \quad \text{when an object is not detected by } V^j.$$

Each visual sensor independently detects an object and that conditional probability is denoted by $P(E^j(n)|H_d(n))$, where d=1,2 and j=1,2...,J. Specifically, $P(E^j(n) = E_0|H_1(n))$ represents the probability that an object detection is missed, and $P(E^j(n) = E_1\ |H_0(n))$ represents the probability that a false object is detected. We denote the false and missing detection probabilities by $P_M^j(n)$ and $P_F^j(n)$, and the probabilities are equivalent to the following conditional probabilities as

$$P_M^j(n) = P(E^j(n) = E_0|H_1(n)), \tag{4.1}$$

$$P_F^j(n) = P(E^j(n) = E_1|H_0(n)). \tag{4.2}$$

We assume $P_M^j(n)$ and $P_F^j(n)$ are known.

After processing the object detections locally, the local object status $E^j(n)$ such as E_0 or E_1 from V^j is transmitted to a global information center. Based on above specification, [56] formulates an overall decision function based on an optimal decision rule as

$$\begin{aligned} Y(E^1(n), ..., E^J(n)) &= \log \frac{P_1(n)}{P_0(n)} + \sum_{S_{E_1}} \log \frac{1 - P_M^j(n)}{P_F^j(n)} \\ &\quad + \sum_{S_{E_0}} \log \frac{P_M^j(n)}{1 - P_F^j(n)} \\ &= \begin{cases} > 0 \Rightarrow G(n) = G_1 \\ < 0 \Rightarrow G(n) = G_0, \end{cases} \end{aligned} \tag{4.3}$$

where

$G(n) = G_0$: the detected object is declared false at time n

$G(n) = G_1$: the detected object is declared true at time n.

4.3 Object Detection Enhancement

4.3.1 Quality Information based Object Decision

The overall decision function $Y(\cdot)$ is easily biased by the number of visual sensors. That is, given P_M^j=P_F^j,

$$Y(\cdot) = \log \frac{P_1(n)}{P_0(n)} + (N(S_{E_1}) - N(S_{E_0})) \log \frac{1 - P_M^j}{P_F^j}, \tag{4.4}$$

where $N(S_{E_1})$ is the element number of a set S_{E_1}, and $N(S_{E_0})$ is the element number of a set S_{E_0}.

Hence, we consider quality information corresponding to a detected object by visual sensor V^j. The quality information indicates the degree of confidence as to $E^j = E_1$, and denoted by W^j, where $j \in S_{E_1}$ and $0 \leq W^j \leq 1$. Note that W^j is not existent when $E^j = E - 0$. When W^j is close to zero, E^j is with *less confidence*. On the other hand, when W^j is close to one, E^j is with *more confidence*. By considering quality information W^j corresponding to E^j, the maximum a-posterior probability

based decision rule is

$$P(H_1|E^{1:J}, W^{1:J}) \underset{H_0}{\overset{H_1}{\gtrless}} P(H_0|E^{1:J}, W^{1:J}), \tag{4.5}$$

where $E^{1:J} = \{E^1, ..., E^J\}$ and $W^{1:J} = \{W^1, ..., W^J\}$. Note we omit the time notation n for simplicity.

Since W^j indicates the degree of confidence as to $E^j = E_1$, E^j is weighted by each corresponding quality information W^j as

$$P(H_1|E^1 \cdot W^1, ...) \underset{H_0}{\overset{H_1}{\gtrless}} P(H_0|E^1 \cdot W^1, ...). \tag{4.6}$$

From Bayes theorem, we have

$$\frac{P(E^1 \cdot W^1, ..., E^J \cdot W^J|H_1)}{P(E^1 \cdot W^1, ..., E^J \cdot W^J|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \frac{P_0}{P_1}, \tag{4.7}$$

and the corresponding likelihood ratio test (LRT) is described as

$$\frac{P(E^1 \cdot W^1, ..., E^J \cdot W^J|H_1)}{P(E^1 \cdot W^1, ..., E^J \cdot W^J|H_0)} = \begin{cases} > \frac{P_0}{P_1} \Rightarrow E = E_1 \\ < \frac{P_0}{P_1} \Rightarrow E = E_0 \end{cases}. \tag{4.8}$$

For simplicity, we denote $E^j \cdot W^j$ by E^j_w, and the left-hand side of (4.8) is simplified and decomposed as

$$\frac{P(E^{1:J}_w|H_1)}{P(E^{1:J}_w|H_0)} = \underbrace{\prod_{S_{E_0}} \frac{P(E^j_w|H_1)}{P(E^j_w|H_0)}}_{\mathbf{A}} \cdot \underbrace{\prod_{S_{E_1}} \frac{P(E^j_w|H_1)}{P(E^j_w|H_0)}}_{\mathbf{B}} \tag{4.9}$$

where $\{E^1_w, ..., E^J_w\} = E^{1:J}_w$, **A** is for only E^j consideration on the condition of $E^j = E_0$, and **B** is for both E^j and W^j consideration on the condition of $E^j = E_1$.

A of (4.9) is

$$\prod_{S_{E_0}} \frac{P(E^j_w|H_1)}{P(E^j_w|H_0)} = \prod_{S_{E_0}} \frac{P(E^j = E_0|H_1)}{P(E^j = E_0|H_0)} = \prod_{S_{E_0}} \frac{P^j_M}{1 - P^j_F}, \tag{4.10}$$

and **B** of (4.9) is

$$\begin{aligned} \prod_{S_{E_1}} \frac{P(E^j_w|H_1)}{P(E^j_w|H_0)} &= \prod_{S_{E_1}} \frac{P(E^j = E_1|H_1) \cdot W^j}{P(E^j = E_1|H_0) \cdot (1 - W^j)} \\ &= \prod_{S_{E_1}} \frac{(1 - P^j_F) \cdot W^j}{P^j_F \cdot (1 - W^j)}. \end{aligned} \tag{4.11}$$

By substituting (4.10) and (4.11) into (4.9),

$$\frac{P(E^{1:J}_w|H_1)}{P(E^{1:J}_w|H_0)} = \prod_{S_{E_0}} \frac{P^j_M}{1 - P^j_F} \cdot \prod_{S_{E_1}} \frac{(1 - P^j_F) \cdot W^j}{P^j_F \cdot (1 - W^j)}, \tag{4.12}$$

and the corresponding log-LRT is

$$\begin{aligned}\log\frac{P(E_w^{1:J}|H_1)}{P(E_w^{1:J}|H_0)} &= \log\frac{P_1}{P_0}-\sum_{S_{E_0}}\log\frac{1-P_F^j}{P_M^j}\\ &\quad+\sum_{S_{E_1}}\log\frac{1-P_F^j}{P_F^j}+\sum_{S_{E_1}}\log\frac{W^j}{1-W^j}\\ &= \begin{cases} >0 \Rightarrow G(n)=G_1\\ <0 \Rightarrow G(n)=G_0 \end{cases}.\end{aligned} \tag{4.13}$$

By considering the quality information W^j corresponding to the local decision E^j, we reduce the decision bias as to the difference between $N(S_{E_1})$ and $N(S_{E_0})$.

4.3.2 Dynamic Model based a Priori Probabilities

As in (4.13), if $G(n)=G_1$, the detected object is considered true, and $\tilde{\mathbf{l}}^j(n)$ replaces the original $\mathbf{l}^j(n)$. On the other hand, if $G(n)=G_0$, the detected object is considered as false, and the original $\mathbf{l}^j(n)$ is eliminated. However, the overall decision function requires exact knowledge of the a-priori probabilities of the hypotheses, $P_1(n)$ and $P_0(n)$.

In order to obtain the $P_1(n)$ and $P_0(n)$, we first define a global object state $\hat{\mathbf{g}}(n)$ as

$$\hat{\mathbf{g}}(n)=[\hat{g}_x(n)\ \ \hat{g}_{vx}(n)\ \ \hat{g}_y(n)\ \ \hat{g}_{vy}(n)]^T \tag{4.14}$$

where $[\hat{g}_x(n)\ \ \hat{g}_y(n)]$ and $[\hat{g}_{vx}(n)\ \ \hat{g}_{vy}(n)]$ are true global object position and velocity at time n. Consider the global object state $\hat{\mathbf{g}}(n)$ with discrete time instant $n\in$

$\{1, 2, \cdots\}$, evolving according to

$$\hat{\mathbf{g}}(n) = \mathbf{F}(n-1) \cdot \hat{\mathbf{g}}(n-1) + \mathbf{Q}, \tag{4.15}$$

where $\mathbf{Q}$ includes a Gaussian noise for an object position described as

$$\mathbf{Q} = \begin{bmatrix} N(0, \sigma^2) & 0 & N(0, \sigma^2) & 0 \end{bmatrix}^T, \tag{4.16}$$

and $\mathbf{F}(n-1)$ is a dynamic transition function [49].

Given the previous global object position $(g_x(n-1), g_y(n-1))$ at time $n-1$, the possible global object position range at time n is estimated with (5.35). Let denote the estimated mean position as $(\bar{g}_x(n), \bar{g}_y(n))$, and it is obtained from $\bar{\mathbf{g}}(n)$ as

$$\bar{\mathbf{g}}(n) = \mathbf{F}(n-1) \cdot \hat{\mathbf{g}}(n-1). \tag{4.17}$$

From the perspective of Bayesian estimation, the posterior probability density function (PDF) as to $\mathbf{g}(n)$ is estimated by propagating the PDF over time [8]:

$$p(\mathbf{g}(n)|Z(1:n)) \propto p(Z(n)|\mathbf{g}(n)) \cdot p(\mathbf{g}(n)|Z(1:n-1)), \tag{4.18}$$

where $Z(n)$ represents a measurement at time n, and $Z(1:n)$ represents a history of measurements up to time n. Generally, the measurement term depends on a type of sensor and an application. In this chapter, the measurement $Z(n)$ is replaced by

$\bar{\mathbf{g}}(n)$, and it is obtained as

$$\bar{\mathbf{g}}(n) = \mathbf{F}(n-1) \cdot \mathbf{g}(n-1), \tag{4.19}$$

where $\mathbf{g}(n-1)$ is the final global object state at time $n-1$, and $\bar{\mathbf{g}}(n)$ represents $[\bar{g}_x(n) \;\; \bar{g}_{vx}(n) \;\; \bar{g}_y(n) \;\; \bar{g}_{vy}(n)]^T$. Given that an object follows the dynamic model $\mathbf{F}(n-1)$, the weight $w^j(n)$ corresponding to $\mathbf{g}^j(n)$ is evaluated how $\mathbf{g}^j(n)$ is close to $\bar{\mathbf{g}}(n)$ as

$$w^j(n) = \exp(-(\frac{(g_x^j(n) - \bar{g}_x(n))^2}{2\sigma^2} + \frac{(g_y^j(n) - \bar{g}_y(n))^2}{2\sigma^2})), \tag{4.20}$$

where $w^j(n)$ is obtained based on the 2-D Gaussian distribution function, and it denotes the probability that an object corresponding to $\mathbf{g}^j(n)$ and $\mathbf{l}^j(n)$ follows the dynamic model $\mathbf{F}(n-1)$. On the condition which an object moves based on a given dynamic model, $w^j(n)$ represents the quality information as to $\mathbf{g}^j(n)$ and $\mathbf{l}^j(n)$: $W^j(n) = w^j(n)$.

The associated set $\{\mathbf{g}^j(n), w^j(n)\}$ approximates the posterior pdf $p(\mathbf{g}(n)|\bar{\mathbf{g}}(1:n))$ as [8]

$$\begin{aligned} & p(\mathbf{g}(n)|\bar{\mathbf{g}}(1:n)) \\ \simeq \;\; & \frac{1}{\sum_{S_{E_1}} w^k(n)} \cdot \sum_{S_{E_1}} w^j(n) \cdot \delta\left(\bar{\mathbf{g}}(n) - \mathbf{g}^j(n)\right). \end{aligned} \tag{4.21}$$

For the final global object state $\mathbf{g}(n)$, each component of $\mathbf{g}^j(n)$ is weighted and averaged, which is based on a probability data association method (PDA) [58]. That is, $g_x^j(n)$ and $g_y^j(n)$ contribute the final global object position with each corresponding

$w^j(n)$ as

$$g_x(n) = \frac{\sum_{S_{E_1}} g_x^j(n) w^j(n)}{\sum_{S_{E_1}} w^j(n)}, \quad g_y(n) = \frac{\sum_{S_{E_1}} g_y^j(n) w^j(n)}{\sum_{S_{E_1}} w^j(n)}. \tag{4.22}$$

Once the final global object position $(g_x(n), g_y(n))$ is obtained, the position is evaluated how $(g_x(n), g_y(n))$ is close to $(\bar{g}_x(n), \bar{g}_y(n))$ as

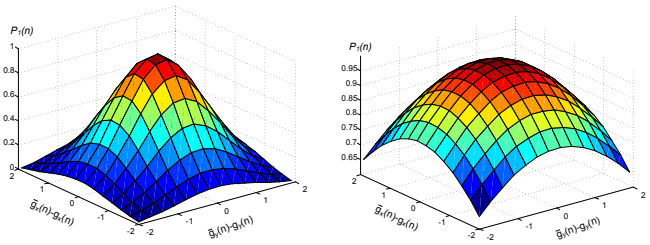
$$w(n) = \exp(-(\frac{(g_x(n) - \bar{g}_x(n))^2}{2\sigma^2} + \frac{(g_y(n) - \bar{g}_y(n))^2}{2\sigma^2})). \tag{4.23}$$

The evaluated $w(n)$ also denotes the probability that an object follows the dynamic model $\mathbf{F}(n-1)$. If $w(n)$ is close to zero, the object is completely deviated from the position based on dynamic model. On the other hand, if $w(n)$ is close to one, the object fully follows the dynamic model. On the condition which an object moves based on a dynamic model, $w(n)$ represents a-priori probability $P_1(n)$; thus the a-priori probabilities are

$$P_1(n) = w(n) \quad \text{and} \quad P_0(n) = 1 - w(n). \tag{4.24}$$

Figure 4-2 shows the a-priori probability $P_1(n)$ according to $g_x(n) - \bar{g}_x(n)$, $g_y(n) - \bar{g}_y(n)$ and Gaussian distribution noise σ.

In addition, in order to recursively obtain $\bar{\mathbf{g}}(n)$ in (4.19), the velocity is derived as $g_{vx}(n) = g_x(n) - g_x(n-1)$ and $g_{vy}(n) = g_y(n) - g_y(n-1)$, and $\mathbf{g}(n)$ is updated as $[g_x(n) \;\; g_{vx}(n) \;\; g_y(n) \;\; g_{vy}(n)]^T$.



(a) with $\sigma = 1$ (b) with $\sigma = 3$

Figure 4-2: The a-priori probability $P_1(n)$ corresponding to $w(n)$ are shown according to $g_x(n) - \bar{g}_x(n)$, $g_y(n) - \bar{g}_y(n)$ and σ.

4.4 Performance Analysis and Case Studies

4.4.1 Performance Analysis

We use six visual sensors sharing a viewable range with $P_M^j = P_F^j = 0.2$ for the performance analysis. Given $(\bar{g}_x(n), \bar{g}_y(n))$, each transformed global object position $(g_x^j(n), g_y^j(n))$, where $j \in S_{E_1}$, is collected to a decision center. We analyze the performance in declaring a detected object as a true according to a distance between $(g_x^j(n), g_y^j(n))$ and $(\bar{g}_x(n), \bar{g}_y(n))$. For the distance variation, we define σ_{xy} as

$$\begin{aligned} |g_x^j(n) - \bar{g}_x(n)| &\sim N(0, \sigma_{xy}^2) \\ |g_y^j(n) - \bar{g}_y(n)| &\sim N(0, \sigma_{xy}^2), \end{aligned} \tag{4.25}$$

and the corresponding $W^j(n)$ is investigated with Gaussian variance $\sigma^2 = 1$.

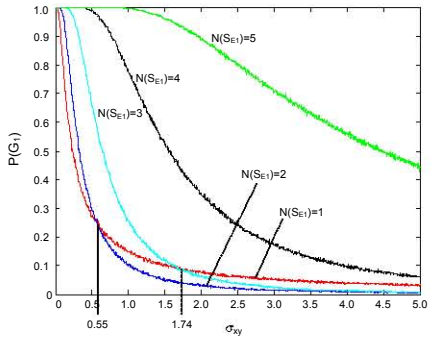
Figure 4-3(a) shows a probability of a true object declaration, $P(G_1)$, according to $N(S_{E_1})$ and σ_{xy}^2. The probability is calculated through 10,000 simulation trials: $P(G_1)$=$N(G_1)/10,000$. Figure 4-3(b) shows the values σ_{xy}^2 which are satisfied with

$P(G_1) > 0.99$ and $P(G_1) > 0.50$ according to $N(S_{E_1})$. When $N(S_{E_1}) = 5$, $P(G_1)$ is more than 0.99 until σ_{xy}^2 reaches 1.135, and $P(G_1)$ is more than 0.5 when σ_{xy}^2 reaches 4.485. When $N(S_{E_1}) = 4$, $P(G_1)$ is more than 0.99 until σ_{xy}^2 reaches 0.480, and $P(G_1)$ is more than 0.5 when σ_{xy}^2 reaches 1.565. When $N(S_{E_1}) = 3$, $P(G_1)$ is more than 0.99 until σ_{xy}^2 reaches 0.195, and $P(G_1)$ is more than 0.50 when σ_{xy}^2 reaches 0.635. When $N(S_{E_1}) = 2$, $P(G_1)$ is more than 0.99 until σ_{xy}^2 reaches 0.085, and $P(G_1)$ is more than 0.50 when σ_{xy}^2 reaches 0.325. When $N(S_{E_1}) = 1$, $P(G_1)$ is more than 0.99 until σ_{xy}^2 reaches 0.030, and $P(G_1)$ is more than 0.55 when σ_{xy}^2 reaches 0.235. Note when σ_{xy}^2 exceeds 0.55, $P(G_1)$ with $N(S_{E_1}) = 1$ becomes larger than $P(G_1)$ with $N(S_{E_1}) = 2$ as illustrated in Figure 4-3(a). Also, note when σ_{xy}^2 exceeds 1.74, $P(G_1)$ with $N(S_{E_1}) = 1$ becomes larger than $P(G_1)$ with $N(S_{E_1}) = 3$. It shows that the quality information based true/false object decision is dependent on the accuracy of detected local information and not entirely dependent on the number of visual sensors detecting an object.

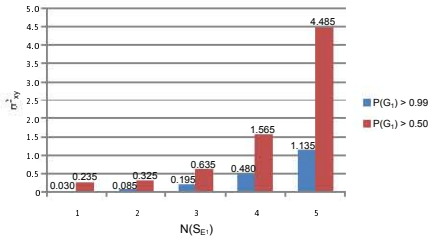
4.4.2 Case Studies

Figure 4-4 through 4-6 show the object detection enhancement with three visual sensors in different scenarios. For the object detection enhancement, we assume that $P_M^j = P_F^j = 0.2$ and $W^j(n)$ is investigated with Gaussian variance $\sigma^2 = 1$.

In Figure 4-4(a), the visual sensor 1 does not detect an object due to an occlusion while the other two visual sensors correctly detect the object. Each two detected local object position is transformed into the global object position as $\mathbf{g}^2(n) = (5.2, 5.2)$ and



(a) $P(G_1)$ according to $N(S_{E_1})$ and σ_{xy}^2

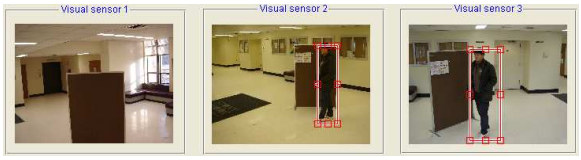


(b) σ_{xy}^2 for $P(G_1) > 0.99$ and $P(G_1) > 0.50$

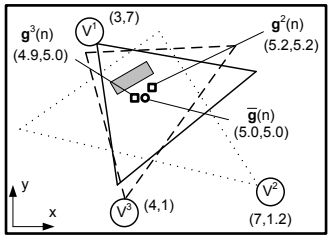
Figure 4-3: Relationship among $N(S_{E_1})$, σ_{xy}^2 and $P(G_1)$

$\mathbf{g}^3(n) = (4.9, 5.0)$. Figure 4-4(b) illustrates the surveillance environment in a global coordinates including visual sensor positions, viewable ranges as well as the positions $\mathbf{g}^2(n)$, $\mathbf{g}^3(n)$ and $\bar{\mathbf{g}}(n)$. Given the position $\bar{\mathbf{g}}(n)$ as (5.0,5.0), $W^2(n)$ and $W^3(n)$ are 0.9608 and 0.9950, respectively. Thus, we obtain the final global object position $\mathbf{g}(n)$ as (5.05,5.10) and $P_1(n)$ as 0.9938. In addition, it results in the value of an overall decision as 2.78. Based on the true object declaration, the object detection is enhanced as shown in Figure 4-4(c), where the missed local object position from the visual sensor 1 is recovered.

In Figure 4-5(a), the visual sensor 1 correctly detects an object while the other



(a) Original detection



(b) Global map

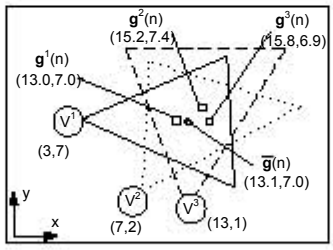
(c) Enhanced detection

Figure 4-4: Original and enhanced detection (case 1)

visual sensors 2 and 3 detect the object with each deviated local object position, which is merged with a shadow. Each three detected local object position is transformed into the global object position as $\mathbf{g}^1(n) = (13.0, 7.0)$, $\mathbf{g}^2(n) = (15.2, 7.4)$ and $\mathbf{g}^3(n) = (15.8, 6.9)$. Figure 4-5(b) also illustrates the surveillance environment in a global coordinates. Given the position $\bar{\mathbf{g}}(n)$ as (13.1,7.0), $W^1(n)$, $W^2(n)$ and $W^3(n)$ are 0.9950, 0.1018 and 0.026, respectively. Thus, we obtain the final global object position $\mathbf{g}(n)$ as (13.55,7.18) and $P_1(n)$ as 0.8892. In addition, it results in the value of an overall decision as 0.13. Based on the true object declaration, the object detection is enhanced as shown in Figure 4-5(c), where the deviated local object positions from

the visual sensors 2 and 3 are correctly recovered.

(a) Original detection



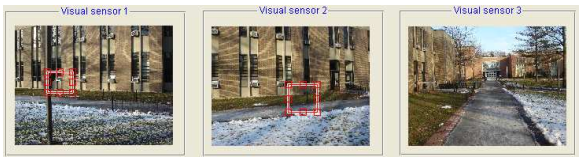
(b) Global map

(c) Enhanced detection

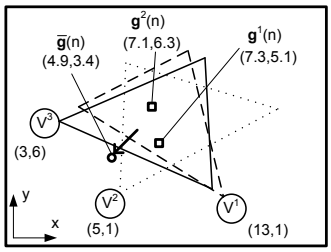
Figure 4-5: Original and enhanced detection (case 2)

In Figure 4-6(a), the visual sensors 1 and 2 detect a false object given an object is positioned out of viewable ranges. The detected local object position is transformed into the global object position as $\mathbf{g}^1(n) = (7.3, 5.1)$ and $\mathbf{g}^2(n) = (7.1, 6.3)$. Figure 4-6(b) also illustrates the surveillance environment in a global coordinates. Given the position $\bar{\mathbf{g}}(n)$ as (4.9,3.4), $W^1(n)$ and $W^2(n)$ are 0.0132 and 0.0013, respectively. Thus, we obtain the final global object position $\mathbf{g}(n)$ as (7.28,5.21) and $P_1(n)$ as 0.0114. In addition, it results in the value of an overall decision as -6.101. Based on the false object declaration, the global information assisted object detection enhancement

is shown in Figure 4-6(c), where all detected local object positions from the visual sensors 2 and 3 are eliminated. For the performance comparison, Figure 4-6(d) shows the result based on (4.3) without a-priori probabilities (i.e. $P_1(n) = P_0(n) = 0.5$). The value of an overall decision is 0.4; thus, the true object is declared, and the false object detection is propagated to other visual sensors.



(a) Original detection



(b) Global map: out of viewable range

(c) Enhanced detection

(d) based on (4.3) without a-priori probabilities

Figure 4-6: Original and enhanced detection (case 3)

4.5 Conclusions

In this chapter, we present an object detection enhancement with the collaboration of local and global information. In order to minimize the detection performance degradation, we use quality information indicating the degree of confidence as to each object detection. We show that the quality information based object true/false decision considers not only the number of visual sensors detecting an objects, but also the accuracy of detected local object position. In addition, it supports the a-priori probabilities for making a more precise decision on true/false object. Finally, the performance is analyzed and evaluated with occlusion, deviated detection and false detection cases.

Chapter 5

Conclusions and Future Work

This thesis is primarily dealing with three themes in object tracking based on acoustic and visual sensors with the frame of particle filter. The first one is the problem of 3 dimension formulation: "How do we efficiently formulate 3 dimensional object tracking under noisy characteristics?". The second one is to find the optimal sensor cooperation method, which considers resource usage minimization. The third one is to enhance object detection in visual tracking.

These themes have led us to explore the use of heterogeneous sensors network as well as apply particle filter for object tracking. This chapter summarizes the main contribution of the thesis and introduces ongoing future research.

5.1 Contributions

- We showed 3-D decomposition and plane selection. By exploiting the fact that the noisy measurements of the acoustic sensor differs on projected planes, we

proved the effective plane selection based on the characteristics. We illustrated that the particle filtering with the proposed plane selection is more flexible than the *direct 3-D method* where the proposed method can be easily extended to multiple sensor particle filtering. We have also analyzed the performance of the proposed methods using Cramer-Rao Lower Bound (CRLB) and compared to that of the *direct 3-D method*. We have shown that the proposed methods outperforms the *direct 3-D method*.

- The limitations of the particle filtering using a passive acoustic sensor for an object tracking are first addressed in mathematical and empirical studies. The performance of the tracking based on the passive acoustic sensor suffers from inability to detect the change of the dynamic model, unreliable measurements and unknown initial object state. From the perspectives, we propose and analyze an approach to enhance the performance of the tracking by incorporating visual association. The proposed approach is to minimize resource since a visual sensor require much higher resources than an acoustic sensor.

- we present an object detection enhancement with the collaboration of local and global information. In order to minimize the detection performance degradation, we use quality information indicating the degree of confidence as to each object detection. We show that the quality information based object true/false decision considers not only the number of visual sensors detecting an objects, but also the accuracy of detected local object position. In addition, it supports the a-priori probabilities for making a more precise decision on true/false object.

Finally, the performance is analyzed and evaluated with occlusion, deviated detection and false detection cases.

5.2 Future research

5.2.1 Temporal and Spatial Human Face Characterization

Application Model

In an environment with multiple humans as illustrated in Figure 5-1, a searching for specific person given a target face image is addressed.

Figure 5-1: A picture sample illustrating crowded environment

Figure 5-2 illustrates a system overview for searching specific person given a target face image. Let denote f_T as target face information, and f_{Ci} as candidate face information, where i=1,2,..., I for the number of detected faces I. Once a visual sensor detects human faces, the candidate face information f_{Ci} is compared with target face f_T. Similarity function $S(\cdot)$ has an argument $f_{Ci}|f_T$.

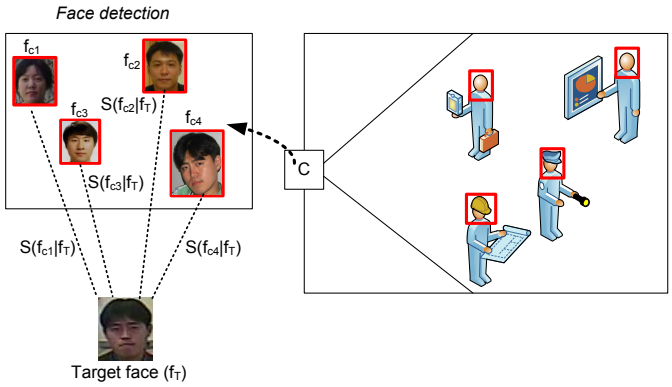


Figure 5-2: System overview: once a visual sensor detects human faces, each detected face f_{Ci} is compared with target face f_T. Similarity function $S(\cdot)$ has an argument $f_{Ci}|f_T$.

Face Information Characterization

In order to characterize face information, the target face f_T as a reference is represented by probability density function (pdf) of color. A candidate face f_{Ci} is also characterized by the pdf. The pdf is discrete densities with m-bin histogram, where the maximum value of m is 256. Thus, a target face pdf f_T and a candidate face f_{Ci} is formulated as

$$f_T = \{f_T^u\}_{u=1,...m}, \quad \sum_{u=1}^{m} f_T^u = 1 \tag{5.1}$$

$$f_{Ci} = \{f_{Ci}^u\}_{u=1,...m}, \quad \sum_{u=1}^{m} f_{Ci}^u = 1 \tag{5.2}$$

Given the target face pdf f_T and the candidate face pdf f_{Ci}, a similarity function $S(f_{Ci}|f_T)$ computes a likelihood representing similarity between the target face and the candidate face. Then, the following issues and questions arise: Do we properly match faces? If not, why? What is the limitation?

If we know additional information of moving direction, we estimate the part of face: front face, back head, etc. Suppose we have only a front face, the face recognition system should suspend decision making until a visual sensor views front face. If a visual sensor views part of the front face such as front right side, then we may make decision with the right side information.

Figure 5-3: Target image for search

(a) Face detection

(b) Candidate faces F_1 and F_2

Figure 5-4: Frame #1

Based on known camera positions, actual movement is roughly estimated with a monitored face movement. Figure 5-8 illustrates the movement and/or size of monitored faces according to camera positions and actual movement directions. For example, when a person is moving toward to N direction, left movement in Camera

(a) Face detection

(b) Candidate faces F_1, F_2 and F_3

Figure 5-5: Frame #26

1 and 2, smaller size in Camera 3, right movement in Camera 4, and bigger size in Camera 5 are monitored. Figure 5-9 represents the movement of monitored faces in more specific directions. In those movement directions, movement and size are both changed. When a person is moving toward *NE* direction, left movement and smaller size for Camera 1 and 2, right movement and smaller size for Camera 3, right movement and bigger size for Camera 4, and left movement and bigger size for Camera 5 are monitored. Hence, multiple faces from cameras are differentiated based on known camera positions.

To generalize the concept, the actual movement M_n^{pq} is expressed as

$$M_n^{pq} = r\cos\theta, \tag{5.3}$$

where M_n^{pq} is actual movement state of p^{th} face from q^{th} camera at time-instant n,

(a) Face detection

(b) Candidate faces F_2 and F_3

Figure 5-6: Frame #35

and r represents movement degree and θ is .

However, in the case that more than one person are moving toward the same direction, it is difficult to differentiate them. In addition, the uncertainty of the monitored face movement and size change needs to be considered as well.

Temporal and Spatial Face Characteristics

Feature space of an object is represented by probability density functions (PDF). The PDF is estimated by m- bin histograms, where m is the number of colors of an object. Given a reference PDF, a similarity degree is evaluated by comparing with candidate PDFs. Both the reference and the candidate PDFs are represented by m-bin histograms as an estimate to their PDFs as

$$\hat{f}^r = \{\hat{f}(u)^r\}_{u=1,\ldots,m} \ , \quad \hat{f}_i^c = \{\hat{f}(u)_i^c\}_{u=1,\ldots,m} \ , \tag{5.4}$$

(a) Face detection

(b) Candidate faces F_2 and F_3

Figure 5-7: Frame #46

where $\hat{f}^r$ and $\hat{f}_i^c$ represent the PDFs of a reference object and the i-th candidate object with m-bin histograms, respectively, where $i = \{1, 2, ..., I\}$ for the number of candidate objects I. $\hat{f}(u)^r$ and $\hat{f}(u)_i^c$ represent density of a reference object and i-th candidate object according to u-th bin, respectively.

For the similarity degree evaluation, Bhattacharyya coefficient is widely used []. The coefficient defines a normalized distance between $\hat{f}^r$ and $\hat{f}_i^c$. The sample estimate of Bhattacharyya coefficient between $\hat{f}^r$ and $\hat{f}_i^c$ is defined as

$$\rho_i \equiv \hat{\rho}_i[\hat{f}^r,\ \hat{f}_i^c] = \sum_{u=1}^{m} \sqrt{\hat{f}(u)^r \cdot \hat{f}(u)_i^c}. \tag{5.5}$$

The Bhattacharyya coefficient is in the range as $0 \leq \rho_i \leq 1$. If $\hat{f}^r$ and $\hat{f}_i^c$ are similar, ρ_i becomes close to one. Otherwise, ρ_i becomes close to zero. Note when we construct a histogram, we need to consider the width of the bins and the end points of the bins.

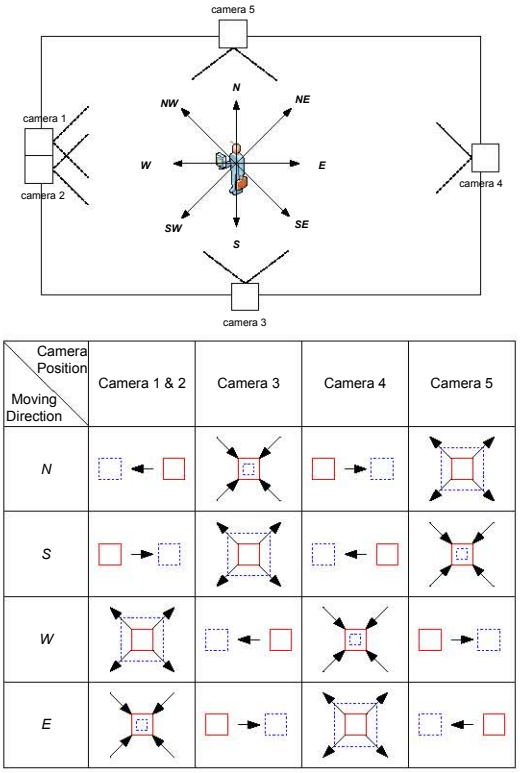


Figure 5-8: Movement or size change of monitored faces according to camera positions and moving directions (N, S, W and E)

As a result, the problems with histograms are that they are not smooth, depend on the width of the bins and the end points of the bins. In order to make a smooth histogram, which is independent on the end points of the bins, kernel estimators are applied in many applications. However, we are in fact interested in the accuracy of object identification without computational issue. In other words, we are exploring the question that how well a candidate object is identified given a reference object.

In order to investigate the performance evaluation of the Bhattacharyya coefficient, for simplicity, one reference object and three candidate objects are considered as illustrated in Figure 5-10. Figure 5-11 shows a PDF of the reference object, and Figure 5-12, 5-13 and 5-14 show PDFs of the candidate objects. According to the

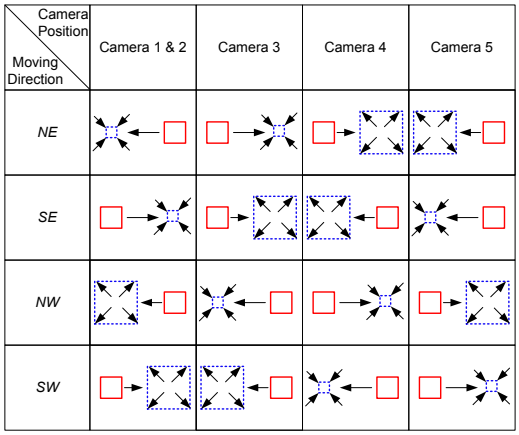


Figure 5-9: Movement and change of monitored faces according to camera positions and moving directions (NE, SE, NW and SW)

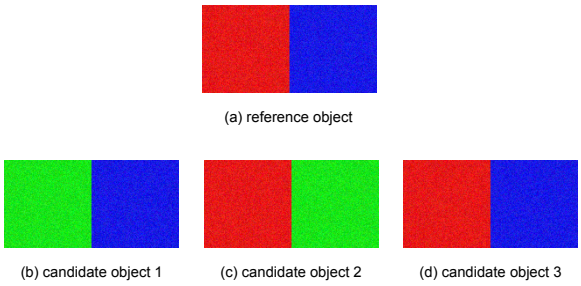


Figure 5-10: One reference object and three candidate objects

condition, the coefficient is obtained as $\rho_1 = 0.8039$, $\rho_2 = 0.8040$ and $\rho_3 = 0.9989$.

This methodology is applied in a face identification problem and the corresponding Bhattacharyya coefficients are shown as an example in Figure 5-20. The example shows a face identification among different races. However, within same race with same skin and hair color, the face identification becomes challenging. Figure 5-21 shows the limitation of the Bhattacharyya coefficients.

The proposed face identification technique is to use multiple spatial face information. The spatial face information is obtained by capturing a face image from different angles of a visual sensor. Figure 5-22 shows the image planes from different

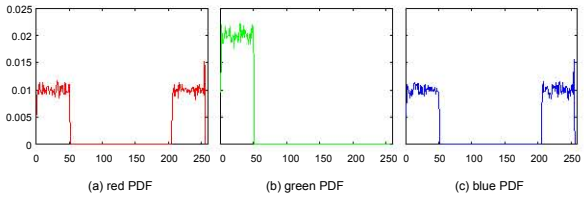


Figure 5-11: $\hat{f}^r$: reference object

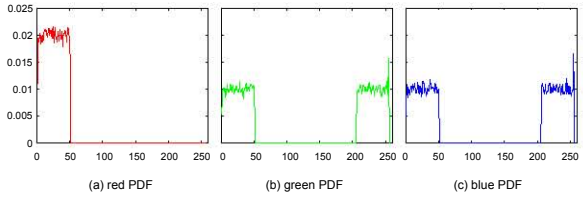


Figure 5-12: $\hat{f}_1^c$: candidate object 1

visual sensor angle by $\Delta\theta$. It stores a face information from front face to back head. Let denote the face (head) reference PDF viewed by relative angle $(n-1)\Delta\theta$ from a visual sensor as $\hat{f}^{r,[n]}$, where n=$\{1, 2, ..., N\}$ for the total number of spatial face information N. More specifically,

$$
\begin{aligned}
\hat{f}^{r,[1]} &: \text{a front face PDF} \\
&\vdots \\
\hat{f}^{r,[n]} &: \text{a left or right side of face (head) PDF with } (n-1)\Delta\theta \\
&\vdots \\
\hat{f}^{r,[N]} &: \text{a back head PDF}
\end{aligned}
$$

We assume that a face is symmetrical between left and right sides. Since $f^{r,[N]}$

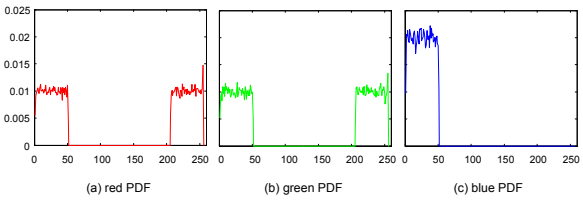


Figure 5-13: $\hat{f}_2^c$: candidate object 2

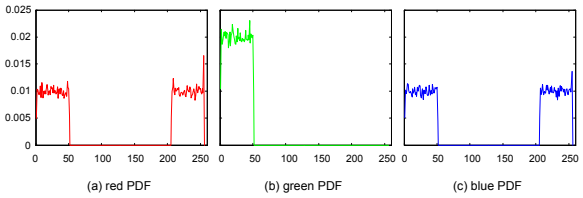


Figure 5-14: $\hat{f}_3^c$: candidate object 3

represents a back head,

$$\Delta\theta = \frac{\pi}{N-1} \quad \text{or} \quad N = \frac{\pi}{\theta} + 1 \tag{5.6}$$

Figure 5-23 shows an arrangement of multiple spatial face PDF.

5.2.2 Robot Navigation

The potential field method has been widely studied for autonomous mobile robot path planning, whose purpose is that a robot reaches a goal with an obstacle avoidance [63] [64]. The principle of the potential field method is that an obstacle exerts a repulsive force onto a robot while an goal applies an attractive force to a robot [65]. One of the reasons for the popularity of the method is its simplicity and mathematical elegance. However, it has some inherent limitations such as trap situations due to local minima. The trap situations may occur when a robot runs into a a dead end,

Figure 5-15: Target image for search

(a) Face detection

(b) Candidate faces F_1 and F_2

Figure 5-16: Frame #1

and a robot never reach a goal [66]. Many researchers have addressed their solutions with heuristic recovery or new potential functions [67] [68] [69].

Besides the limitations, there exists an additional problem, which is a collision by aligned robot-obstacle-goal (CAROG). In most of the previous studies, the positions of a robot, an obstacle and a goal are not aligned in a line. In this case, when a robot and a goal are blocked by an obstacle in a line, the repulsive force from an obstacle and the attractive force from a goal are exerted onto a robot in an opposite direction. If the repulsive force is bigger than the attractive force, a robot moves away from an obstacle and stops until the repulsive force and the attractive force are equal. That

(a) Face detection

(b) Candidate faces F_1, F_2 and F_3

Figure 5-17: Frame #26

is the one of trap situations due to local minima mentioned above. On the other hand, if the attractive force is bigger than the repulsive force, a robot moves toward an obstacle, and it collides with an obstacle when the inequality force condition is continuously kept hold.

In order to overcome this problem, we propose a new force function by taking into account a robot speed and sampling time. The new force function ensures that the repulsive force is always bigger than the attractive force before a robot collides with an obstacle; and thus, a robot never collide with an obstacle. However, another subsequent problem arises after the CAROG prevention. Once a robot moves away from an obstacle by the modified force function, it moves back and forth between two positions. Thus, we also propose the shortest path when a robot continuously oscillates two points.

(a) Face detection

(b) Candidate faces F_2 and F_3

Figure 5-18: Frame #35

Potential Field Method and CAROG Problem

For simplicity, we assume that a robot, an obstacle and a goal are represented by a point mass in two-dimension coordinates, respectively. Given a space with size $X_s \times Y_s$, each position is denoted by

$$\mathbf{p} = [x \quad y]^T \tag{5.7}$$

where $0 \leq x \leq X_s$ and $0 \leq y \leq Y_s$. In addition, each position of a robot, an obstacle and a goal is denoted by

$$\mathbf{p}_r = [x_r \quad y_r]^T, \quad \mathbf{p}_o = [x_o \quad y_o]^T \text{ and } \mathbf{p}_g = [x_g \quad y_g]^T. \tag{5.8}$$

In the potential field method, an attractive potential is defined as a function of the relative distance between a robot and a goal while a repulsive potential is defined as a

(a) Face detection

(b) Candidate faces F_2 and F_3

Figure 5-19: Frame #46

function of the relative distance between a robot and an obstacle. The two potential functions are commonly expressed as [65] [67] [70] [71]

$$U_{att}(\mathbf{p}) = c_{att} \cdot \big(\rho(\mathbf{p}, \mathbf{p}_g)\big)^m, \tag{5.9}$$

and

$$U_{rep}(\mathbf{p}) = \begin{cases} c_{rep} \cdot \left(\frac{1}{\rho(\mathbf{p},\mathbf{p}_o)} - \frac{1}{\rho_0}\right)^n & \text{if } \rho(\mathbf{p}, \mathbf{p}_o) \leq \rho_0 \\ 0 & \text{if } \rho(\mathbf{p}, \mathbf{p}_o) > \rho_0 \end{cases}, \tag{5.10}$$

where c_{att} and c_{rep} are constant values for an attractive potential and a repulsive potential. $\rho(\mathbf{p}, \mathbf{p}_g)$=$||\mathbf{p}, \mathbf{p}_g||$ is the shortest distance between two positions, $\mathbf{p}$ and $\mathbf{p}_g$. Similarly, $\rho(\mathbf{p}, \mathbf{p}_o)$=$||\mathbf{p}, \mathbf{p}_o||$ is the shortest distance between two positions, $\mathbf{p}$ and $\mathbf{p}_o$. ρ_0 is a positive constant denoting the distance influence of an obstacle. m and n are

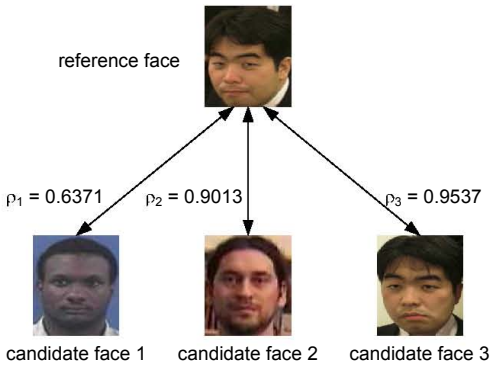


Figure 5-20: Among races: different skin and hair color

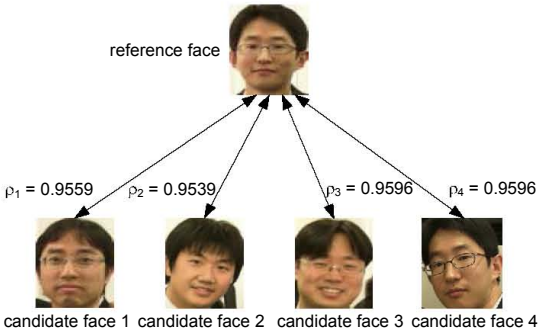


Figure 5-21: within a race: same skin and hair color

positive integer constants. For $m = n = 1$, each potential is conic in shape, and for $m = n = 2$, each potential is parabolic in shape.

The corresponding attractive force and repulsive force are then given by the negative gradient of each attractive potential and repulsive potential as

$$\begin{aligned} \mathbf{F}_{att}(\mathbf{p}) &= -\nabla U_{att}(\mathbf{p}) \\ &= -m \cdot c_{att} \cdot \big(\rho(\mathbf{p}, \mathbf{p}_g)\big)^{m-1} \cdot \nabla \rho(\mathbf{p}, \mathbf{p}_g), \end{aligned} \tag{5.11}$$

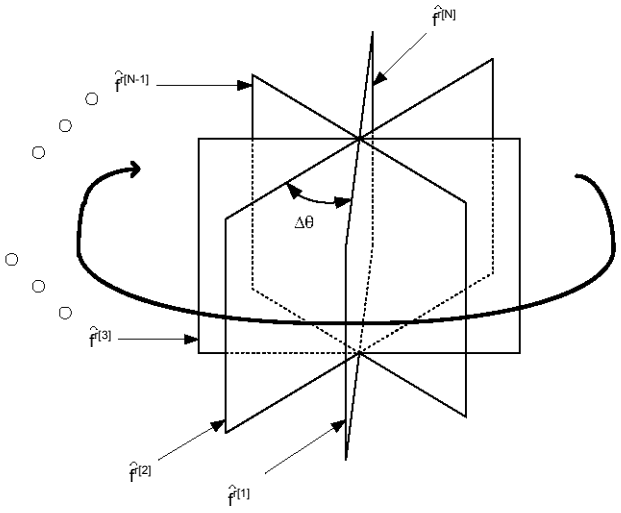


Figure 5-22: Image planes from different visual sensor angle by $\Delta\theta$

and

$$\begin{aligned}
\mathbf{F}_{rep}(\mathbf{p}) &= -\nabla U_{rep}(\mathbf{p}) \\
&= \begin{cases} n \cdot c_{rep} \cdot \left(\frac{1}{\rho(\mathbf{p},\mathbf{p}_o)} - \frac{1}{\rho_0}\right)^{n-1} \\ \cdot \left(\frac{1}{\rho(\mathbf{p},\mathbf{p}_o)}\right)^2 \cdot \nabla\rho(\mathbf{p},\mathbf{p}_o) \quad \text{if } \rho(\mathbf{p},\mathbf{p}_o) \le \rho_0 \\ 0 \qquad\qquad\qquad\qquad\qquad \text{if } \rho(\mathbf{p},\mathbf{p}_o) > \rho_0 \end{cases}
\end{aligned} \tag{5.12}$$

where $\nabla\rho(\mathbf{p},\mathbf{p}_o)$ and $\nabla\rho(\mathbf{p},\mathbf{p}_g)$ are two unit vectors pointing from an obstacle to a robot and from a goal to a robot, respectively. That is, the two unit vectors are expresses as

$$\nabla\rho(\mathbf{p},\mathbf{p}_o) = \frac{(x-x_o)\mathbf{u}_x + (y-y_o)\mathbf{u}_y}{\sqrt{(x-x_o)^2+(y-y_o)^2}}, \tag{5.13}$$

$$\nabla\rho(\mathbf{p},\mathbf{p}_g) = \frac{(x-x_g)\mathbf{u}_x + (y-y_g)\mathbf{u}_y}{\sqrt{(x-x_g)^2+(y-y_g)^2}} \tag{5.14}$$

where $\mathbf{u}_x$ and $\mathbf{u}_y$ are unit vectors in x and y direction, respectively.

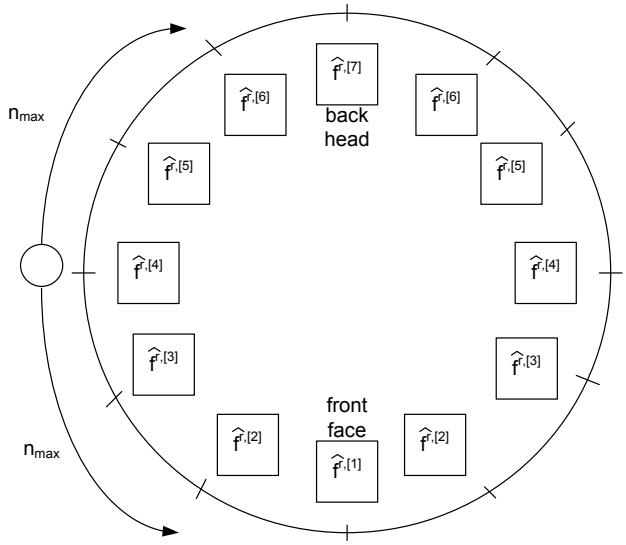


Figure 5-23: An arrangement of multiple spatial face PDF

The total force applied to each position $\mathbf{p}$ is the sum of the attractive force and the repulsive force as

$$\mathbf{F}_{tot}(\mathbf{p}) = \mathbf{F}_{att}(\mathbf{p}) + \mathbf{F}_{rep}(\mathbf{p}), \tag{5.15}$$

which determines the robot direction for reaching a goal with an obstacle avoidance. We assume that a robot moves with constant speed s_r and change its direction based on $\mathbf{F}_{tot}(\mathbf{p}_r)$ by every sampling time T_s. That is, a robot moves $s_r \cdot T_s$ with the direction of $\mathbf{F}_{tot}(\mathbf{p}_r)$ every T_s.

Figure 5-24 illustrates the total force $\mathbf{F}_{tot}(\mathbf{p}_r)$ onto a robot by addition of the attractive force $\mathbf{F}_{att}(\mathbf{p}_r)$ and the repulsive force $\mathbf{F}_{rep}(\mathbf{p}_r)$. When the robot approaches its goal, $\mathbf{F}_{att}(\mathbf{p}_r)$ becomes dominating and $\mathbf{F}_{rep}(\mathbf{p}_r)$ becomes negligible.

However, the above induced force based on the potential field method has an inherent problem, a collision by aligned robot-obstacle-goal (CAROG). Figure 5-25

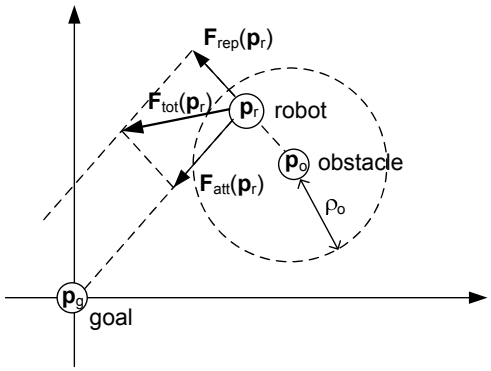


Figure 5-24: Total force $\mathbf{F}_{tot}(\mathbf{p}_r)$ onto a robot by addition of the attractive force $\mathbf{F}_{att}(\mathbf{p}_r)$ and the repulsive force $\mathbf{F}_{rep}(\mathbf{p}_r)$

illustrates the CAROG problem, where a robot and a goal are blocked by an obstacle in a line. A robot positioned at $\mathbf{p}_r$ is exerted by two forces, $\mathbf{F}_{att}(\mathbf{p}_r)$ and $\mathbf{F}_{rep}(\mathbf{p}_r)$, in an opposite direction: $\nabla\rho(\mathbf{p}, \mathbf{p}_o) = \nabla\rho(\mathbf{p}, \mathbf{p}_g)$. In the case, if $|\mathbf{F}_{att}(\mathbf{p}_r)| < |\mathbf{F}_{rep}(\mathbf{p}_r)|$, a robot moves away from a goal and an obstacle until $|\mathbf{F}_{att}(\mathbf{p}_r)| = |\mathbf{F}_{rep}(\mathbf{p}_r)|$. It is called trap situations due to local minima, which is another inherent limitation of the potential field method addressed by [67]. On the other hand, if $|\mathbf{F}_{att}(\mathbf{p}_r)| > |\mathbf{F}_{rep}(\mathbf{p}_r)|$ a robot moves close to an obstacle as well as a goal. Once the inequality condition is continuously kept hold until $|\mathbf{p}_r - \mathbf{p}_0| \leq s_r \cdot T_s$, a robot collides with an obstacle. In other words, CAROG problem is defined on the following three conditions

$$|\mathbf{F}_{att}(\mathbf{p}_r)| > |\mathbf{F}_{rep}(\mathbf{p}_r)| \tag{5.16}$$

$$|\mathbf{p}_r - \mathbf{p}_0| \leq s_r \cdot T_s, \tag{5.17}$$

$$\nabla\rho(\mathbf{p}, \mathbf{p}_o) = \nabla\rho(\mathbf{p}, \mathbf{p}_g). \tag{5.18}$$

For example, consider the case illustrated in Figure 5-25, where $\mathbf{p}_r$=$[12 \quad 12]^T$,

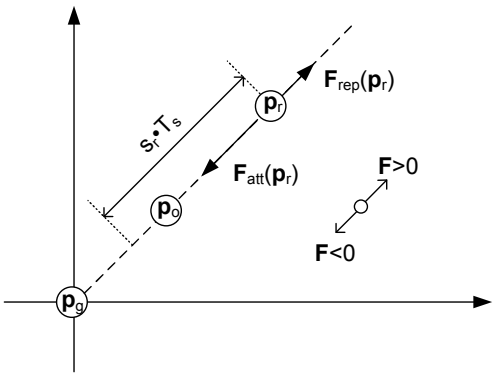


Figure 5-25: Robot and a goal are blocked by an obstacle in a line.

$\mathbf{p}_o$=[8 8]^T and $\mathbf{p}_g$=[6 6]^T. A robot moves with s_r=$\sqrt{2}$ and T_s=1; a robot moves $\sqrt{2}$ every sampling time. Both a robot and a goal are within the distance of influence of an obstacle with ρ_o=10. The attractive potential and repulsive potential are specifically given by [68]

$$U_{att}(\mathbf{p}) = \frac{1}{2}\left(\rho(\mathbf{p}, \mathbf{p}_g)\right)^2, \tag{5.19}$$

$$U_{rep}(\mathbf{p}) = 5\left(\frac{1}{\rho(\mathbf{p}, \mathbf{p}_o)} - \frac{1}{10}\right)^2. \tag{5.20}$$

Note that m=2, n=2, c_{att}=0.5 and c_{rep}=5 are used in a general form of (5.9) and (5.10). Their corresponding attractive force and repulsive force are obtained as

$$\mathbf{F}_{att}(\mathbf{p}) = -\rho(\mathbf{p}, \mathbf{p}_g) \cdot \nabla\rho(\mathbf{p}, \mathbf{p}_g), \tag{5.21}$$

$$\mathbf{F}_{rep}(\mathbf{p}) = 10\left(\frac{1}{\rho(\mathbf{p}, \mathbf{p}_o)} - \frac{1}{10}\right) \cdot \left(\frac{1}{\rho(\mathbf{p}, \mathbf{p}_o)}\right)^2 \cdot \nabla\rho(\mathbf{p}, \mathbf{p}_o). \tag{5.22}$$

Figure 5-26 shows the attractive force, the repulsive force and the total force in a

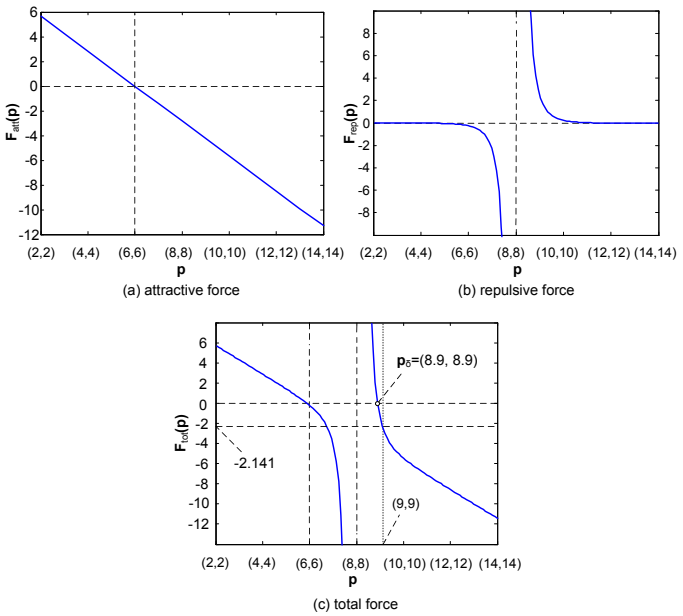


Figure 5-26: Example of CAROG problem

diagonal line from (2,2) to (14,14). In this example, for better understanding, each force direction away from an origin is denoted as positive and each force direction toward an origin is denoted as negative. From the goal position (6,6), the attractive force is symmetrically exerted as shown in Figure 5-26 (a). From the obstacle position (8,8), the repulsive force is symmetrically exerted as shown in Figure 5-26 (b). By addition of the attractive and repulsive forces, the total force is shown in Figure 5-26 (c), where the total forces from (14,14) to (8.9,8.9) are negative, which causes a robot to move toward an obstacle. We denote the transitional point having zero total force by $\mathbf{p}_\delta$. Note that a robot moves toward an obstacle from the starting point (12,12), and pass by the points (11,11), (10,10) and (9,9). From the point (9,9), a robot is still exerted by negative total force; and thus it continuously moves toward an obstacle and collides with the obstacle positioned at (8,8). Note that we are not trying to tackle the common trap situation due to the relatively bigger repulsive force. We restrict

our attention to the problem of collision by aligned robot-obstacle-goal (CAROG).

The CAROG problem has been not addressed yet, and it should be taken into account to prevent a robot from damage by collision. We define the distance between two points, $\mathbf{p}_\delta$ and $\mathbf{p}_o$ by

$$\delta = |\mathbf{p}_\delta - \mathbf{p}_o|. \tag{5.23}$$

If δ is larger than the distance $s_r \cdot T_s$, the collision is prevented. In the previous example, δ is 1.27, and $s_r \cdot T_s$ is $\sqrt{2}$. We deal with the CAROG prevention method by presenting a new force function as well as the method for finding the shortest path to a goal.

Total Force Modification and Oscillation Escape

The CAROG problem arises because an obstacle blocks both a goal and a robot in a straight line, and $|\mathbf{F}_{att}(\mathbf{p}_r)| > |\mathbf{F}_{rep}(\mathbf{p}_r)|$ when $|\mathbf{p}_r - \mathbf{p}_0| \leq s_r \cdot T_s$. It is found that if δ is larger than $s_r \cdot T_s$, the collision is avoided by having $|\mathbf{F}_{att}(\mathbf{p}_r)| < |\mathbf{F}_{rep}(\mathbf{p}_r)|$ when $|\mathbf{p}_r - \mathbf{p}_0| \leq s_r \cdot T_s$. This motivates us to consider a new total force as

$$\mathbf{F}_{tot}(\mathbf{p}) = \begin{cases} \nabla\rho(\mathbf{p}, \mathbf{p}_o), & \text{where } |\mathbf{p} - \mathbf{p}_o| \leq s_r \cdot T_s, \\ & \text{if } \ \delta < s_r \cdot T_s \\ & \& \ \nabla\rho(\mathbf{p}, \mathbf{p}_o) = \nabla\rho(\mathbf{p}, \mathbf{p}_g) \\ \mathbf{F}_{att}(\mathbf{p}) + \mathbf{F}_{rep}(\mathbf{p}), & \text{elsewhere.} \end{cases} \tag{5.24}$$

In comparison with (5.15), the new total force ensures that a robot moves away

from an obstacle before it is collided with an obstacle. In the previously shown example, since δ is 1.27 and $s_r \cdot T_s$ is $\sqrt{2}$, $\mathbf{F}_{tot}(\mathbf{p})$, where $|\mathbf{p} - \mathbf{p}_o| \leq \sqrt{2}$, becomes $\nabla\rho(\mathbf{p}, \mathbf{p}_o)$. Figure 5-27 shows the total force based on (5.24). Then, when a robot moves from the position (10,10) to the position (9,9), it moves back to the position (10,10), and the collision is avoided.

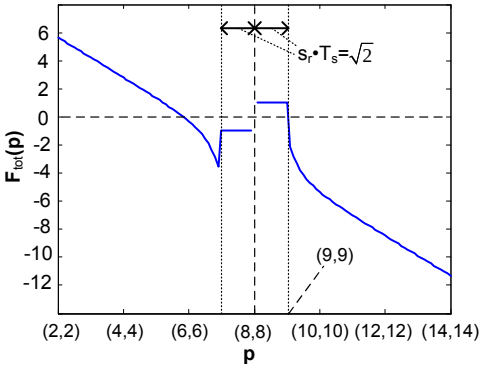


Figure 5-27: Total force based on (5.24)

As the new total force in (5.24) is considered, each size of robot and an obstacle becomes an important parameter. We assume that a robot and an obstacle are circles in shape with radius r_r and r_o, respectively. Then, the distance, δ, for an obstacle avoidance is extended from $s_r \cdot T_s$ to $s_r \cdot T_s + r_r + r_o$. Also, the new total force is revised as

$$\mathbf{F}_{tot}(\mathbf{p}) = \begin{cases} \nabla\rho(\mathbf{p}, \mathbf{p}_o), \text{ where } |\mathbf{p} - \mathbf{p}_o| \leq s_r \cdot T_s + r_r + r_o, \\ \qquad\qquad \text{if } \ \delta < s_r \cdot T_s + r_r + r_o \\ \qquad\qquad \& \ \nabla\rho(\mathbf{p}, \mathbf{p}_o) = \nabla\rho(\mathbf{p}, \mathbf{p}_g) \\ \mathbf{F}_{att}(\mathbf{p}) + \mathbf{F}_{rep}(\mathbf{p}), \text{ elsewhere.} \end{cases} \tag{5.25}$$

Though the new total force described in (5.25) guarantees the CAROG prevention,

a subsequent problem arises. For example, when the robot moves back to the position (9,9) from the position (10,10) in the previous example, it again moves back and forth between the two positions (9,9) and (10,10). It is an oscillated robot movement problem, which a robot never reach to a goal by oscillating between two positions. In order to reach to a goal, a robot should be out of the oscillated line track as illustrated in Figure 5-28.

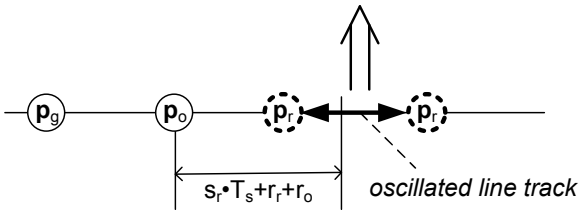


Figure 5-28: Illustration of oscillated line track

Since the two successive total forces are with an opposite direction, the oscillated robot movement is recognized at time t when

$$\mathbf{F}_{tot}^{t-1}(\mathbf{p}_r) \cdot \mathbf{F}_{tot}^{t}(\mathbf{p}_r) = -|\mathbf{F}_{tot}^{t-1}(\mathbf{p}_r)| \cdot |\mathbf{F}_{tot}^{t}(\mathbf{p}_r)| \tag{5.26}$$

where $t-1$ and t denotes time. Then, a robot deviates the oscillated line track at time t. Figure 5-29 illustrates the CAROG prevention and oscillation escape. Especially, in order to escape the oscillated line track in *step 3*, two possible cases are considered: deviate the oscillated track line when a robot is inside the region $|\mathbf{p} - \mathbf{p}_o| \leq s_r \cdot T_s + r_r + r_o$ (*step 3.1*) or outside (*step 3.2*).

We denote the deviation angle for the oscillated robot movement by θ_{osc} as illustrated in Figure 5-30. Also, we denote the distance between center points of an

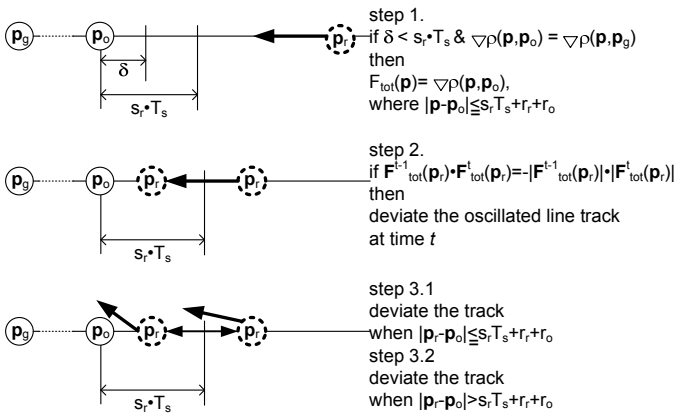


Figure 5-29: Illustration of CAROG prevention and oscillation escape

obstacle and a robot by d_{osc}. The distance, d_{osc}, is

$$0 < d_{osc} \leq s_r \cdot T_s + r_r + r_o, \tag{5.27}$$

when a robot deviates the oscillated line track from the point inside the region $|\mathbf{p} - \mathbf{p}_o| \leq s_r \cdot T_s + r_r + r_o$, and

$$s_r \cdot T_s + r_r + r_o < d_{osc} \leq 2(s_r \cdot T_s) + r_r + r_o, \tag{5.28}$$

when a robot deviates the oscillated line track from the point outside the region $|\mathbf{p} - \mathbf{p}_o| > s_r \cdot T_s + r_r + r_o$.

Since the relationship between θ_{osc} and d_{osc} is established as

$$\frac{r_r + r_o}{\sin\theta} = d_{osc}, \tag{5.29}$$

$$\theta_{osc} = \csc\left(\frac{r_r + r_o}{d_{osc}}\right), \tag{5.30}$$

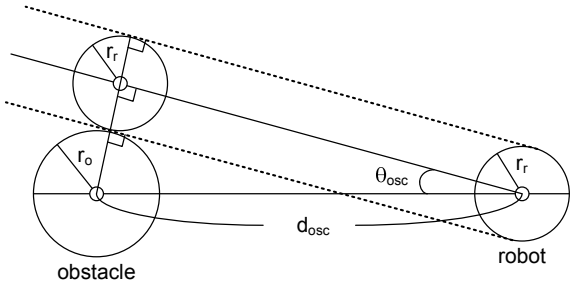


Figure 5-30: Deviation angle for the oscillated robot movement by θ_{osc}

the angle, θ_{osc}, has ranges as

$$\csc\left(\frac{r_r + r_o}{s_r \cdot T_s + r_r + r_o}\right) \leq \theta_{osc} < \frac{\pi}{2}, \tag{5.31}$$

when a robot deviates the oscillated line track from the point inside the region $|\mathbf{p} - \mathbf{p}_o| \leq s_r \cdot T_s + r_r + r_o$, and

$$\csc\left(\frac{r_r + r_o}{2(s_r \cdot T_s) + r_r + r_o}\right) \leq \theta_{osc} < \csc\left(\frac{r_r + r_o}{s_r \cdot T_s + r_r + r_o}\right), \tag{5.32}$$

when a robot deviates the oscillated line track from the point outside the region $|\mathbf{p} - \mathbf{p}_o| > s_r \cdot T_s + r_r + r_o$.

Note that the range of distance, d_{osc}, is known, but the exact position is unknown. Thus, in order to completely avoid the collision with an obstacle, we always should consider the maximum value of d_{osc}. Then, the deviation angle for the oscillated robot movement, θ_{osc}, should be chosen as a maximum value of (5.31) and (5.32) according to each case. In other words, the maximum values of θ_{osc}, $\pi/2$ and $\csc\left((r_r + r_o)/(s_r \cdot T_s + r_r + r_o)\right)$, support the shortest path to a goal as well as the obstacle avoidance, respectively.

We denote the unit vector of total force $\mathbf{F}_{tot}^{t}$ at time t as $\mathbf{u}_{F_{tot}}^{t}$, and its angle in a xy- plane as $\theta_{F_{tot}}^{t}$. Then, $\mathbf{F}_{tot}^{t}$ is expressed as

$$\mathbf{F}_{tot}^{t} = |\mathbf{F}_{tot}^{t}|\mathbf{u}_{F_{tot}}^{t} \tag{5.33}$$

$$= |\mathbf{F}_{tot}^{t}| \left(\cos(\theta_{F_{tot}}^{t})\mathbf{u}_x + \sin(\theta_{F_{tot}}^{t})\mathbf{u}_y\right) \tag{5.34}$$

5.2.3 Object Tracking with RFID and Visual Sensors Association and Data Traffic Analysis

Localization and tracking of multiple objects have been great interest to numerous surveillance-required areas. Tracking system is applied in diverse fields such as military, hospital and mining. However, the reliable and robust tracking is hardly accomplished due to unexpected trajectory and diversified environmental errors to be adapted. As a widely used tracking example, the Global Positioning System (GPS) utilizes a constellation of satellites which broadcasts precise timing signals based on a receiver's call. However, in contrast to GPS, tracking in wireless sensor network without a receiver calling is more difficult and challengeable.

For tracking objects, acoustic sensors have been widely used in many applications. Acoustic sensors have flexibility, low cost and easiness of deployment. However, an acoustic sensor is not only sensitive to surrounding environment with noisy data, but also not fully satisfying the requirement of consistent data. Thus, the substitution for acoustic sensor is necessary for more consistent and reliable data. Among a variety of sensors, visual sensors support consistent and reliable data for localization. However,

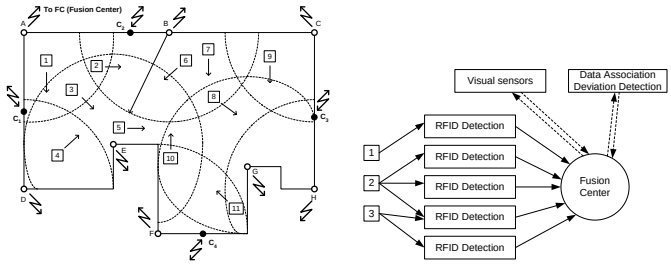
the disadvantage of visual sensor is that the roughly estimated position of observed objects should be known in advance.

Our objective is to propose object tracking algorithm based on association with RFID coverage scheme and a visual sensor. Firstly, we estimate coarse location of objects with multiple sensor nodes with a RFID reader. Secondly, the location coordinate determination is further improved with visual data compensation. In the RFID sensor nodes, objects with proximity to RFID readers are detected. Especially, redundant detection from different RFID readers is inferred that the object is positioned in the common coverage of the readers. Hence, the concept of virtual sensors is applied for the localization [74]. A virtual sensor is a RFID readers combination. In addition, each virtual sensor has a reference point which represents the central point of the overlapped coverage from RFID readers. Visual sensors are compensating for the coarse estimation based on parallel projection model. The visual compensation requires a reference point for improved estimation. The reference points are obtained from RFID coverage scheme. Finally, we present the experimental results; localization with RFID sensor nodes, visual compensation with one reference point and multiple reference points.

Background and Motivation

Figure 5-31 illustrates our proposed system model for tracking objects. RFID sensors and visual sensors are placed in a closed space with association. In Figure 5-31(a), the circles with A to H and the dots with C_1 to C_4 represent RFID sensors and visual sensors, respectively. All of the sensors are tracking objects (squares 1 to 11)

by accomplishing the coarse and the refined localization in order. Figure 5-31(b) illustrates the flow from RFID detection to visual sensing. Multiple RFID readers detect moving objects and accomplish the coarse estimates. The estimated position is determined using a virtual sensor with a reference point which will be discussed. After the coarse localization, image frames from two cameras improve the estimation based on parallel projection model. For instance from Figure 5-31(a), the object with ID 2 is detected by reader A, B and E. The global range is estimated based on common range of three circles from readers A, B and E. Finally, the refined position is determined by visual sensors C_1 and C_2 with improvement.



(a) Objects (1∼11) with RFID sensors (A∼H) coverage scheme and visual sensor (C_1∼C_4)

(b) Tracking assistant of RFID detection for visual sensors and/or particle filter

Figure 5-31: System model with RFID coverage scheme for tracking.

For the previous work on tracking with RFID, self-localization with a mobile robot is accomplished [72] [73] with visual sensors. However, the self-localization with a mobile robot is sensitive to a variety of environments.

Our proposed method tracks objects with sensor nodes in wireless network. Thus, the approach is independent of the environment and easy to implement. However, for the complete tracking, RFID collision problem should be considered. Especially in

dense RFID readers environment, overlapping of RFID coverage prevents readers from tags detection due to interference [75]. In order to solve the problem, [76] proposes coloring solution, and [77] proposes fine-tuning method with several frequencies or channels.

Incorporated RFID coverage scheme

In dealing with multiple objects tracking, data association is a problem of great importance. Recently the particle filter based estimation has addressed to the data association proposing a variety of strategies [78] [79]. Consider single object state vector $\mathbf{X}_n$ evolving according to

$$\mathbf{X}_n = f_{n-1}(\mathbf{X}_{n-1}) + \mathbf{Q}_{n-1}, \tag{5.35}$$

where f_n is a nonlinear, state transition function of the state $\mathbf{X}_n$, and $\mathbf{Q}_{n-1}$ is the non-Gaussian, process noise in the interval time-instant between n and $n-1$. The measurements of the evolving target state vector is expressed as

$$\mathbf{Z}_n = h_n(\mathbf{X}_n) + \mathbf{E}_n, \tag{5.36}$$

where h_n is a nonlinear and time-varying function of the target state, $\mathbf{E}_n$ is the measurement error which is independent identically distributed white noise process.

In order to estimate target state vector, dynamic prior probability density function

(pdf) is obtained as

$$p(\mathbf{X}_n|\mathbf{Z}_{1:n-1}) = \int p(\mathbf{X}_n|\mathbf{X}_{n-1})p(\mathbf{X}_{n-1}|\mathbf{Z}_{1:n-1})d\mathbf{X}_{n-1}, \tag{5.37}$$

where $\mathbf{Z}_{1:n}$ represents the sequence of measurements up to time instant n, and $p(\mathbf{X}_n|\mathbf{X}_{n-1})$ is the state transition density with Markov process of order one related to $f_n(\cdot)$ and $\mathbf{Q}_{n-1}$ in Equation (5.35).

For the next time estimation based on Bayes' rule, posterior pdf involving prediction pdf is obtained as

$$p(\mathbf{X}_n|\mathbf{Z}_{1:n}) = \frac{p(\mathbf{Z}_n|\mathbf{X}_n)p(\mathbf{X}_n|\mathbf{Z}_{1:n-1})}{\int p(\mathbf{Z}_n|\mathbf{X}_n)p(\mathbf{X}_n|\mathbf{Z}_{1:n-1})d\mathbf{X}_n}, \tag{5.38}$$

where $p(\mathbf{Z}_n|\mathbf{X}_n)$ is a likelihood function and the denominator is the normalizing constant. The measurement $\mathbf{Z}_n$ modifies the prior density (5.37) to obtain the current posterior density (5.38), which extends to multiple objects posterior pdf is expressed as:

$$p_{\tau_n|\Sigma_{1:n}}(\mathbf{X}_n|\mathbf{Z}_{1:n}) = \frac{p_{\Sigma_n|\tau_n}(\mathbf{Z}_n|\mathbf{X}_n)p_{\tau_n|\Sigma_{1:n-1}}(\mathbf{X}_n|\mathbf{Z}_{1:n-1})}{\int p_{\Sigma_n|\tau_n}(\mathbf{Z}_n|\mathbf{X}_n)p_{\tau_n|\Sigma_{1:n-1}}(\mathbf{X}_n|\mathbf{Z}_{1:n-1})d\mathbf{X}_n},$$

where Σ_n is the random set of K observations received at time n representing $\{\mathbf{Z}_n^1$, ..., $\mathbf{Z}_n^K\}$, and τ_n is the random set of K state vectors at time n representing $\{\mathbf{X}_n^1$, ..., $\mathbf{X}_n^K\}$. The random sets have multiple objects prior $p_{\tau_n|\Sigma_{1:n-1}}(\mathbf{X}_n|\mathbf{Z}_{1:n-1})$ and posterior pdf $p_{\tau_n|\Sigma_{1:n}}(\mathbf{X}_n|\mathbf{Z}_{1:n})$ with multiple observation likelihood function $p_{\Sigma_n|\tau_n}(\mathbf{Z}_n|\mathbf{X}_n)$.

The RFID coverage scheme reduces the data association computation by recogniz-

ing each object identification. By classifying objects based on different RFID readers detection, the multiple posterior pdf is simplified as:

$$p_{\tau_n^{(i)}|\Sigma_{1:n}}(\mathbf{X}_n|\mathbf{Z}_{1:n}) = \frac{p_{\Sigma_n|\tau_n^{(i)}}(\mathbf{Z}_n|\mathbf{X}_n)p_{\tau_n^{(i)}|\Sigma_{1:n-1}}(\mathbf{X}_n|\mathbf{Z}_{1:n-1})}{\int p_{\Sigma_n|\tau_n^{(i)}}(\mathbf{Z}_n|\mathbf{X}_n)p_{\tau_n^{(i)}|\Sigma_{1:n-1}}(\mathbf{X}_n|\mathbf{Z}_{1:n-1})d\mathbf{X}_n},$$

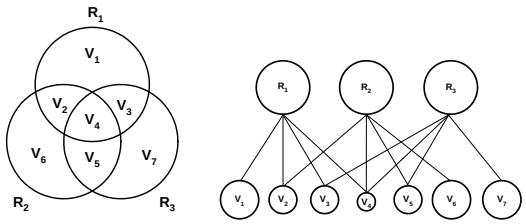
where $\tau_n^{(i)}$ is a subset of τ_n representing objects placed in reader coverage i (i = 1, 2, ..., I for the number of readers). For simplicity, we assumed that the number of objects N_n^i in coverage i is equal to N_{n-1}^i in order to validate the evolving pdf $p_{\tau_n^{(i)}|\Sigma_{1:n}}(\mathbf{X}_n|\mathbf{Z}_{1:n})$.

Visual sensor association

Turning now to an object tracking, visual sensors are associated with readers detection. Coarse localization with readers and refined localization with visual association are presented.

For coarse localization, each divided range from each RFID reader detection coverage should be considered as shown in Figure 5-32(a). In order to effectively utilize all overlapped ranges, the concept of virtual sensor nodes is introduced. A virtual sensor is applied to a wide range of field in order to figure out complicated structure [74]. In addition, it helps us approach problems in a graphical model which is viewed at a glance. Figure 5-32(b) shows a graphic model of virtual sensors based on each reader coverage including the overlapped area. The size of circles (v_1 to v_7) represents the coverage size. In the same method, Figure 5-31(a) is restated that 26 virtual sensors are tracking 11 targets (1 to 11); externally, 8 deployed RFID readers

(A to H) are tracking 11 targets. The 26 virtual sensors are created by separation and overlapping of 11 RFID reader coverages. In the coarse localization, each virtual sensor has a reference point which represents a center of rough range. By conversion of a large scale with RFID coverage to each reference point of a virtual sensor, coarse localization is accomplished.



(a) separation of overlapping

(b) Representation as virtual sensors

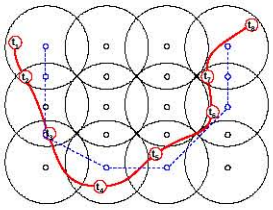
Figure 5-32: RFID coverages and virtual sensors (RFID coverages : R_1 to R_3, virtual sensors : v_1 to v_7).

Figure 5-33 illustrates the RFID readers detection with object trajectory. According to time, moving objects are detected by 12 different readers as shown in Figure 5-33(b). Under the assumption in which the overlapping area is generated by two readers maximally, the number of possible combination of 12 readers is ${}_{12}C_1 + {}_{12}C_2$ or 78; the numbers of covered ranges by one reader and two readers are ${}_{12}C_1$ and ${}_{12}C_2$, respectively. However, the readers combinations with long distance may be excluded even though the RFID coverage pattern is irregular and changed with time in real environment. Thus, in the total k RFID readers environment with p maximum coverage overlapping, the number of possible virtual sensors is equal to or less than $\sum_{i=1}^{p} {}_kC_i$. Tracking accuracy is in inverse proportion to not only RFID coverage but

also distance between adjacent readers. The small RFID coverage and short distance between adjacent readers increase tracking accuracy while deployed readers density increases.

Object	Detection Reader according to Time			
O_1	t_1	t_2	t_3	t_4
	R_1	R_1 R_3	R_5 R_9	R_{10}
	t_5	t_6	t_7	t_8
	R_{11}	R_8	R_4 R_8	R_4

(a) RFID readers detection



(b) estimate: dash line, real: solid line

Figure 5-33: RFID readers detection and global estimation by virtual sensor nodes from RFID coverage.

After the coarse range tracking with virtual sensor nodes based on the RFID detection, two visual sensors take object snapshots for refined localization. The estimate improvement is based on the parallel projection model as shown in Figure 5-34. The global estimation from RFID detection and real object position is denoted as E(x_e, y_e) and P, respectively. Virtual viewable planes are the image frames with parallel projection of a real object; the parallel planes to Z_1 or Z_2. The object planes (Z_1' and Z_2') are the parallel planes with respect to each virtual viewable plane. Basically, the estimated position (E_v') with visual sensors is expressed as

$$E_v' = (x_e \pm \Delta u_2, y_e \pm \Delta u_1), \tag{5.39}$$

where the sign $\pm$ is determined according to the relative object position (P) with respect to coarse estimated position (x_e and y_e)

The estimate conversion scheme in Equation (5.39) is illustrated in Figure 5-34. At first, projections to virtual viewable planes of the coarse estimate and the real object are from different height with Δd_1 or Δd_2. Thus, the estimation should be modified by considering the parameters d_1, Δd_1, d_2 and Δd_2. In the ratio of triangles similitude, the Equation (5.39) is modified as

$$\begin{aligned} E_v &= (x_e \pm \Delta u_{r2}, y_e \pm \Delta u_{r1}), \\ \Delta u_{r1} &= \Delta u_1(\frac{d_1 + \Delta d_1}{d_1})(\frac{Z_1}{Z_1'}), \\ \Delta u_{r2} &= \Delta u_2(\frac{d_2 + \Delta d_2}{d_2})(\frac{Z_2}{Z_2'}), \end{aligned} \tag{5.40}$$

where E_v represents the final estimation with visual sensors, and the sign $\pm$ is determined according to the relative object position (P) with respect to the global estimated position (x_e and y_e).

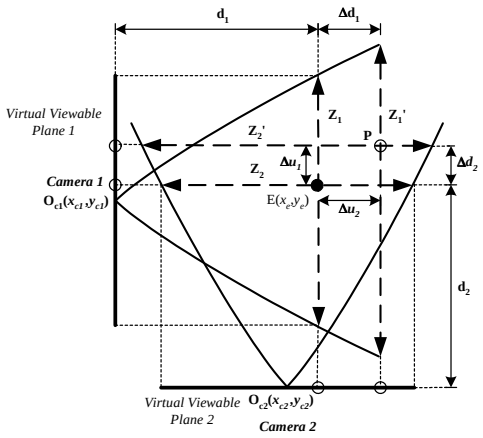


Figure 5-34: The parallel projection model in visual sensors.

The other factor for consideration is scale distortion. The parallel projection model estimates position under the assumption in which the sight of vision boundary

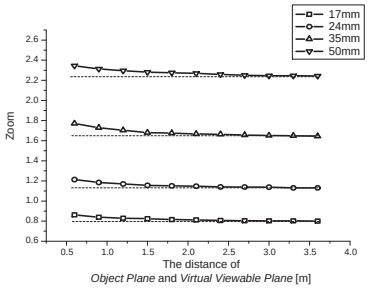


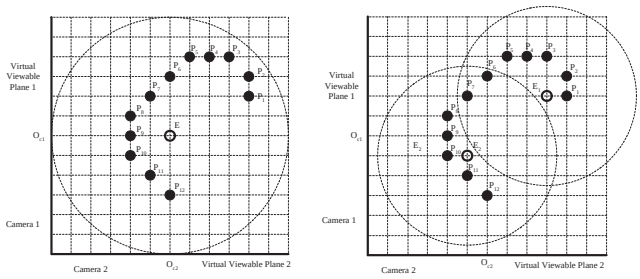
Figure 5-35: Distortion error for nonlinear sight of vision boundary.

is linearly drawn as solid lines in Figure 5-34. However, a real camera supports the nonlinear boundary which results in scale distortion. This non-linearilty contributes to the difference in Z and Z'. Figure 5-35 shows the zooming factor distortion error which is measured by *Canon Digital Rabel XT with Tamron SP AF17-50mm Zoom Lens*. The data shows the error according to the distance between a object plane and a virtual viewable plane. As the planes distance increases, the error is found to be decreased with zooming discrepancy reduction.

Simulation and Analysis

Figure 5-36 shows object trajectory (P_1 to P_{12}) with different readers environment. Two visual sensors (O_{c1}, O_{c2}) are placed in the middle of left and bottom wall. Figure 5-36(a) illustrates an object tracking with one reference point from single RFID reader. Figure 5-36(b) illustrates an object tracking with two reference points from two RFID readers. Virtual sensors based on RFID detection results in estimating a point E, E_1 or E_2 relying on the belonged coverage. From the coarse estimation, the point E, E_1 or E_2 becomes a reference point for visual sensing.

Figure 5-37 shows the visual compensation results from global localization in Fig-



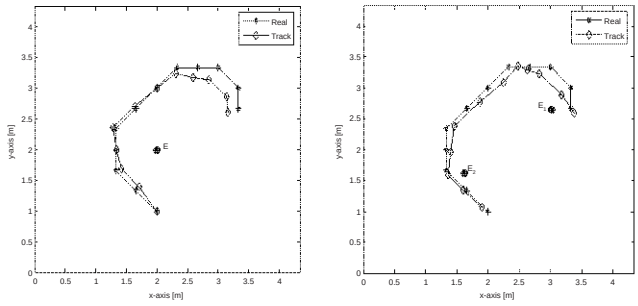
(a) Object trajectory in single RFID coverage.

(b) Object trajectory in two RFID coverage.

Figure 5-36: Simulation setup based on RFID detection (the coarse estimation with RFID is a point E in (a) and E_1 or E_2 in (b)).

ure 5-36(a) and 5-36(b). From the simulation results, improved estimates with visual sensors are shown with the refined tracking. The tracking accuracy is almost no difference between one and two reference points having visual compensation. Since passive tags have practical read distances ranging from approximately 4 inches up to a few meters, visual sensing refines estimation with similar accuracy to any object. In other words, visual sensing compensation results in almost same tracking accuracy unless RFID detection range is scores of meters,

The accuracy of the coarse estimation with RFID readers is dependent on the number of readers and the adjacent readers distance. The more readers are placed, the more accurate the estimation is. However, the refined estimation with visual sensors improves any coarse estimation similarly regardless of error difference. Thus, the final estimation with visual sensors has little effect on the number of RFID readers as well as distance between adjacent readers.



(a) Visual compensation with single reference point.

(b) Visual compensation with two reference points.

Figure 5-37: Visual compensation result after the coarse estimation with RFID detection based on Figure 5-36.

Data Traffic Analysis

As time passes by, demanding degree of tracking accuracy is increased while existent sensors and tracking algorithm still have limitation. The acoustic sensors, RFID and visual sensors are widely used in many applications that needs the localization of sensors. However, an acoustic sensor is not only sensitive to surrounding environment with noisy data, but also requires endless sound wave from sources. RFID localization accomplishes only approximate estimation with specific coverage [80]. A visual sensor requires initial position in advance. In other words, only single sensor is not enough to manage accurate tracking in diversified environments. Thus, fusion sensors association is required for object position compensation. However, in the fusion sensor network, data traffic may stir up the whole tracking performance, and data packets are dropped in heavy traffic network.

Our objective is to present fusion sensor association with our proposed network protocols. The data traffic is investigated and analyzed in wireless sensor network

quipped with mesh routers [81]. An acoustic sensor and RFID reader are incorporated in a sensor node. Based on RFID detection, acoustic sensors determine which objects are taken charged. The acoustic sensor detecting Direction of Arrival (DOA) estimates objects position based on Particle Filtering as an example of tracking algorithm [16] [82]. Since the weakness of the acoustic sensors and the tracking algorithm, the estimation is refined with visual sensors in a final stage [17]. For the sensors data association, routers and a server are connected composing the whole wireless fusion sensor network. Based on the network protocol construction, we analyze the network traffic mixed by multiple types of fusion data by using NS-2 simulations under various network scenarios. We mention the feasible placement of the functionalities for object tracking with fusion sensors by observing the end-to-end and hop-by-hop delays. In addition, we also get the insight how we configure the wireless fusion sensor network for more accurate object tracking.

Tracking with acoustic sensors based on traditional algorithms should be satisfied by a few conditions for complete performance. For example, the Kalman filter is not applied to nonlinear trajectory, and the Particle Filter requires initial position and velocity [6]. Furthermore, sound waves for acoustic sensors are transmitted sporadically due to blocks. For those limitations, additional data with visual sensors compensate the performance. RFID, acoustic sensors and visual sensors are cooperating with multimodal signal processing as shown in Figure 5-38.

A RFID reader and an acoustic sensor are assumed to be fused in a single sensor node (S_1 to S_3). Objects (O_1 to O_4) are detected with proximity to RFID readers and acoustic sensors determine which objects are taken charged based on the RFID

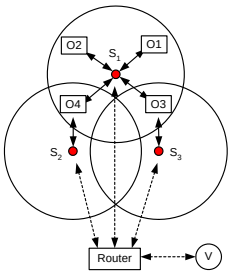
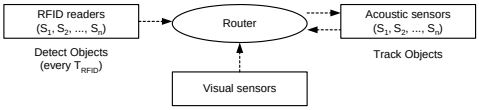


Figure 5-38: Fusion sensors with RFID detection, acoustic sensing and visual sensing (Object : O, RFID and acoustic sensor : S, visual sensor : V)

detection. After the base estimation from the acoustic sensors, visual sensors (V) refine the estimation for the final result.



(a) Fusion sensors association based on a router.

RFID	Detected Object ID
S_1	O_1, O_2, O_3
S_2	O_2, O_3, O_4, O_7
⋮	⋮
S_n	O_4, O_5, O_8, O_{10}

(a) RFID detection

Acoustic sensor	Object ID for sensing
S_1	O_1, O_2
S_2	O_3, O_7
⋮	⋮
S_n	O_4, O_8, O_{10}

(b) Acoustic sensing

(b) Re-arrangement for acoustic sensing based on RFID detection

Figure 5-39: RFID detection, acoustic sensing and visual sensing.

The sensors association is illustrated in Figure 5-39(a). Based on RFID detection, a router rearranges objects for each acoustic sensor as shown in Figure 5-39(b). Object positions based on acoustic sensing are also estimated in a router by using particle filtering algorithm. Furthermore, the refined estimation with visual sensors may be

processed in a router, or a server in lieu of a router may estimate the refined estimation after receiving the image frames containing snapshot of space from visual sensors. The estimation with visual sensing will be more discussed in Chapter 5.2.3 with respect to network performance. In summary for tracking with multiple sensors, DOA (Direction of Arrival) from acoustic sensors are two angle components which are azimuth angle θ, elevation angle ϕ between a sensor and an object [82]. Basic algorithm with the acoustic data is bearings that is applied in tracking of an object in 2-D plane with one angle measurement based on a particle filter. In a single sensor node, the 2-D tracking results are extended for 3-D result using 2-D planes combination [16]. The estimation is refined by visual sensors by parallel projection model [45]. The parallel projection model improves estimation by comparing unreliable estimation with real object position from two image frames.

For a large space, multiple routers and a server need to manage and control the overall tracking system. In a network aspect, data traffic with transmission delay should be considered. Figure 5-40 illustrates the overall network topology with multiple routers and a server in a sensor network, where the routers are configured as wireless mesh backbone. A sensor node equipped by a RFID reader and an acoustic sensor communicates with a router which is connected to a server. Visual sensors are connected to each router as well. However, the visual sensors are not required for connecting to every single router since visual sensors such as cameras sufficiently accomplish the wide range sensing so that it may capture the whole tracking space in a single image frame. For example, only two visual sensors are used in the parallel projection model in [45]. A sensor node for RFID detection with acoustic sensing,

a visual sensor, and a router are denoted as S_{ij} and V_i, R_i, respectively. Here, i^{th} represents router ID and j is a sensor ID.

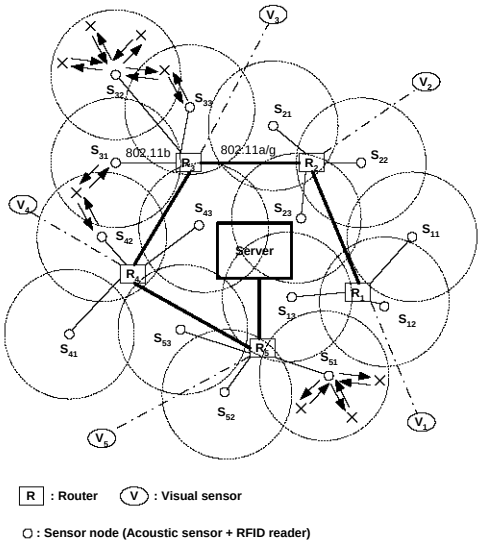


Figure 5-40: Fusion sensor network model.

We propose the networking protocols to support the object tracking with multiple types of sensors. In order to simplify the network complexity, we reconfigure Figure 5-40 to string scenario as shown in Figure 5-41. In order to accomplish feasible network environment, multiple channels are assumed to be utilized such that different channels are used for different group of sensors and a router. Therefore, acoustic sensors incorporated with RFID readers, routers, visual sensors and a sever communicate each other in more reliable manner by eliminating channel interference. In this environment, two possible communication protocols should be considered with analysis: one is the router base visual compensation (RBVC) and the other one is the server base visual compensation (SBVC). In RBVC, the routers process the visual compensation as well as perform the sensor management and particle filtering. To do

this, the image frames should to be distributed to all the routers. In case of SBVC, the parallel projection model is conducted in a server so that the estimated object positions from acoustic sensors and image frames from visual sensors are sent to the server. Note, in the sensor delivery, an image frame is range of dozens of Kbytes to a few Mbytes, so that the visual compensation is mainly affected by the image transferring. The visual estimation process is not necessary but assistant for estimate compensation in the case of tracking deviation from acoustic sensing because visual data is more reliable than acoustic data. Thus, the estimations based on visual sensing and acoustic sensing have different sampling times. Our proposed protocols

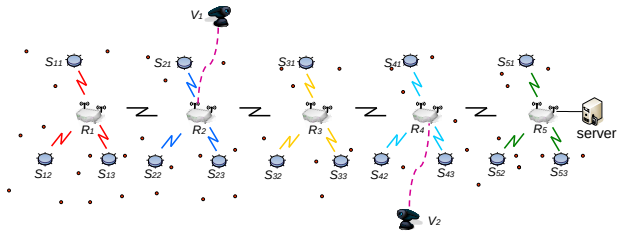


Figure 5-41: String scenario for wireless fusion sensor network. Different color represents different channel.

require following achievements. At first, a server should have final estimated positions in order to control all system. Secondly, a router manages acoustic data based on RFID detection on its own. Thirdly, in the case processing visual compensation, a router needs final estimated positions in order to estimate next position based on the particle filtering algorithm which generates the possible particles with compensated estimation for the next estimates. At last, visual compensation requires the estimation from acoustic sensing and the image frames from at least two cameras.

Figure 5-42 illustrates the router base visual compensation (RBVC) protocol in

which the compensation with visual sensors is processed in routers. The points E and E' represent the compensation processes with acoustic data and image frames, respectively. *RFID detection data* is transmitted to a router which determines target objects among the numerous objects within acoustic sensing range. A router transmits *assigned object data* to acoustic sensors receiving *acoustic sensing data*. A router estimates the objects positions based on the *acoustic sensing data* by using particle filtering algorithm. In the meantime, *visual sensing data* (image frame) is sent to a router. The data is transmitted with several packets with ACK since the image frame should be provided with reliable transmission. Based on the *visual sensing data* and the estimation from acoustic sensing, the final estimated position is determined. In case the *visual sensing data* is not sent, the acoustic sensing estimation is the final estimated position. Finally, the *final estimation data* is sent to a server based on the first protocol requirement.

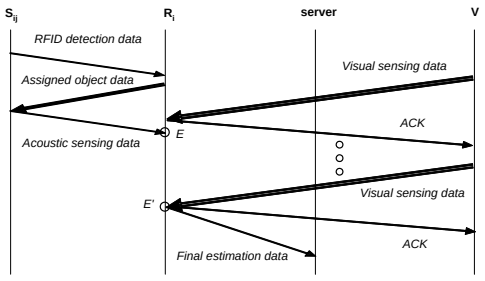


Figure 5-42: Router base visual compensation (RBVC)

In contrast to the RBVC protocol, the server base visual compensation (SBVC) protocol is to lift a router burden to a server in estimating visual compensation as shown in Figure 5-43. The transmission between RFID-acoustic sensors and a router is the same as RBVC protocol. After estimating object positions in a router, the

estimation based on acoustic sensors is sent to a server. In the meantime, *visual sensing data* is transmitted to a server with the acknowledge transmissions. The server estimates the final objects positions based on the *estimation based on acoustic sensors* and the *visual sensing data*. In case the *visual sensing data* is not sent, the acoustic sensing estimation is the final estimated position similar to RBVC. Finally, for the third protocol requirement, the *final estimation data* is sent to each router.

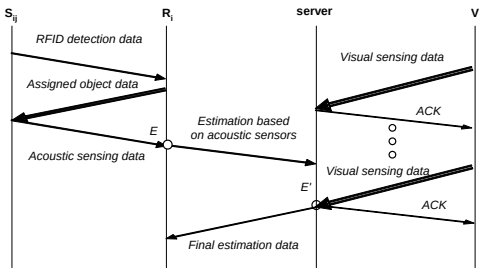
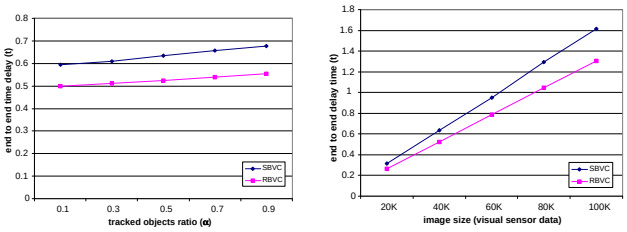


Figure 5-43: Server base visual compensation (SBVC)

In order to perform the traffic analysis, we conduct the extensive NS-2 simulations in this Chapter. Simulation setup is based on the network in Figure 5-40 with a server, 5 routers and 3 sensors for each router. 100 objects are distributed in each router (R_i) range and each RFID reader in sensor nodes (S_{i1}, S_{i2}, S_{i3}) are detecting objects such that S_{i1} detects 30 objects, S_{i2} detects 40 objects, and S_{i3} detects 50 objects, respectively. The objects assignment for acoustic sensing is that 30α, 30α and 40α objects for acoustic sensors (S_{i1}, S_{i2}, S_{i3}), respectively, where α is the ratio of the number of tracked objects with respect to the number of total objects. For example, if the α is 0.1, acoustic sensors S_{i1}, S_{i2}, S_{i3} receive acoustic data from 3, 3 and 4 objects. In this case, total 10 objects get tracked among 100 objects. Based on these objects distribution, three possible scenarios are examined from the simulations. The first

scenario is that 100 objects are distributed in only a router 1 (R_1) range, and the other routers have no objects. In addition, visual sensors are connected to a router 2 (R_2) and a router 4 (R_4). The second scenario is that 100 objects are distributed in each router range, that is, total 500 objects are distributed in the whole sensor network range. Similar to the first case, the visual sensors are connected to the same routers (R_2, R_4). In the third scenario, two visual sensors are simultaneously connected to a router 4 (R_2) which is the closest to a server. This scenario is devised to minimize the transmission of image frames from visual sensors to a server.

In this first scenario, the end to end delay is shown in Figure 5-44. As the observed objects ratio or the image frame size increases, the end-to-end delay increases as well. However, we understand the delay is more sensitive to the image frame size since the image is so large that it overwhelms the overall network traffic. As the number of observed objects increase, the delay difference is negligible as shown in Figure 5-44(a). We also figure out RBVC achieves lower end-to-end delay than SBVC.

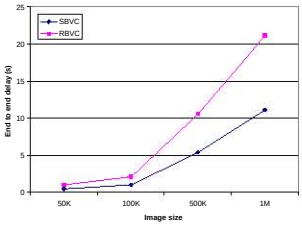


(a) The end to end delay according to the ratio of tracked objects (image size : 40k).

(b) The end to end delay according to visual data size.

Figure 5-44: Scenario 1: 100 objects are distributed only in a R_1 range and visual sensors are connected to routers R_2 and R_4.

The second and third scenario are more realistic than the first one as objects are

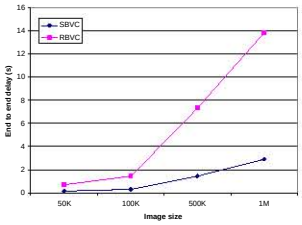


(a) The end-to-end delay according to visual data size.

Hop by hop	SBVC	RBVC
RFID detection data	0.001919	0.000615
Assigned object data	0.001625	0.001639
Acoustic sensing data	0.001288	0.001263
Visual sensing data	0.004149	0.008797
Visual sensing ACK	0.002909	0.010697
Sum of visual sensing	2.687739	6.998317
Final estimation data	0.009267	0.023861
Estimation based on acoustic sensors	0.014093	

(b) The hop-by-hop delay.

Figure 5-45: Scenario 2: 100 objects are distributed in each router R_i range for i=1,2,...,5. Total 500 objects are distributed, and visual sensors are connected to routers R_2 and R_4.



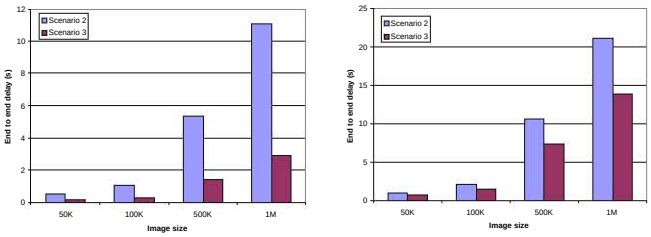
(a) The end-to-end delay according visual data size.

Hop by hop	SBVC	RBVC
RFID detection data	0.000624	0.001853
Assigned object data	0.001741	0.001620
Acoustic sensing data	0.001252	0.001281
Visual sensing data	0.001783	0.010335
Visual sensing ACK	0.001943	0.010114
Sum of visual sensing	1.413363	7.083349
Final estimation data	0.004588	0.010127
Estimation based on acoustic sensors	0.012266	

(b) The hop-by-hop delay.

Figure 5-46: Scenario 3: the same objects distribution as the scenario 2 except that two visual sensors are connected to R_5.

distributed in all routers range. In contrast to the first case, the end-to-end delay of SBVC is lower than RBVC since the large image frame should be distributed to all the routers in RBVC. This leads to heavy network traffic, consequently, large delays. Figure 5-45(b) and 5-46(b) illustrates this consequence, in which the hop-by-hop delay of visual sensing in RBVC is larger than that of SBVC. We find out the end-to-end delay in Figure 5-46(a) is lower than Figure 5-45(a) since the image frames in the third scenario are fast transmitted to the server. This fact is well described in Figure



(a) The end-to-end delay according visual data size.

(b) The hop-by-hop delay.

Figure 5-47: Visual positions dependence

5-47.

Bibliography

[1] C. H. Knapp and G. C Carter, "The generalized correlation method of estimation of time delay," in *IEEE Trans. Acoust., Speech, Signal Processing*, vol.24, no.4, pp.320-327, Aug. 1976.

[2] J. H. Dibiase, H. F. Silverman, and M. S. Brandstein, "Robust localization in reverberant rooms," *Microphone Arrays: Signal Processing Techiniques and Applications*, chapter 8, pp.157-180, Springer, 2001.

[3] N. Strobel, T. Meier and R. Rabenstein, "Speaker localization using steered filtered-and-sum beamformers", in *Proc. Erlangen Workshop on Vision, Modeling, and Visualization, Germany*, pp.195-202, 1999.

[4] J. Vermaak and A. Blake, "Nonlinear filtering for speaker tracking in noisy and reverberant environments," in *IEEE Int. Conf. Acoust., Speech, Signal Processing (ICASSP-01)*, pp.3021-3024, Salt Lake City, UT, May 2001.

[5] D. B. Ward and R. C. Williamson, "Particle filter beamforming for acoustic source localization in a reverberant environment," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing (ICASSP-02)*, pp.1777-1780, Orlando, FL, May 2002.

[6] W. R. Gilks and C. Berzuini, "Following a moving target – Monte Carlo inference for dynamic Bayesian models," in *Journal of the Royal Statistical Society, B*, vol.63, pp.127-146, 2001.

[7] P. M. Djurić, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo and J. Miguez, "Particle filtering," in *IEEE Signal Processing Magazine*, vol.20, no.5, pp. 19-38, Sep. 2003.

[8] M. S. Arulampalam and S. Maskell and N. Gordon and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," in *IEEE Trans. Signal Processing*, vol.50, no.2, pp.174-188, Feb. 2002.

[9] D. B. Ward, E. A. Lehmann, and R. C. Williamson "Particle filtering algorithms for tracking an acoustic source in a reverberant environment," in *IEEE Trans. Speech and Audio Processing*, vol.11, no.6, pp.826-836, Nov. 2003.

[10] J. M. Passerieux and D. V. Cappel, "Optimal observer maneuver for bearings-only tracking," in *IEEE Trans. Aerospace and Electronic Systems*, vol.34, no.3, pp.777-788, Jul. 1998.

[11] R. A. Iltis and K. L. Anderson, "A consistent estimation criterion for multisensor bearings-only tracking," in *IEEE Trans. Aerospace and Electronic Systems*, vol.32, no.1, pp.108-120, Jan. 1996.

[12] M. S. Arulampalam, B. Ristic, N. Gordon and T. Mansell, "Bearings-only tracking of manoeuvring targets using particle filters," in *EURASIP Journal on Applied Signal Processing*, vol.2004, no.1, pp.2351-2365, Jan. 2004.

[13] K. Dogancay and G. Ibal, "Instrumental variable estimator for 3D bearings-only emitter localization," in *Proc. IEEE Int. Conf. Intelligent Sensors, Sensor networks and information Processing*, pp.63-68, Dec. 2005.

[14] H. W. Tian, Z. L. Jing, S. Q. Hu, J. X. Li and H. Leung, "Tracking a 3D maneuvering target using high-rate bearings-only measurements" in *Proc. IEEE Int. Conf. Machine Learing and Cybenetics*, vol.2, pp.845-850, Aug. 2004.

[15] M. Stanacevic and G. Cauwenberghs, "Micropower gradient flow acoustic localizer,"in *IEEE Trans. Circuits and Systems I*, vol 52, no. 10, pp.2148-2157, Oct. 2005

[16] J. Lee, J. Lim, S. Hong and P. Park, "Tracking an object in 3-D space using particle filtering based on sensor array" in *Proc. The Sixth IEEE International Conference on Computer and Information Technology (CIT '06)*, p.242, Sep. 2006.

[17] S. Hong, J. Lee, A. athalye, P. Djuric and W. D. Cho, "Design Methodology for Domain Specific Parameterizable Particle Filter Realization,"in *IEEE Trans. Circuits and Systems I*, vol 54, no. 9, pp.1987-2000, Sep. 2007.

[18] M. Bolic, P. M. Djurić and S. Hong, "Resampling algorithms and architectures for distributed particle filters," in *IEEE Trans. Signal Processing*, vol.53, no.7, pp.2442-2450, Jul. 2005.

[19] B. Ristic, S. Arulampalam and N. gordon, *Beyond the Kalman filter: particle filter for tracking application*, Artech House Publishers, Boston and London, 2004.

[20] M. F. Bugallo, T. Lu and P. M. Djurić, "Target tracking by multiple particle filtering," in *Proc. IEEE Aerospace Conference*, pp.1-7, Mar. 2007.

[21] N. Vaswani, "Additive change detection in nonlinear systems with unknown change parameters," in *IEEE Trans. Signal Processing*, vol. 55, no. 3, pp.859-872, Mar. 2007.

[22] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson and P. Nordlund, "Particle filters for positioning, navigation, and tracking," in *IEEE Trans. Signal Processing*, vol.50, no.2, pp.425-437, Feb. 2002.

[23] B. Ristic and M.S. Arulampalam, "Tracking a manoeuvring target using angle-only measurements : algorithms and performance," in *Signal Processing*, vol. 83, no. 6, pp. 1223-1238, Jun. 2003.

[24] P. Tichavsky, C. H. Muravchik and A. Nehorai, "Posterior Cramer-Rao bounds for discrete-time nonlinear filtering," in *IEEE Trans. Signal Processing*, vol.46, no.5, pp.1386-1396, May, 1998.

[25] J. H. Taylor, "The Cramer-Rao estimation error lower bound computation for deterministic nonlinear systems," in *IEEE Trans. Automatic Control*, vol.24, no.2, pp.343-344, Apr. 1979.

[26] N. Siebel, "People Tracking for Visual Surveillance" *Ph.D dissertation, The University of Reading, UK*, March 2003.

[27] Y. Bar-Shalom and X. R. Li, *Estimation and Tracking: Principles*, Techniques and Software, Artech House, Norwood, Mass, USA, 1993.

[28] R. J. Kozick and B. M. Sadler, "Source Localization with Distributed Sensor Arrays and Partial and Spatial Coherence," in *IEEE Transactions on Signal Processing*, vol.52, no.3, pp.601-616, Mar. 2004.

[29] G. Jacovitti and G. Scarano, "Discrete Time Techniques for Time Delay Estimation," in *IEEE Transactions on Signal Processing*, vol.41, no.2, pp.525-533, Feb. 1993.

[30] D. D. Feldman and L. J. Griffiths, "A Projection Approach for Robust Adaptive Beamforming," in *IEEE Transactions on Signal Processing*, vol.42, no.4, pp.867-876, Apr. 1994.

[31] T. Kirubarajan, Y. Bar-Sralom and D. Lerro, "Bearings-only Tracking of Maneuvering Targets using Abatch-recursive Estimator," in *IEEE Transactions on Aerospace and Electronic Systems*, vol.37, no.3, pp.770-780, Jul. 2001.

[32] J. Lee, S. H. Cho, S. Hong and W. D. Cho, "Multitarget Tracking (MTT) in 3-D using 2-D Particle Filters with Single Passive Sensor," in *Proceedings of IEEE International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp.389-392, Aug. 2007.

[33] J. Lim, J. Lee, S. Hong and P. Park, "Algorithm for Detection with Localization of Multi-targets in Wireless Acoustic Sensor Networks," in *Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence*, pp.547-554, Nov. 2006.

[34] D. N. Zotkin, R. Duraiswami and L. S. Davis, "Joint Audio-Visual Tracking Using Particle Filters," in *EURASIP Journal on Applied Signal Processing*, vol.2002, no.1, pp.1154-1164, Jan. 2002.

[35] S. T. Shivappa, M. M. Trivedi and B. D. Rao, "Person Tracking with Audio-visual Cues using the Iterative Decoding Framework," in *Proceedings of IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance, (AVSS '08)*, pp.260-267, Sep. 2008.

[36] S. Dupont and J. Leuttin, "Audio-visual Speech Modeling for Continuous Speech Recognition," in *IEEE Transactions on Multimedia*, vol.2, no.3, pp.141-151, Sep. 2000.

[37] T. Chen, "Audiovisual Speech Processing," in *IEEE Signal Processing Magazine*, vol.18, no.1, pp.9-21, Jan. 2001.

[38] K. Chow, K. Lui and E. Y. Lam, "Efficient On-Demand Image Transmission in Visual Sensor Networks," in *EURASIP Journal on Applied Signal Processing*, vol.2007, no.1, p.225, Jan. 2007.

[39] D. K. Park, H. S. Yoon and C. S. Won, "Fast Object Tracking in Digital Video," in *IEEE Transactions on Consumer Electronics*, vol.46, no.3, pp.785-790, Aug. 2000.

[40] J. Carpenter, P. Clifford and P. Fearnhead, "An improved particle filter for nonlinear problems," in *IEE Proceedings - Radar, Sonar and Navigation*, vol.146, no.1, pp.2-7, Feb. 1999.

[41] A. Doucet, N. J. Gordon and V. Krishnamurthy, "Particle Filters for State Estimation of Jump Markov Linear Systems," in *IEEE Transactions on Signal Processing*, vol.49, no. 3, pp.613-624, Mar. 2001.

[42] Y. Boers and J. N. Driessen, "Interacting Multiple Model Particle Filter," in *IEE Proceedings - Radar, Sonar and Navigation*, vol.150, no.5, pp.344-349, Oct. 2003.

[43] R. Velmurugan, S. Subramanian, V. Cevher, D. Abramson, F. Odame, J. Gray, H. Lo, J. McClellan and D. Anderson, "On Low-Power Analog Implementation of Particle Filters for Target Tracking," in *Proceedings of 14th European Signal Processing Conference (EUSIPCO)*, Sep. 2006.

[44] M. Yeddanapudi, Y. Bar-Shalom and K. R. Pattipati, "IMM Estimation for Multitarget-Multisensor Air Traffic Surveillance," in *Proceedings of the IEEE*, vol.85, no.1, pp.80-96, Jan. 1997.

[45] K. S. Park, J. Lee, M. Stanacevic, S. Hong and W. D. Cho, "Iterative Object Localization Algorithm Using Visual Images with a Reference Coordinate," in *EURASIP Journal on Image and Video Processing*, vol.2008, pp.1-16, 2008.

[46] J. Lee, S. Hong, P. Park and W. D. Cho, "Object tracking based on RFID coverage and visual compensation in wireless sensor network," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS 2007)*, pp.1597-1600, May, 2007.

[47] M. Han, A. Sethi, W. Hua and Y. Gong, "A detection-based multiple object tracking method," in *Proceedings of the International Conference on Image Processing (ICIP '04)*, vol.5, no.24-27, pp.3065-3068, Oct. 2004.

[48] E. Hjelmas and B. K. Low, "Face Detection: A Survey," in *Computer Vision and Image Understanding*, vol. 83, pp.236-274, 2001.

[49] X. R. Li and V. P. Jilkov, "Survey of Maneuvering Target Tracking, Part I. Dynamic Models," in *IEEE Transactions on Aerospace and Electronic Systems*, vol.39, no.4, pp.1333-1364, Oct. 2003.

[50] G. Peremans, K. Audenaert and J. M. Van Campenhout, "A High-Resolution Sensor based on Tri-Aural Perception," in *IEEE Transactions on Robotics and Automation*, vol.9, no.1, pp.36-48, Feb. 1993.

[51] C. Hue, J-P. L. Cadre and P. Perez,' 'Tracking Multiple Objects with Particle Filtering," in *IEEE Transactions on Aerospace and Electronic Systems*, vol.38, no.3, pp.791-812, Jul. 2002.

[52] R. Feraud, O.J. Bernier, J.-E. Viallet and M. Collobert, "A Fast and Accurate Face Detection Based on Neural Network", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 1, pp. 42-53, Jan. 2001.

[53] R. Hsu, M. Mottaleb and A. Jain, "Face Detection in Color Images", *IEEE Transactions on Patterns on Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 696-706, May 2002.

[54] W. Hu, T. Tan, L. Wang and S. Maybank, "A Survey on Visual Survillance of Object Motion and Behaviors", *IEEE Transactions on Information Systems, Man, And Cybernetics - Part C: Applications and Reviews*, vol. 34, no. 3, pp. 334-352, Aug. 2004.

[55] S. Chien, Y. Huang, B. Hsieh, S. Ma and L. Chen, "Fast Video Segmentation Algorithm with Shadow Cancellation, Global Motion Compensation and Adaptive Threshold Techniques", *IEEE Transactions on Multimedia*, vol. 6, no. 5, pp. 732-748, Oct. 2004.

[56] Z. Chair and P. K. Varshney, "Optimal Data Fusion in Multiple Sensor Detection Systems", *IEEE Transactions on Aerospace and Electronic Systems*, vol.22, no.1, P. 98-101, 1986.

[57] M. Pollefeys, R. Koch and L. V. Gool, "Self-Calibration and Metric Reconstruction Inspite of Varying and Unknown Intrinsic Camera Parameters", *International Journal of Computer Vision*, vol.32, no.1, P. 7-25, Nov. 1999.

[58] Y. B. Shalom and W. D. Blair, *Multitarget-Multisensor Tracking: Applications and Advances*, Artech House Publishers, 2000, pp.78.

[59] W. Zhao, R. Chellappa, P. J. Phillips, A. Rosenfeld, "Face recognition: A literature survey", in *ACM Computing Surveys (CSUR)*, Vol. 35, Issue 4, 2003, pp.399-458.

[60] M. Kass, A. Witkin and D. Terzopoulos,"Snakes: Active contour models" *International Journal of Computer Vision*, pp. 321-331, Jan. 1988.

[61] D. Nguyen, D. Halupka, P. Aarabi, and A. Sheikholeslami, "Real-time face detection and lip feature extraction using field-programmable gate arrays," *IEEE Trans. Systems, Man and Cybernetics*, vol. 36, num. 4, pp. 902-912, Aug. 2006.

[62] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," *Proc. IEEE computer Society Conference on Computer Vision and Pattern Recognition*, pp. 586-591, Jun. 2001.

[63] P. Veelaert and W. Bogaerts, "Ultrasonic Potential Field Sensor for Obstacle Avoidance", in *IEEE Transactions on Robotics and Automation*, vol. 15, pp. 774-779, Aug. 1999.

[64] B. Damas, P. Lima and L. Custodio, "A Modified Potential Fields Method for Robot Navigation Applied to Dribbling in Robotic Soccer", in *Proceddings of RoboCup 2002 Symposium*, Fukuoka, Japan, 2002.

[65] J. Latombe, *Robot Motion Planning.*, Norwell, MA: Kluwer, 1991.

[66] R. Tilove, "Local Obstacle Avoidance for Mobile Robots Based on the Method of Artificial Potentials", in *Proceedings of the IEEE Confernce on Robotics and Automation, Cincinnati, Ohio*, May, 1990, pp.566-571.

[67] Y. Koren and J. Borenstein , "Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation", in *Proceedings of the IEEE Confernce on Robotics and Automation, Sacramento, California*, April, 1991, pp.1398-1404.

[68] S. Ge and Y. Cui, "Dynamic Motion Planning for Mobile Robots Using Potential Field Method", in *Autonomous Robots*, vol. 13, pp. 207-222, 2002.

[69] S. Ge and Y. Cui, "New Potential Functions for Mobile Robot Path Planning", in *IEEE Transactions on Robotics and Automation*, vol. 16, No. 5, pp. 615-620, Oct. 2000.

[70] J. Borenstein and Y. Koren, "Real-Time Obstacle Avoidance for Fast Mobile Robots", in *IEEE Transactions on systems, Man and Cybernetics*, vol. 19, pp. 1179-1187, Sept/Oct. 1989.

[71] J. H. Chuang and N. Ahuja, "An Analytically Tractable Potential Field Model of Free Space and its Application in Obstacle Avoidance", in *IEEE Transactions on systems, Man and Cybernetics-Part B*, vol. 28, pp. 729-736, Oct. 1998.

[72] H. S. Chae, H. S. Han, "Combination of RFID and Vision for Mobile Robot Localization," *Intelligent Sensors, Sensor Networks and Informatin Processing Conference*, 2005.

[73] K. Yamano, K. Tanaka, M. Hirayama, E. Kondo, Y. Kimuro, M. Matsumoto, "Self-localization of Mobile Robots with RFID System by using Support Vector Machine," *Intelligent Robots and System Proceedings, IEEE Conference*, 2004.

[74] M. Cetin, L. Chen, J. W. Fisher III, A. T. Ihler, R. L. oses, M. J. Wainwright, A. S. Willsky, "Distributed Fusion in Sensor Networks," *IEEE Signal Processing Magazine*, July. 2006.

[75] K. S. Leong, M. L. Ng, P. H. Cole, "Positioning Analysis of Multiple Antennas in a Dense RFID Reader Environment," *Proc. of Application and the Internet Workshops*, Jan. 2006.

[76] D. W. Engels, S. E. Sarma, "The Reader Collision Problem," *Proc. of IEEE International Conference, Systems, Man and Cybernetics Application and the Internet Workshops*, Oct. 2002.

[77] K. S. Leong, M. L. Ng, A. R. Grasso, P. H. Cole, "Synchronization of RFID Readers for Dense RFID Reader Environment," *Proc. of Application and the Internet Workshops*, Jan. 2006.

[78] H. Sidenbladh, "Multi-Target Particle Filtering for the Probability Hypothesis Density," *Proceedings of the International Conference on Information Fusion*, pp. 800-806, 2003.

[79] S. Cho, J. Lee, S Hong and W. Cho, "Passive Sensor Based Multiple Object Tracking and Association Method in Wireless Sensor Network," *International Journal of Distributed Sensor Networks*, In Press, 2008.

[80] K. Finkenzeller, "RFID Handbook, Radio-Frequency Identification: Fundamentals and Applications," Wiley & Sons LTD., 1999.

[81] I. F. Ahyildiz, "A survey on Wireless Mesh Networks," IEEE Radio Communications, Sept. 2005.

[82] C. Volkan, J. H. McClellan, "General Direction-of-Arrival Tracking with Acoustic Nodes," *IEEE Trans. Signal processing*, vol. 53, no. 1, pp.1-12, 2005.