MULTIPLE HUMAN TRACKING AND IDENTIFICATION WITH WIRELESS DISTRIBUTED PYROELECTRIC SENSORS

by

Qi Hao

Department of Electrical and Computer Engineering
Duke University

Date:__________________________

Committee:

________________________________________
Dr. David J. Brady, Supervisor

________________________________________
Dr. Robert D. Guenther

________________________________________
Dr. Qing H. Liu

________________________________________
Dr. Jeffrey Krolik

________________________________________
Dr. Krishnendu Chakrabarty

Dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Electrical and Computer Engineering in the Graduate School of Duke University

2006
ABSTRACT

MULTIPLE HUMAN TRACKING AND IDENTIFICATION WITH WIRELESS DISTRIBUTED PYROELECTRIC SENSORS

by

Qi Hao

Department of Electrical and Computer Engineering
Duke University

Date: ____________________________

Approved:

_______________________________
Dr. David J. Brady, Supervisor

_______________________________
Dr. Robert D. Guenther

_______________________________
Dr. Qing H. Liu

_______________________________
Dr. Jeffrey Krolik

_______________________________
Dr. Krishnendu Chakrabarty

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Electrical and Computer Engineering in the Graduate School of Duke University

2006
Abstract

The advantage of thermal human tracking over the optical counterpart lies in its theoretical and practical illumination invariance. The goal of our research is to develop a prototype wireless distributed pyroelectric sensor system, which can track multiple human objects inside a room, while maintaining their identities under all-illumination circumstances. It involves two sub-problems: multiple thermal source tracking and thermal object identification.

Throughout construction of the prototype sensor system, the following topics have been addressed, investigated, and managed:

(1) Signal processing balance between photonics and electronics. Given the feasibility of visibility modulation using Fresnel lens arrays for the pyroelectric sensor, we have explored different visibility coding schemes and sensor configurations & deployments to achieve higher sensing efficiency and efficacy, better tracking precision, and more representative human thermal features.

(2) Algorithm trade-off between performance and cost, for real-time system implementation. We have investigated and tested various signal processing, event inference, data-object-association, and tracking techniques, for the multiple human tracking purpose, as well as supervised regression and unsupervised expectation-maximization learning schemes, to the human thermal feature clustering and identification end. Computational choices and compromises have been made to neutralize the conflicts between tasks and resources of the sensor
(3) **Distributed computation and communication management amongst host, master, and slave modules.** We have developed a prototype sensor system, consisting of several sensing enabled slaves, one master, and one host. Efforts have been made in assigning all the computing, from event detection/capture and signal digitization to communication synchronization and error rejection to data fusion/association/learning and task synthesis, amongst these three kinds of modules, so as to draw an efficient management of computation and communication resources out of their heterogeneous capabilities and purposes.

The main accomplishments of this thesis include the following four components.

1. A real-time demo of one human object tracking under a Bayesian tracking scheme with wireless distributed pyroelectric sensors.
2. A real-time demo of multiple human object tracking under an EM-Bayesian tracking scheme with wireless distributed pyroelectric sensors, whose unique two-column optic geometry is designed to facilitate the process of data-object-association.
3. A real-time demo of path-dependent human identification based on linear regressions of spectral features of event signals.
4. A real-time demo of path-independent human identification based on Hidden Markov Model’s accommodation of digital event index sequences that are chosen as statistical features of human objects.

This thesis proposes and explores the unexploited frontier and terrain of multiple human object tracking and identification in pyroelectric terms. On account of what we accomplished, it can be heralded that the low cost, low power consumption yet reliable pyroelectric detector will, in the near future, rise to be a mainstream human detection instrument, beside its video and audio counterparts, for extensive
applications from human-machine interfaces to human biometrics.
Acknowledgements

I want to express my gratitude to all those people who have helped me to accomplish this thesis.

The most credit should go to my adviser, Dr. David J. Brady. By sharpening the edges of experimental vigor and theoretical rigor, drawing the line between wildness in imagination and strictness in demonstration, he has brought the most creative essence and the best work out of me. His guidance reinforces the notion that among all the virtues of an engineering scientist, the innovation always ranks the first.

I am indebted to Dr. Robert D. Guenther, for all the composition instructions and technical discussions generously and intelligently offered. His accommodating manner in no short of the critical edges and humorous style with an insightful depth are among the top qualities I have been desiring to possess myself.

I am grateful to my committee members, Dr. Qing H. Liu, Dr. Jeffrey Krolik, and Dr. Krishnendu Chakrabarty, for providing many thoughtful and constructive comments on this thesis.

It is my privilege to have the chance to work with the Duke Integrated Sensing and Processing group (DISP). The working experience shared with Steve Feller, Unni Gopinathan, Mingbo Xu, Mohan Shankar, John Burchett, Paul Vosburgh, and Wendy Lesesne has been pleasant and unforgettable. Their endless optimism and enthusiasm in experimental practices have been both appealing and revealing. The fruitful cooperation with Jian-Shuen Fang, resulting in three journal papers in
total, all but reasserts the claim that the reunion of Jiao Tong University alumni from the two sides of the strait can trump any technical obstacle.

DISP is also a place to cultivate friendships. From both the Chinese students, Zhaochu Xu, Yunhui Zheng, Zhanglei Wang, Lin Wang, and the American students, Evan Cull, Andrew Portnoy, Cristina Fernandez, Scott McCain, I always can find advice and assistance, in either technology or language.

This thesis is dedicated to Zhaochu Xu, my dear friend gifted in mathematics and English.
Contents

Abstract iii

Acknowledgements vi

List of Figures xiii

List of Tables xix

1 Introduction 1

1.1 Background and Motivation ......................................... 1

1.1.1 Human Tracking and Identification ............................. 1

1.1.2 Wireless Distributed Sensor Networks ............................. 4

1.1.3 Compressive Sensing and Geometric Sensors ...................... 7

1.1.4 Signal Inference .................................................. 11

1.2 Goal and Achievements ............................................... 17

1.2.1 Main Goal and Challenges ........................................ 17

1.2.2 Main Issues for System Implementation .......................... 19

1.2.3 Main Achievements ............................................... 22

Real-time Single Human Tracking ....................................... 23

Real-time Multiple Human Tracking ..................................... 23

Real-time Path-dependent Human Identification ....................... 24

Real-time Path-independent Human Identification ..................... 24

viii
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.3</td>
<td>Thesis Outline</td>
<td>25</td>
</tr>
<tr>
<td>2</td>
<td>System Setup and Problem Statement</td>
<td>27</td>
</tr>
<tr>
<td>2.1</td>
<td>Sensor Models and Visibility Modulation</td>
<td>27</td>
</tr>
<tr>
<td>2.1.1</td>
<td>Pyroelectric Sensor Modeling</td>
<td>27</td>
</tr>
<tr>
<td>2.1.2</td>
<td>Fresnel Lens Array and Visibility Modulation</td>
<td>30</td>
</tr>
<tr>
<td>2.2</td>
<td>Problem Statements</td>
<td>40</td>
</tr>
<tr>
<td>2.2.1</td>
<td>Single Human Tracking</td>
<td>40</td>
</tr>
<tr>
<td>2.2.2</td>
<td>Multiple Human Tracking</td>
<td>42</td>
</tr>
<tr>
<td>2.2.3</td>
<td>Human Identification</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>Statistical Models of Features</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>Data Learning</td>
<td>47</td>
</tr>
<tr>
<td></td>
<td>Hypothesis Testing</td>
<td>48</td>
</tr>
<tr>
<td>3</td>
<td>Event Capture and Feature Extraction</td>
<td>49</td>
</tr>
<tr>
<td>3.1</td>
<td>Tracking and Identification Strategies</td>
<td>49</td>
</tr>
<tr>
<td>3.2</td>
<td>Event Detection and digitization</td>
<td>51</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Kalman Filtering</td>
<td>52</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Robust Gaussian Mixture Estimation of Noise</td>
<td>55</td>
</tr>
<tr>
<td>3.2.3</td>
<td>Windowed FFT</td>
<td>58</td>
</tr>
<tr>
<td>3.2.4</td>
<td>Band-Pass Sine Filtering</td>
<td>59</td>
</tr>
<tr>
<td>3.3</td>
<td>Event Registration</td>
<td>62</td>
</tr>
<tr>
<td>3.4</td>
<td>Event Validation and Interpretation</td>
<td>64</td>
</tr>
<tr>
<td>3.5</td>
<td>Feature Extraction</td>
<td>67</td>
</tr>
</tbody>
</table>
4 Single Human Tracking

4.1 Data Fusion and Tracking Synthesis

4.1.1 Motion Inference

4.1.2 Trajectory Smoothing by Bayesian Tracking

4.1.3 Kalman Tracking

4.1.4 Hidden Markov Model Tracking

4.1.5 Gaussian Particle Filter Tracking

4.2 Results and Discussion

5 Multiple Human Tracking

5.1 Introduction

5.2 EM-Bayesian Tracking

5.3 EM Object Localization

5.3.1 E Step: Event Decomposition

5.3.2 M Step: Measurement to Object Association

5.4 Data Fusion and Tracking Synthesis

5.4.1 Template Operation

5.4.2 Bayesian Filtering and Prediction

5.5 Results and Discussion

6 Walker Recognition
xii
List of Figures

1.1 The general diagram of thermal motion tracking. .......................... 2
1.2 The general diagram of a biometric system. ................................. 3
1.3 Sensor node architecture .......................................................... 5
1.4 An illustration of motion detection by the pyroelectric sensor [1] .... 6
1.5 An illustration of the concept of geometric sensor. The $i^{th}$ bit of ‘1’ or ‘0’ of the signature of a segmentation in object space represents that the region can or cannot be seen by the $i^{th}$ sensor. ..................... 9
1.6 The levels of data processed in tracking and identification systems .. 12
1.7 The setup of the distributed wireless pyroelectric sensor system ... 18
1.8 Main issues for distributed pyroelectric sensor system building ...... 20
1.9 The architecture of the thesis ................................................... 25

2.1 Polar plot of visibility of dual element pyroelectric detector ...... 29
2.2 Bode plot of one pyroelectric sensor with an amplifier ............... 29
2.3 Step response of the pyroelectric sensor system ......................... 30
2.4 Visibility pattern a pyroelectric sensor with an 11-element Fresnel lens array as the collection optics. Details of the beam pattern are shown on the second beam from the left. (a) Top view. (b) Side view. 31
2.5 Response signals of a dual element pyroelectric detector to a passing thermal source, (a)without multiplex visibilities modulation, (b)with multiplex visibilities modulated by Fresnel lens ....................... 32
2.6 The radial sensor module used for the characterization of Fresnel lens Field Of View ................................................................. 33
2.7 Visibilities of one sensor module .............................................. 33
4.9 The computation time per echo in MTALAB with respect to different tracking algorithms and signal processing filters. 83

4.10 The response signals at different channels of four sensor nodes. 85

4.11 The measured angular displacements from the four nodes. 86

4.12 The tracking results with real sensory data with different event detection schemes. circle: Kalman IIR; square: windowed FFT; diamond: sine FIR. 86

5.1 The dynamics of multiple targets. 88

5.2 The proposed EM Bayesian tracking scheme. 90

5.3 Grid approximation to linearize the measurements of angular displacements with respect to four sensor nodes. 92

5.4 Typical validation gates and validation matrices. 94

5.5 Data fusion architecture. 96

5.6 The experiment setup. 99

5.7 A snapshot of the one object tracking. 99

5.8 The 8-bit event sequences of four nodes for one object tracking. 100

5.9 The tracked trajectory of the human object, represented by circles, along the prescribed route (solid line). 100

5.10 The tracking errors and their histogram. 101

5.11 A snapshot of the two object tracking. 102

5.12 The 8-bit event sequences of four nodes for two object tracking. 102

5.13 The two tracked trajectories of two human objects, represented by circles and crosses, along the prescribed route (solid line). 103

5.14 The tracking errors of two trackers and their histogram. 103

5.15 The 8-bit event sequences of four nodes for tracking two objects having crossing paths. 104
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.4</td>
<td>The sensor-obscurant pattern to form a new signature.</td>
<td>145</td>
</tr>
<tr>
<td>A.5</td>
<td>More efficient use of the sensor-obscurant pattern to obtain a new</td>
<td>146</td>
</tr>
<tr>
<td></td>
<td>signature.</td>
<td></td>
</tr>
<tr>
<td>A.6</td>
<td>The obscurant pattern to form a new signature in 2-D model.</td>
<td>148</td>
</tr>
<tr>
<td>A.7</td>
<td>The patterns to form new signatures with multi-layer obscurants.</td>
<td>149</td>
</tr>
<tr>
<td>A.8</td>
<td>The patterns to form new signatures with subsets of sensors and</td>
<td>151</td>
</tr>
<tr>
<td></td>
<td>obscurants.</td>
<td></td>
</tr>
<tr>
<td>A.9</td>
<td>The distribution of signatures for 10 sensor and 10 obscurants.</td>
<td>151</td>
</tr>
<tr>
<td>A.10</td>
<td>The patterns to form new signatures with multi-layer obscurants.</td>
<td>153</td>
</tr>
</tbody>
</table>
List of Tables

1.1 Comparison of pyroelectric and speech signal processing . . . . . . . 16

2.1 Visibility coding scheme for radial sensor modules. . . . . . . . . . 34
2.2 Visibility coding scheme for two-column radial sensor modules. . . 36
2.3 Visibility coding scheme for the digital feature extraction sensor mod-
ule. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 39

4.1 Comparison of Bayesian tracking algorithms . . . . . . . . . . . . 82
4.2 Comparison of embedded signal processing algorithms . . . . . . . 83

6.1 Open-set path-dependent identification results using LRM . . . . . 121
6.2 Close-set path-dependent identification results using HMMs . . . . 122
6.3 Close-set path-independent identification results using HMMs . . . 124

7.1 Comparison of DISP electronic platform with Mote . . . . . . . . 126
7.2 Communication interface for slave, master, and host . . . . . . . . 128
Chapter 1

Introduction

1.1 Background and Motivation

1.1.1 Human Tracking and Identification

Human tracking includes capturing displacements and body movements, such as postures and gestures, of human objects, and has been proposed for extensive applications as human-machine interfaces, ranging from surveillance, computer vision, and robotics, to virtual reality, intelligent space, smart video conference, etc. [3], [4], [5], [6], [7], [8], [9], [10], [11]. Compared with its non-vision based counterpart, the nonintrusive sensory vision based approach, either optical or thermal, can interpret human motions in a more natural way without interfering with the dynamics of human motions. The advantage of thermal motion tracking over the optical counterpart lies in its theoretical and practical illumination invariance, a notion that its sensing and thus tracking capabilities are independent of the intensities and spectra, in a considerable range, of external lighting sources. It is because that the mid-to-long wave infrared radiation from the human body, from 8 $\mu m$ to 14 $\mu m$ with a peak at 9.55 $\mu m$, is emitted rather than reflected [12] [13]. In most cases, a vision based motion tracking problem consists of several sub-problems, as shown in Fig. 1.1, including feature extraction (detection), feature-to-object association
(localization), and motion estimation (tracking) [14], [15].

![Diagram of thermal motion tracking]

**Figure 1.1:** The general diagram of thermal motion tracking.

Meanwhile, many intelligent machine and secure systems demand collectable, stable, and reliable biometrics, which could identify an individual based on his/her physiological or organic characteristics, such as fingerprints, iris patterns, voice spectra, and face features, and learned or behavioral ones, like signatures, keystroke dynamics, and gaits, in the applications for human object verification and identification. [16]. Any functional biometric feature should be characterized by its universality among all human objects under examination, distinctiveness between any two human objects, characteristic invariance over a period of time, and feasibility for quantitative measures of the characteristic. [17] The metrics to evaluate a practical biometric system include the recognition accuracy, response speed, resource requirements, acceptability to users, and robustness to clutter and fraud [18]. A biometric system indeed is an intrinsic pattern recognition system, as illustrated in Fig. 1.2, and thus comprises three parts: feature representation, feature training (clustering), and feature testing.

From the thermal perspective, each person acts as a distributed infrared source, whose distribution is determined by the shapes of the head, torso, and limbs and their thermal emission. Combined with the idiosyncrasies in how an individual carries himself/herself and the habits of how he/she moves, the heat will impact the surrounding sensor field in a unique way, statistically, and generate a signature
Figure 1.2: The general diagram of a biometric system.

of its own in the signal space. As a result, different types of information of the thermal sources can be extracted from the sensory data, particularly static features and dynamics features. The static features depict physiological aspects of a human thermal source; and the dynamic features represent the dynamic change of the static features, and are assumed to correlate with the movement speeds, rhythms of arm swing and leg crossing, and other parameters associated with the kinetic idiosyncrasies of people. Meanwhile, the statistics of static/dynamic features over a period can characterize walkers who follow random paths and can be used as statistical features.

From the viewpoint of trackers, human tracking and human identification are a pair of coupled problems, particularly in the case of multiple human tracking, which relies on target identification, from time to time, to reduce the mutual interference among those multiple sources when their paths cross [19]. Even for the single target tracking case, a refined estimation of radiation source distribution, a type of static
target identification, can improve the quality of whole tracking performance. On the other hand, in order to estimate the distributions of thermal sources and the motion idiosyncrasies thereof from the sensory data, their locations and speeds should be well known at first through a tracking scheme.

1.1.2 Wireless Distributed Sensor Networks

Apart from the expensive hardware cost (in hundreds folds of that of an optical video camera), however, even a conventional monocular thermal tracking system imposes, as indeed does its optical counterpart, challenges of massive amounts of real-time computations and huge volumes of data-loads, in thermal flow computing and estimations in configuration space of source distributions and motion trajectories from input high spatial resolution image sequences; let alone a multi-perspective one, which at times is required to localize features in 3D space and of multiple targets [20], [21]. Consequently, even detection of moving objects in video stream is known to be a significant and difficult research problem, as conventional approaches to moving object detection include temporal differencing, background subtraction, and optical/thermal flow computing [7], [15], [22]. In many cases, due to the availability of prior knowledge on target motion dynamics and the scarce amount of information of interest about targets, the intensive and expensive imaging detector array appears rather inefficient and unnecessary. For instance, for an image consisting of 100 × 100 pixels in 8-bit grey level, the information of interest, such as position and velocity, only needs several bits to represent [23].

The unprecedented advances in micro-processor, radio frequency transceiver, micro-electro-mechanical system (MEMS), and novel sensor technologies in the last decade allow the development of promising alternatives for coping with those chal-
challenges: distributed sensor networks (DSNs). In a typical DSN, many small, low-cost, spatially dispersed sensor nodes, as shown in Fig.1.3, endowed with computation and communication capabilities, collaborate with each other and transfer only the information of interest to achieve complicated tasks. For most DSN applications, passive sensors are preferred to active ones because of their low costs, low power consumption, and high stealthiness. Specifically, cameras and microphones commonly used unobtrusive sensors to capture the audio-visual signals associated with various static and dynamic features of events generated by human objects. Despite their success in many applications of human detection, localization, and identification, both cameras and microphones have their own inherent limits. The former is very sensitive to the intensity and spectrum of illumination; the latter cannot perform if the human objects keep silent. Besides, as numerous applications of DSNs are reported in vehicle tracking [19], [24], [25], [26], and robot positioning [27], few reports can be found on human motion tracking [3] and human identification.

In our study of human tracking and identification under the all-illumination circumstances, passive infrared (PIR) $LiTaO_3$ pyroelectric sensors are chosen, because of

![Figure 1.3: Sensor node architecture.](image)
Figure 1.4: An illustration of motion detection by the pyroelectric sensor [1]

1) their dramatically lower costs, only 2 dollars per piece, and commercial availability;

2) their sensitivity to the radiation emitted by the human body (8~14 µm) [13];

3) their sensitivity to angular velocities of a thermal target ranging from 0.1 rad/s to 3 rad/s, in a distance less than 15 m;

4) their response signals proportional to the change of temperature on the crystal other than the temperature itself [28], such that only the motions of targets can be detected, as shown in Fig. 1.4, a main property of compressive sensing;

5) their large pyroelectric coefficient and chemical stability [29];

6) the feasibility of manipulation of their visibilities, as contrasted with those acoustic sensors, one key concept of geometric sensors [30], [31].
To implement a wireless DSN with the limited computation resources, narrow communication bandwidth, and restricted power supply, major issues including collaborative signal processing [19], [32], networking, embedded computing, sensor deployment, and power management need to be addressed [33]. Robust collaborative signal processing techniques can map the signal states, measured by the distributed sensors, through low-level local computation at each node, into configuration state sequences, which after decision fusion constitute the final motion trajectories and identities of targets. A suitable networking protocol guarantees reliable information routing and data dissemination. Embedded computing is the main activity of a wireless sensor network and should be succinct in coding and effective for the tasks. In a DSN, the distributed sensors as well have to be located intelligently to meet specifications on system performance such as sensing coverage and tracking resolution.

1.1.3 Compressive Sensing and Geometric Sensors

The new paradigms of compressive sensing and geometric sensors can play promising roles in the trade off between the high information gain in expectation out of distributed sensor systems and the bottleneck in reality of their narrow data throughput and limited computation power. Compressive sensing technology bears resemblance to the technique of compressive sampling, which stems from the convergence between generalized sampling theory and signal compression technology and can implement sub-Nyquist sampling and data compression in the physical layer, enabled by multiplex sensing, multichannel and multi-scale signal analysis, and signal inference, such as group testing and various Bayesian inference based methods [34]. Compressive sensing, by comparison, takes into account the dynam-
ics of the pyroelectric and associated transconductance circuits, as well as the sensor visibilities or measurement coding schemes. It demands a selection of proper signal processing methods to render sparse representation of signal features based upon prior knowledge of sensor models, for a further signal inference with a selection of proper measurement coding schemes that function as spatial filters.

The multiplex sensing technique, in essence, enables each sensor to take measurements that depend jointly on multiple source points in the object space under examination, such that the linear combinations of object data can increase the mean power per measurement and hence the signal-to-noise ratio in the presence of additive noise. The multiplex advantage is substantial in the infrared, where thermal noise dominates. It has widely been used in the applications of high radiation imaging, such as X, γ-ray imaging, to improve the signal-to-noise ratio during the photon collection. [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45], [46], [47].

The multiplex sensing technique also allows the coherence of distributed source motions to be synthesized and recorded in one signal channel. By using a multiplex sensing technique, the source motions can be captured and converted into spectral content of temporal sensory signals. An 11-element Fresnel lens array produces 11 visibilities for one pyroelectric sensor. The response data spectra as a human walks through the 11 areas can be used to discriminate between two individuals. The spectral signatures are believed to be caused by the difference in the motions of arms and legs, despite human objects’ similar walking speeds [48].

The concept of geometric sensors assumes that the radiation field that propagates from the object space to the measurement space is characterized by a simple ray propagation model rather than a wave propagation one, as shown in Fig. 1.5. By manipulating the global and local visibilities of the sensors, redundant radiation information can be filtered out [49], [50], [51]. For example, a Fresnel lens can shape
Figure 1.5: An illustration of the concept of geometric sensor. The $i^{th}$ bit of ‘1’ or ‘0’ of the signature of a segmentation in object space represents that the region can or cannot be seen by the $i^{th}$ sensor.

the visibilities of pyroelectric sensors to detection regions with different visibility patterns; when objects enter some of these regions, different sets of sensors can fire simultaneously, and from those signal patterns the locations of objects can be obtained. For each sensor, however, a trade-off between global and local visibility modulation has to be made. With more global visibility one sensor can detect a larger area and its sensing efficiency will hence be higher, yet at the expense of increased sensing ambiguities. In our study, local visibilities of sensors are modulated with deliberation for two purposes. On the one hand, visibility modulation can form some unique local detection regions, which can detect the angular displacements of thermal sources and function as measurement validation gates to facilitate the data-object-association process, to help reduce computational cost and sensing ambiguities. On the other hand, when a human walks through the visibility
modulated object space, the statistical pattern of the binary sequence generated by the sensor array can be associated with the identity of that person [52]. Some mathematical discussions about geometric sensors in segmentation of object space can be found in Appendix A.

The underlying mechanism of exploitation of multiplex sensing advantages and geometric sensor leverages comprises the study of reference structure tomography [53], [54]. It suggests that multi-dimensional features of a radiation source could be captured at an arbitrary level, once there exists such a set of base functions that structurally pose and numerically condition the reconstruction procedure, by state-of-the-art fine tuning the multi-dimensional visibilities of distributed sensors. Through its scan-free multi-dimensional imaging, the feature abstraction, shape parameterization, and even characteristic classification of radiation sources under examination can be achieved in a data-efficient and computation-efficient way. A general visibility design procedure has yet to be proposed, but many visibility coding schemes have already been applied in different coded-aperture imaging systems, from Hadamard codes to pseudo-random codes [35], [36].

The complexity and advantages of compressive sensor systems under our study lie in the accommodation of nonlinear and finite supportive multiplex visibilities, associated with some coding schemes, which inevitably are characterized as one of those discrete-event systems with partial observation, in contrast to the more commonly used counterparts—those sensor systems featured with locally homogeneous visibilities, albeit oftentimes nonlinear too, which yield continuous measurements with respect to varying configuration states of objects. Here, we use the term of “linear”, or “nonlinear”, for a sensor system to describe whether there exist a linear/nonlinear transformation between the object configuration space and the sensor measurement space through the mapping of that system. Two kinds of systems dif-
fer in their ways of observing target dynamics. The former only capture target dynamics (events) inside specific multiple detection regions, formed through visibility modulation by so-called reference structure, such as Fresnel lens arrays and coded masks in practice; whereas the latter detect target dynamics inside more uniformly distributed, inherent sensing zones alike, yet without such a modulation, whose sensitivity is only attenuated with the increasing of the distance between the target and each sensor, in a negative exponential or an inverse square law.

The implementation of compressive sensing and the geometric sensor originates from the efforts to improve data collection efficiency—one sensor can sense different type of events, to capture main components of both motions and radiation features of human objects at the stage of sensory data acquisition, to increase the robustness of the mapping between measurement states of sensors and configuration states of human objects by properly exploiting the sensor redundancy, and to achieve higher data processing efficacy and efficiency. As such, the pyroelectric signals are converted into quantized vectors whenever events are detected. Those digitized event indexes are then applied with logical analysis and reasoning, such as group testing and Bayesian inference, for the purposes of data fusion/association/learning and human object tracking/identification.

1.1.4 Signal Inference

From the computational point of view, for a complicated tracking and identification sensor system, the data are processed on different levels, illustrated in Fig. 1.6.

1) Signal: This is the lowest level of abstraction where sensory data from distributed and clustered slaves are captured, digitalized, and processed locally before broadcasting to a master node;
Figure 1.6: The levels of data processed in tracking and identification systems.
2) Event: This is a spatial-temporal pattern represented by an 8-bit index. It represents the on/off status of eight sensors of one slave node. It contains information about the angular displacement measurement with respect to this slave node.

3) Feature: This is a pattern made up of 4-bit event indexes, or spectral data of response signals of an event. It is associated with registered objects and used to distinguish them from each other.

4) Object: This is the source of a pattern defined in the spatial domain. In our study, an object is a walker whose identity is associated with features generated by a hidden Markov model (HMM), or a linear regression model (LRM).

5) Template: This is a pattern of multiple objects, such as the number of people in the object space and who they are.

During the operation of a sensor system, signals are processed by various techniques. Events are represented by multiple dimensional binary index sequences. Features are generated into two types: digital and analog. The former are represented by event indexes sequences and the latter by spectral segments of sampled event data. The objects need to be associated with the measured events and extracted features. Then their states can be updated recursively with new measurements. For different scenarios, different templates are chosen to minimize the computation costs. All these activities inevitably involve reasoning and inferences.

Among all the machine intelligence approaches Bayesian inference has emerged as a powerful tool in signal inference, machine learning, and computational decision-making due to its theoretical completeness, weighting on prior knowledge, intrinsic suitability for hierarchical reasoning networks, and flexibility for introducing various
approximations. Most applications of Bayesian inference can fall into three categories: Bayesian decision, about how to choose sensor location and configuration to maximize the information utility of sensor systems based on the predicted posterior; Bayesian tracking, about how to estimate instantaneous states of dynamic systems by Kalman filters, hidden Markov model filters, and numerous particle filters, given dynamics prior and observation likelihood; Bayesian learning, based on optimization of the explicit model parameters with prior distribution, for unknown systems or data-sets, in the light of marginal likelihood of posterior, specifically for clustering hidden Markov models and Gaussian mixture models under the expectation maximization (EM) scheme, or its extended variational versions [55], [56].

Despite having been generalized in one theoretical framework and employing rather similar strategies, from mixture Gaussian approximation to Monte Carlo simulation, to break the non-linear/non-Gaussian barriers, which, inextricably, inflict numerical difficulties of high-dimensional integration, these three classes of applications, in practice, are demanded for different types of performance and thereby their algorithmic implementations exhibit quite a lot of diversities. For example, for Bayesian tracking problems, more weights are put upon real-time computation and newly collected data, and therefore most particle filters take advantages of an independence sampling technique, namely importance sampling, for its algorithmic simplicity, as opposed to those dependence sampling schemes, such as Gibbs and Metropolis samplers, or a hybrid one, used in Markov Chain Monte Carlo methods for off-line Bayesian learning problems, which, on the other hand, emphasize more on fast convergence and tractable optimality of computation results, all in a higher standard than those for Bayesian tracking and decision. As a result, sampling/re-sampling (sample improvement) schemes play a pivotal role in the Monte Carlo based estimation approaches. A spectrum of techniques, or improved ones, have
been proposed and practiced, besides Gibbs, Metropolis and straightforward importance sampling, including recursively smoothing, re-sample-move, and regularization, [55], [56], [57], [58].

In general, the learning (clustering) methods can be categorized into three groups: regression based, similarity based, and kernel mixture model based. The typical regression based methods include supporting vector machine [19] and principal component analysis (PCA) enhanced linear regression [48]. Intrinsically, during the training phase, the regression based methods try to find a set of linear, or kernel based, boundaries which separates the different data sets [59]. Similarity based approaches include K-mean, PCA enhanced K-mean, and some kernel based extensions. By defining distances among data vectors, the data sets are grouped according to their mutual distances [59]. With the development of Bayesian machine learning theory, the kernel mixture model based clustering methods, such as Gaussian mixture model and hidden Markov model mixture, have attracted increasing research attention and have been extended to variational Bayesian modes, under general expectation-maximization schemes and with closed-form posterior updating [55], [56].

It is also worthy to note the parallelism between pyroelectric spatial-temporal signal processing and speech signal processing, from the perspective of signal inference. Pyroelectric spatial-temporal signals are processed for human tracking and walker recognition; speech signals for word recognition and speaker recognition [60]. Both human motions and spoken words can be modeled as hidden Markov chains for tracking and recognition [61]. Multiple human tracking bears a resemblance to tracking words of multiple speakers, as both utilize the concept of coupled multiple hidden Markov chains [62]. For speaker recognition, there are text-dependent and text-independent solutions [63]; for walker recognition, path-dependent and
<table>
<thead>
<tr>
<th></th>
<th>Pyroelectric</th>
<th>Speech</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Spatial-Temporal</strong></td>
<td><strong>Signal Processing</strong></td>
<td><strong>Signal Processing</strong></td>
</tr>
<tr>
<td>Source</td>
<td>Human objects</td>
<td>Human objects</td>
</tr>
<tr>
<td>Events</td>
<td>Motions</td>
<td>Words</td>
</tr>
<tr>
<td>Event model</td>
<td>Hidden Markov chains</td>
<td>Hidden Markov chains</td>
</tr>
<tr>
<td>Static features</td>
<td>Thermal distribution</td>
<td>Vocal organ size</td>
</tr>
<tr>
<td>Dynamic features</td>
<td>Walking habits</td>
<td>Speaking habits</td>
</tr>
<tr>
<td>Feature</td>
<td>Event index/spectrum</td>
<td>LPC/Cepstrum</td>
</tr>
<tr>
<td>representation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feature Models</td>
<td>HMM/LRM</td>
<td>HMM/GMM/LRM/NLRM</td>
</tr>
<tr>
<td>Recognition approach</td>
<td>Path dependent /independent</td>
<td>Text dependent /independent</td>
</tr>
<tr>
<td>Publications</td>
<td>&lt;10</td>
<td>&gt;1000</td>
</tr>
<tr>
<td>Funding</td>
<td>&lt;1 million</td>
<td>Hundreds of millions</td>
</tr>
</tbody>
</table>
path-independent. It is rather reasonable and attractive to transfer and transform numerous techniques developed for speech signal processing, from signal feature representation to feature statistical model building, for the solutions of pyroelectric spatial-temporal signal processing [64], [65], [66], [67], [68], [69], [70], [71], [72], [73], [74], [75], [76], [77], [78], [79], [80]. The whole comparison is listed in Table 1.1.

1.2 Goal and Achievements

1.2.1 Main Goal and Challenges

The ultimate goal of our work is to develop a wireless distributed pyroelectric sensor network, which can track multiple human motion inside rooms, while maintaining their identities, as illustrated in Fig. 1.7. We are more concerned with the detection of the global motion (i.e. positions and speeds) of human objects. Meanwhile, those signal patterns associated with the locomotions, such as the swing of arms and crossing of legs, are utilized as human dynamic features to the human identification end. The information obtained by pyroelectric temporal-spatial signal processing techniques can be used to extract the following information from the walkers:

1) Walker detection: *Is there someone moving?*

2) Walker tracking: *What are their locations?*

3) Walker recognition: *Who are the walkers?*

The challenges for integrated human tracking and identification systems with pyroelectric sensors include

1) high variability and volatility of human motions and their thermal biometrics;
Figure 1.7: The setup of the distributed wireless pyroelectric sensor system.

2) difficulties in capturing and representing main components of motions/thermal features;

3) longer latencies and reduced sensitivity of pyroelectric sensors with respect to distant objects;

4) over-simplifications in geometric optics modeling and errors in sensor system alignment;

5) limits and constraints upon sensor number, sampling resolution, computation capabilities, and communication bandwidth;

6) development of simple and practical data fusion, motion inference, and feature generation techniques, subject to data transmission bandwidth and embedded computation capabilities;
7) development of a proper data-to-object-association technique for multiple human tracking;

8) selection of effective feature models that can accommodate signal pattern distinctions among human objects.

1.2.2 Main Issues for System Implementation

In an initial effort toward development of more complicated, advanced wireless pyroelectric sensor networks for human tracking and identification, we developed first a prototype distributed pyroelectric sensor system, with five slave modules, four for tracking and one for identification, one master module, and one host PC. The slave module for tracking contains eight pyroelectric sensors with modulated visibilities to measure the angular displacements of thermal sources. The slave module for identification contains four sensors, whose visibilities are modulated with different codes to identify the enrolled walkers. Each slave module is able to collect the sensor response signals, convert them into digital event indexes, by filtering, threshold testing, and smoothing, and send them to the master module via wireless channels. The master module collects event indexes from different slave modules, and synthesizes those from tracking slaves into one composite event message, while synchronizing the communication and rejecting the event errors. The master module sends the processed messages to the host via a UART serial cable. The host converts the event messages from tracking nodes into local bearing measurements or into walker features, determines the number of objects at presence, associates measurements with different objects, estimates the memberships of the walker features, and then fuses all the data, through Bayesian tracking scheme and likelihood comparison, to estimate the motion trajectories and the identities of
Figure 1.8: Main issues for distributed pyroelectric sensor system building.

As shown in Fig. 1.8, the main design issues for building a distributed pyroelectric sensor system include sensor visibility modulation, embedded computation & communication, and data learning/association/fusion & tracking/identification synthesis. Through building prototype sensor systems, the following topics are addressed, investigated, and managed:

1) **Signal processing balance between photonics and electronics.** Given the feasibility of visibility modulation via Fresnel lens arrays for the pyroelectric sensor, we have explored different visibility coding schemes and sensor configurations & deployments to achieve higher sensing efficiency and efficacy, better tracking precision, and more representative human thermal features. For one object tracking, we employ Fresnel lens to modulate visibilities of eight sensors along a circle such that each sensor module turns out to be a thermal angular displacement sensor in 360°. To improve the resolution, the visibilities overlap
of neighboring sensors are utilized. For multiple human tracking, a two-column lens structure for sensor visibility modulation is designed such that the presence of more-than-one targets can be detected at the stage of data acquisition. The combination of multiple two-column sensor nodes also can facilitate the process of data-to-object-association. For walker recognition, an one-column lens structure with pseudo-random modulation codes is employed to generate digital features which can discriminate among walkers.

2) **Computation distribution among host, master, and slave.** How to assign all the computing including filtering, threshold testing, geometry interpretation/feature generation, error rejection, and tracking/identification synthesis among all the computation components is indeed the central issue to improve system performance to the maximum extent. A basic requirement upon a wireless sensor system is that its operation should be event based such that each time only a few bytes that describe a sensory event are transmitted via the wireless channels with limited bandwidth. As a result, the system should be run in a way so that slaves are able to generate event indexes, the master is able to validate multiple event indexes and remove the inconsistence and errors among them, and the host able to reason and infer from the collection of those event indexes.

3) **Algorithm trade-off between performance and cost.** How to simplify algorithms without sacrificing too much of their performance is critical to meeting the real-time implementation demands. For example, particle filters are capable of synthesis of the nonlinear measurements of dynamic states but suffer from the drawback of intensive computation [57], [58]. Windowed FFT is powerful, robust in signal processing for event detection and feature extraction, but con-
sumes a large amount of computation resources in low-end micro-controllers. For feature training, supervised learning approaches converges quickly but requires a strict training process; the unsupervised learning approaches advantage in demanding a less rigid training process but are likely to be trapped into local optima. Therefore, computational choices and compromises have to be made from time to time to neutralize the conflicts between tasks and methods subject to the resources.

4) **Communication protocol regulation for efficiency and accountability.**
   As a considerable part of battery power is used for wireless communication, a suitable communication protocol not only conserves the limited energy and hence prolongs the system lifetime, but also reduces latency and false alarm rates. In the study of this thesis, our prototype sensor system is more of a distributed sensor system than a distributed sensor network. The communication mode is centralized and the requirement on the number of wireless channels is low. Hence fewer problems on wireless communication than on data fusion are inflicted for real-time implementation.

5) **Sensor deployment in light of sensing coverage and tracking resolution.** Given a number of sensors, different deployments will yield different sensing coverages and tracking resolutions.

### 1.2.3 Main Achievements

The main achievements of this thesis include building a prototype wireless distributed pyroelectric sensor system, and its applications for real-time multiple human tracking, real-time path dependent human verification/identification and real-time path independent human identification.
Real-time Single Human Tracking

In this work, we present a wireless pyroelectric sensor system, composed of distributed radial sensing, sampling, digitalization modules (slaves), a dedicated synchronization, error rejection module (master), and a data fusion, task synthesis module (host), to serve human tracking ends. The signal processing balance between optics and electronics, the computation workload distribution among slave, master, and host, and the algorithmic trade-off between performance and cost of computation are proposed, discussed, and managed, as an initial effort toward the development of more complicated wireless pyroelectric sensor networks. In the prototype system, each pyroelectric sensor module is enabled to detect the angular displacement of a thermal target by visibility modulation using Fresnel lens. The clustered sensor signals are filtered, digitized, and smoothed by sine filtering, threshold testing, and hidden Markov model (HMM) filtering, in either slave or master modules, before finally being synthesized on the host processor to track one human target by using Bayesian tracking techniques.

Real-time Multiple Human Tracking

In this work, we present a framework of a wireless pyroelectric sensor system, whose sensing visibilities are modulated by Fresnel lens arrays in a specific way to facilitate the process of data-object-association for tracking multiple humans in all-illumination environments. The concept of geometric sensors is discussed and utilized for local/global visibility modulation for the node-centric sensor system. An EM-Bayesian tracking scheme, consisting of detection, localization, and filtering, is proposed and implemented among slave, master, and host modules of the prototype sensor system. The experimental results show that by using the prototype sensor
system we can track two humans inside a $9 \times 9 \ m$ room in both the follow-up and the cross-over scenarios.

**Real-time Path-dependent Human Identification**

In this work, we present a low power consumption and low cost human identification system using a pyroelectric infrared (PIR) sensor whose visibility is modulated by a Fresnel lens array. The whole identification consists of two parts: training and testing. For the data training, we employed the principle components regression (PCR) method to cluster data with respect to different registered objects at different speed levels. The feature data of an object walking along the same path in training yet at random speeds are then tested, by multiplying the trained regression coefficients against the pre-trained clusters to decide whether the target is registered or not, and which member it is in the group if it is registered.

**Real-time Path-independent Human Identification**

In this work, we present a human recognition system by using a pyroelectric sensor array, whose visibilities are modulated by coded-aperture Fresnel lens arrays. Binary event index sequences generated by human objects walking across regions with different sensor visibility patterns are selected as dynamic features characterizing individuals, statistically represented in the form of hidden Markov models (HMMs) by a training process. The likelihood of a new event sequence that is generated when a walker moves around is estimated to determine its membership with respect to enrolled objects by testing it against those pre-trained HMMs.
In this document, we present a framework of human tracking and identification by using the concept of the geometric sensor and its embodiment as a distributed, clustered pyroelectric sensor system, with modulated visibilities, to track multiple human targets while maintaining their identities in an all-illumination environment. Fig. 1.9 illustrates the intrinsic architecture of the whole thesis. The whole system is sensor node centric and event based. For tracking multiple humans, Bayesian tracking schemes and data-to-object-association techniques are utilized; for identifying walkers, both supervised and unsupervised learning approaches are employed.
The remainder of the thesis is organized as follows. Chapter II presents a mathematical description of the distributed geometric pyroelectric sensors and a general statement for tracking and identification problems. Chapter III defines the concept of event and feature from the pyroelectric sensor signal perspective. Chapter IV presents a single human tracking system based on radial pyroelectric sensor modules. Chapter V presents a multiple target tracking system based on two-column pyroelectric sensor modules. Chapter VI presents real-time walker recognition systems by using randomly coded one-column and unmodulated full-visibility pyroelectric sensor modules respectively. Chapter VII describes the setup and implementation of the computation and communication platform. Chapter VIII summarizes the whole document and discusses the strength and weakness of the proposed sensor system and outlines its potentials and the future work.
Chapter 2

System Setup and Problem Statement

This chapter presents a pyroelectric sensor system model, including the impulse response, visibility function, and visibility modulation. The problem statement of tracking, in terms of target motion dynamics and sensor observation model, and of identification, in terms of data learning and hypothesis testing are also given. A series of designs and makes of sensor modules serving different purposes for the node-centric sensor system are described.

2.1 Sensor Models and Visibility Modulation

2.1.1 Pyroelectric Sensor Modeling

In general, a distributed sensor system contains a signal space, a reference structure for sensor visibility modulation, and an object space containing all the possible configuration states of objects. In our study, those components are represented by pyroelectric sensing circuits, Fresnel lens arrays, and human thermal sources walking inside a room.

Under the linearity assumption, the response signal of \( m \) sensors, \( s(t) \in \mathbb{R}^m \), is
given by
\[ s(t) = h(t) * \int_{\Omega} v(\mathbf{r}) \psi(\mathbf{r}, t) d\mathbf{r} \]  \hfill (2.1)

where “∗” denotes convolution, \( h(t) \) is the impulse response of one sensor, \( \Omega \) is the object space, \( v(\mathbf{r}) \in [0, 1]^m \) is the positive visibility function between \( m \) sensors and the object space, modulated by a Fresnel lens array or a coded-aperture mask, \( \psi(\mathbf{r}, t) \) is the radiation from the target.

From Eq. (2.1), it can be seen that the source distribution \( \psi(\mathbf{r}) \) is indeed a hidden variable. This is to suggest, a comprehensive tracking strategy, \( i.e. \) a recursive estimation of \( \psi(\mathbf{r}, t) \), should be developed in an expectation maximization (EM) scheme [81], [82],

- E step: to estimate \( \psi(\mathbf{r}) \), after updating \( r(t) \);
- M step: to estimate \( r(t) \), after updating \( \psi(\mathbf{r}) \).

Under the simplified geometric assumptions on the radiation field, the visibility \( v(\mathbf{r}_1, \mathbf{r}_2) \) describes the contribution by the field at point \( \mathbf{r}_2 \) to the field at point \( \mathbf{r}_1 \). The visibility of a dual element pyroelectric detector is shown in Fig. 2.1, where the distance between the point source under testing and the sensor is normalized to show its generic visibility characteristic. Such a dual lobe visibility pattern is formed because the two pyroelectric elements are connected in series opposition. The signals from each the elements as the thermal source crosses the common area of overlap of the fields of view (FOVs) cancel one another.

Pyroelectric detector response signals are proportional to the change in temperature on the crystal rather than the temperature itself. Therefore, the transfer function of a pyroelectric detection system is a high-pass one. Taking into account the response time of the transconductance amplifier, the whole system can be de-
Figure 2.1: Polar plot of visibility of dual element pyroelectric detector
scribed as a band-pass one. A 2\textsuperscript{nd} order transfer function of the system is [28]

$$H(s) = \frac{U(s)}{\Phi(s)} = k_g \left( \frac{s}{s + 1/\tau_l} - \frac{s}{s + 1/(\tau_e + \tau_l)} \right),$$ (2.2)

where $U(s)$ is the amplified voltage signal; $\Phi(s)$ is the thermal flux; $\tau_e$ is the response
time of electronic circuits; $\tau_l$ is the thermal relaxation time.

Figure 2.2: Bode plot of one pyroelectric sensor with an amplifier.

Fig. 2.2 shows the measured transfer function of the pyroelectric detection
system in use. A measured step response of the system is shown as the dashed line
in Fig. 2.3. The solid line is the step response of the identified system model with
Figure 2.3: Step response of the pyroelectric sensor system.

We can see that our detector system has a bandwidth roughly between 0.7 Hz and 2 Hz, matching a human target moving indoor, with angular velocities between 1.1 rad/s and 3.1 rad/s, if we assume a $\pi/2$ FOV for the sensor.

2.1.2 Fresnel Lens Array and Visibility Modulation

The Fresnel lens we employ is made of a light-weight, low-cost plastic material with a good transmission characteristics in the 8~10 $\mu$m range. Concentric grooves, with increasing steepness toward the periphery of the lens, form a concave contour with optical properties, like that of a convex lens. A Fresnel lens array consists of several Fresnel lens, curved around a detector at the focal length of the lens, covers the
entire near $0.5\pi$ FOV, and creates a set of beams on the space that have a uniform angular. Fig. 2.4 shows the beam pattern of Fresnel lens (AA0.9GIT1, Fresnel Technologies) used in our experiment.

**Figure 2.4:** Visibility pattern a pyroelectric sensor with an 11-element Fresnel lens array as the collection optics. Details of the beam pattern are shown on the second beam from the left. (a) Top view. (b) Side view.

We employ the Fresnel lens array to modulate the visibility of our sensors, such that each sensor can observe events uniformly distributed over $M$ angles. In this way, the visibility of the $i^{th}$ sensor in the $j^{th}$ node will be,

$$v^{(i,j)} = (\theta_1^{(i,j)}, \ldots, \theta_M^{(i,j)}).$$

(2.4)

Fig. 2.5 (a) and (b) illustrate the sensor response signals, when a human target passes through the FOVs of sensors with and without visibility modulation. We can see that after visibility modulation sensor response signals have higher amplitude and more spatial information of the target. The lower limit upon angular velocity of target can be improved by almost 10 times, as there are 11 sub-detection-regions formed by a lens array consisting of 11 elements. For a human target 2 m away from sensors, moving below a speed of 2 m/s, we must employ visibility modulation with a Fresnel lens array, to improve the lower limit of 1.1 rad/s upon target angular
velocity by 10 times to that of near 0.1 rad/s [83].

A radial type of sensor module was used as shown in Fig. 2.6. This sensor module has eight pyroelectric detectors with the Fresnel lenses arranged in a circular fashion to monitor a 360° FOV around the sensor. The visibilities are shown in Fig. 2.7. Table 2.1 lists the association between detection regions and 8 sensors. It can be seen that, in such a simple visibility coding scheme, the angular resolution of each sensor module is of roughly π/8.

A two-column radial type of sensor module was designed and used as shown in Fig. 2.8. This sensor module has eight pyroelectric detectors with the Fresnel lenses arranged in two columns. In each column, the FOV of one sensor is modulated as 30°, i.e. consisting of 4 elements of lenses, and there is a 14° shift of FOV among four sensors. Two separable detection regions are formed, each of which has an average angular resolution of 11°. The visibilities are shown in Fig. 2.9. Such a visibility design also can facilitate the process of data-object-association: when sensors of one
**Figure 2.6:** The radial sensor module used for the characterization of Fresnel lens Field Of View.

**Figure 2.7:** Visibilities of one sensor module.
Table 2.1: Visibility coding scheme for radial sensor modules.

<table>
<thead>
<tr>
<th>Detection Region</th>
<th>Sensors</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[1 0 0 0 0 0 0]</td>
</tr>
<tr>
<td>2</td>
<td>[1 1 0 0 0 0 0]</td>
</tr>
<tr>
<td>3</td>
<td>[0 1 0 0 0 0 0]</td>
</tr>
<tr>
<td>4</td>
<td>[0 1 1 0 0 0 0]</td>
</tr>
<tr>
<td>5</td>
<td>[0 0 1 0 0 0 0]</td>
</tr>
<tr>
<td>6</td>
<td>[0 0 1 1 0 0 0]</td>
</tr>
<tr>
<td>7</td>
<td>[0 0 0 1 0 0 0]</td>
</tr>
<tr>
<td>8</td>
<td>[0 0 0 1 1 0 0]</td>
</tr>
<tr>
<td>9</td>
<td>[0 0 0 0 1 0 0]</td>
</tr>
<tr>
<td>10</td>
<td>[0 0 0 0 1 1 0]</td>
</tr>
<tr>
<td>11</td>
<td>[0 0 0 0 0 1 0]</td>
</tr>
<tr>
<td>12</td>
<td>[0 0 0 0 0 1 1]</td>
</tr>
<tr>
<td>13</td>
<td>[0 0 0 0 0 0 1]</td>
</tr>
<tr>
<td>14</td>
<td>[1 0 0 0 0 0 1]</td>
</tr>
</tbody>
</table>
node associated with two different detection region fire, it can be judged for sure that there are at least two objects moving. Table 2.2 lists the association between detection regions and 8 sensors.

When four above sensor nodes are deployed in a 9 m × 9 m room, we will have a global visibility distribution shown in Fig. 2.10. It can be seen that four local regions are formed, marked as region I, II, III, and IV. In each local region, if only one object moves, those associated sensors can provide high angular measurement resolution. The detection range of each sensor can be roughly adjusted to be within a local region by increasing the firing threshold set in the embedded processing. With such a visibility configuration, the number of human objects to be tracked will
Figure 2.9: Visibilities of one tracking sensor module.

Table 2.2: Visibility coding scheme for two-column radial sensor modules.

<table>
<thead>
<tr>
<th>Detection Region</th>
<th>Sensors</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[1 0 0 0 0 0 0 0]</td>
</tr>
<tr>
<td>2</td>
<td>[1 1 0 0 0 0 0 0]</td>
</tr>
<tr>
<td>3</td>
<td>[0 1 0 0 0 0 0 0]</td>
</tr>
<tr>
<td>4</td>
<td>[0 1 1 0 0 0 0 0]</td>
</tr>
<tr>
<td>5</td>
<td>[0 0 1 0 0 0 0 0]</td>
</tr>
<tr>
<td>6</td>
<td>[0 0 1 1 0 0 0 0]</td>
</tr>
<tr>
<td>7</td>
<td>[0 0 0 1 0 0 0 0]</td>
</tr>
<tr>
<td>8</td>
<td>[0 0 0 0 1 0 0 0]</td>
</tr>
<tr>
<td>9</td>
<td>[0 0 0 0 1 1 0 0]</td>
</tr>
<tr>
<td>10</td>
<td>[0 0 0 0 0 1 0 0]</td>
</tr>
<tr>
<td>11</td>
<td>[0 0 0 0 0 1 1 0]</td>
</tr>
<tr>
<td>12</td>
<td>[0 0 0 0 0 0 1 0]</td>
</tr>
<tr>
<td>13</td>
<td>[0 0 0 0 0 0 1 1]</td>
</tr>
<tr>
<td>14</td>
<td>[0 0 0 0 0 0 0 1]</td>
</tr>
</tbody>
</table>
Figure 2.10: Visibilities of four sensor modules and four local detection regions formed for event validation to reduce false alarms.

be less than four and the whole data-object-association scheme becomes two steps: which local region one object are in and which angular displacement measurements should be associated with this object.

In the experiments, we made two kinds of sensor modules, one for analog feature extraction and another for digital feature extraction. [48] suggests that the lens array with more elements can yield a better performance in terms of identification rates. To extract digital event index sequences, on the other hand, visibility of each sensor should be modulated by some pseudo-random codes. Fig. 2.11 shows a sensor module with three sensor-lens units for spectral feature extraction.

Another kind of one-column radial type of sensor module was designed and used for walker recognition based on digital event index sequences as shown in Fig. 2.12. This sensor module has four pyroelectric detectors with the Fresnel lenses arranged in one column. The FOV of one sensor is modulated according to pseudo-random
Figure 2.11: The full-lens sensor module.

Figure 2.12: The one-column radial sensor module.
Table 2.3: Visibility coding scheme for the digital feature extraction sensor module.

<table>
<thead>
<tr>
<th>Lens Array</th>
<th>Visibility</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[0 0 0 1 0 1 1 0 1 0 0]</td>
</tr>
<tr>
<td>2</td>
<td>[0 0 0 1 1 0 0 1 1 0 0]</td>
</tr>
<tr>
<td>3</td>
<td>[0 0 1 0 1 0 1 0 1 0 0]</td>
</tr>
<tr>
<td>4</td>
<td>[0 1 0 0 1 1 0 0 1 0 0]</td>
</tr>
</tbody>
</table>

Figure 2.13: The Eden 333 rapid prototyping device and the high-pressure water jet [2].

codes. Table 2.3 lists the association between detection regions and four sensors.

All the optical heads including lens holders and support bases were designed using a CAD software SOLIDWORKS. Their construction were accomplished via a prototyping device EDEN 333, shown in Fig. 2.13, which makes 3D models out of a resin material, sprayed onto the building tray in those intended locations, and concretes them by casting UV light on. There is another type of resin material called support resin that is used to build support structures or buffer layers, and can be easily washed off in a high-pressure water jet shown in Fig. 2.13.
2.2 Problem Statements

2.2.1 Single Human Tracking

In the configuration space, the configuration state \( x(t) \), served to represent spatial-temporal varying radiation of the target, \( \psi(r, t) \), for example, its position and velocity in a 2-D plane. Under the discrete sampling scheme, the target state at time \( k \) is a random vector \( x_k \). The target state evolution can be modeled by the following linear system:

\[
x_{k+1} = \Phi x_k + \Gamma w_k
\]  

(2.5)

where the object position and displacement \( x_k = [x_k, v_{xk}, y_k, v_{yk}]^T \), the dynamics uncertainty \( w_k = [w_{xk}, w_{yk}]^T \),

\[
\Phi = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \Gamma = \begin{bmatrix} \gamma & 0 \\ 0 & \gamma \\ 1 & 0 \\ 0 & 1 \end{bmatrix}.
\]

Here, \( 0 < \gamma < 1 \), \( x \) and \( y \) denote the Cartesian coordinates of the target, \( v_x \) and \( v_y \) denote the displacements between two events in the \( x \) and \( y \) directions, respectively, roughly proportional to the velocities in the two directions. The system noise is a zero mean Gaussian white noise, that is, \( w_n \sim N(0, \sigma^2 w I_2) \), where \( I_2 \) is the \( 2 \times 2 \) identity matrix. The initial state \( x_0 \) describes the targets initial position and velocity. A priori for the initial state \( p(x_0) \) also needs to be specified for the model; we assume \( x_0 \sim N(0, P_0) \).

This dynamics model in Eq. (2.5) is Markov and can be represented by the conditional density \( p(x_{k+1}|x_k) \). The problem can be stated as, how to track the
target configuration state sequence $x_{1:k} = [x_1, \cdots, x_k]$ with maximum posterior probability from sensor response signals $s_{1:k} = [s_1, \cdots, s_k]$, given the observation model likelihood $p(s_k|x_k)$ and state dynamic prior $p(x_{k+1}|x_k)$,

$$x_{1:k} = \arg \max_x p(x_{1:k}|s_{1:k}), \quad \text{(2.6)}$$

where $p(s_k|x_k)$ is derived from the sensor model, visibility function, and prior on noise. It is known as the maximum a posterior (MAP) Bayesian tracking problem.

By using the visibility coding scheme in Table 2.1, the sensor response signals can be digitized and interpreted as angular displacements with respect to each sensing node. Thus the measurement equation can be written as

$$\tilde{z}_{k}^{(i,j)} = \begin{cases} 
\tan^{-1} \left[ \frac{y_k - y_{k(i,j)}}{x_k - x_{k(i,j)}} \right] + v_n & \text{if sensor } (i, j) \text{ fired}; \\
0 & \text{if sensor } (i, j) \text{ not fired.} 
\end{cases} \quad \text{(2.7)}$$

where sensor $(i, j)$ means the $j$th sensor in the $i$th node at the location of $(x_{(i,j)}, y_{(i,j)})$, the measurement noise $v$ is a zero mean Gaussian white noise, that is, $v_n \sim \mathcal{N}(0, \sigma_v^2)$, measurement $\tilde{z}$ is a random variable uniformly distributed over a set of angles $(\theta_1^{(i,j)}, \cdots, \theta_M^{(i,j)})$.

The general sequential Bayesian tracking problem is to recursively calculate some degree of belief in the state $x_k$ with given measurement $z_{1:k}$. Its solution includes two parts: prediction and filtering, given by

$$p(x_k|z_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}|z_{1:k-1})dx_{k-1},$$

$$p(x_k|z_{1:k}) = \frac{p(z_k|x_k)p(x_k|z_{1:k-1})}{p(z_k|z_{1:k-1})}, \quad \text{(2.8)}$$

where $p(z_k|z_{1:k-1})$ can be viewed as a normalizing constant. The probabilistic model
of the state evolution $p(x_k|x_{k-1})$ is defined by the system equation in Eq. (2.5) and
the known statistics of $\{w_k\}$. The likelihood function $p(z_k|x_k)$ is defined by the
measurement equation in Eq. (2.7) and the known statistics of $\{v_k\}$.

### 2.2.2 Multiple Human Tracking

Likewise, in multiple human tracking, the $l^{th}$ object state at time $k$ is a random
vector $x^l_k$. The object state evolution can be modeled by the following linear sys-
tems:

$$x^l_{k+1} = \Phi x^l_k + \Gamma^l w^l_k$$  \hspace{1cm} (2.9)

where the object label $l = 1,\ldots,L$, the object position and displacement $x^l_k = [x^l_k, \delta x^l_k, y^l_k, \delta y^l_k]^T$, the dynamics uncertainty $w^l_k = [w_{xk}, w_{yk}]^T$,

$$
\Phi = \begin{bmatrix}
1 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 \\
0 & 0 & 0 & 1
\end{bmatrix}, \quad \Gamma^l = \begin{bmatrix}
\gamma^l & 0 \\
1 & 0 \\
0 & \gamma^l \\
0 & 1
\end{bmatrix}.
$$

Here, $0 < \gamma^l < 1$, $x$ and $y$ denote the Cartesian coordinates of the object, $\delta_x$ and
$\delta_y$ denote the displacements in the $x$ and $y$ directions, respectively, between two
events, roughly proportional to the velocities in two directions. The system noises
are zero mean Gaussian white noises, that is, $w^l \sim \mathcal{N}(0, \sigma^2 \text{I}_2)$, where $\text{I}_2$ is the
$2\times2$ identity matrix. The initial state $x_0$ describes the objects’ initial positions and
velocities. A priori for the initial state $p(x^l_0)$ also needs to be specified for the model;
we assume $x_0^l \sim \mathcal{N}(0, P^l_0)$.

This dynamics model in Eq. (2.9) is Markov and can be represented by the
conditional density $p(x_{k+1}|x_k)$. The problem can be stated as, how to determine
object configuration state sequences $\mathbf{x}_{1:k} = [\mathbf{x}_1, \cdots, \mathbf{x}_k]$ with maximum posterior probability from sensor response signals $\mathbf{s}_{1:k} = [\mathbf{s}_1, \cdots, \mathbf{s}_k]$, given the observation model likelihood $p(\mathbf{s}_k | \mathbf{x}_k)$ and state dynamics prior $p(\mathbf{x}_{k+1} | \mathbf{x}_k)$,

$$
\mathbf{x}_{1:k} = \arg \max_{\mathbf{x}} p(\mathbf{x}_{1:k} | \mathbf{s}_{1:k}). \tag{2.10}
$$

Note that $p(\mathbf{s}_k | \mathbf{x}_k)$ is derived from the sensor model, visibility function, and prior on noise. It is known as the maximum a posterior (MAP) Bayesian tracking problem.

By using the visibility coding scheme in Table 2.2, the sensor response signals $\mathbf{s}_{1:k}$ can be digitized into event indexes and then interpreted as angular displacements with respect to each sensing node $\mathbf{z}_{1:k}^{(i,j)}$. To form the new observation likelihood model $p(\mathbf{z}_k | \mathbf{x}_k)$, the measurement equation is written as

$$
\mathbf{z}_{k}^{(i,j)} = \begin{cases} 
\tan^{-1} \left[ \frac{y_{k} - y^{(i,j)}}{x_{k} - x^{(i,j)}} \right] + v & \text{if event } (i, j) \text{ happens;} \\
0 & \text{otherwise.}
\end{cases} \tag{2.11}
$$

where event $(i, j)$ means a motion detection in the $j^{th}$ detection region of the $i^{th}$ node whose origin is $[x^{(i,j)}, y^{(i,j)}]$, the measurement noise $v$ is a zero mean Gaussian white noise, that is, $v \sim \mathcal{N}(0, \sigma_v^2)$, measurement $z$ is a random variable uniformly distributed over a set of angles $[\theta_1^{(i,j)}, \cdots, \theta_M^{(i,j)}]$.

To handle the possible occlusion of multiple persons, local visibilities are designed for event validation, shown in Fig. 2.10. When event indexes from four sensor nodes are received, an area-object-association is made first according to the global sensor visibility geometry shown in Fig. 2.10 and Fig. 3.16. After such a measurement validation, only two measurements are associated with one object,
denoted as,

\[ z_k^{(l)} = \left\{ z_k^{(i,j)} : Pr \left[ \chi_k^{(l,i,j)} \right] > 0 \right\}. \tag{2.12} \]

where \( \chi_k^{(l,i,j)} \) is the event that measurement \( z_k^{(i,j)} \) originates from the \( l \)th object.

The general sequential Bayesian tracking problem requires that we recursively calculate some degree of belief in the state \( x_k^{(l)} \) with validated measurements \( z_{1:k}^{(l)} \). Its solution includes two parts: prediction and filtering, given by

\[
p \left[ x_k^{(l)} | z_{1:k-1}^{(l)} \right] = \int p \left[ x_k^{(l)} | x_{k-1}^{(l)} \right] p \left[ x_{k-1}^{(l)} | z_{1:k-1}^{(l)} \right] dx_{k-1}^{(l)},
\]

\[
p \left[ x_k^{(l)} | z_{1:k}^{(l)} \right] = \frac{p \left[ z_k^{(l)} | x_k^{(l)} \right] p \left[ x_k^{(l)} | z_{1:k-1}^{(l)} \right]}{p \left[ z_k^{(l)} | z_{1:k-1}^{(l)} \right]}, \tag{2.13}
\]

where \( p \left[ z_k^{(l)} | z_{1:k-1}^{(l)} \right] \) can be viewed as a normalizing constant. The probabilistic model of the state evolution \( p \left[ x_k^{(l)} | x_{k-1}^{(l)} \right] \) is defined by the system equation in Eq. (2.9) and the known statistics of \( \{w_k\} \). The likelihood function \( p \left[ z_k^{(l)} | x_k^{(l)} \right] \) is defined by the measurement equation in Eq. (2.11) and the known statistics of \( \{v_k\} \).

### 2.2.3 Human Identification

The whole object recognition problem includes two parts: data learning and hypothesis testing. We here use linear regression models (LRMs), Gaussian mixture models (GMMs), and hidden Markov models (HMMs) to learn the statistics of the feature data.

#### Statistical Models of Features

A LRM can be set up as
1) \( Y = \{ y_1, y_2, ..., y_T \} \): feature label set.

2) \( X = \{ x_1, x_2, ..., x_T \} \): feature data set.

3) \( w = \{ w_1, w_2, ..., w_L \} \): linear regression vector, where \( L \) is the dimension of each feature \( x_i \).

4) \( \theta_L = \cup \{ w_i, \mu_i^Y, \Sigma_i^Y \} \): model’s parameters, where \( \mu_i^Y \) and \( \Sigma_i^Y \) are the mean and covariance of labels \( Y \) in the \( i^{th} \) cluster.

The likelihood of a new feature \( x \) to associate with the LRM is

\[
p(x|\theta_L) = \sum_{i=1}^{M} \mathcal{N}(wx|\mu_i^Y, \Sigma_i^Y),
\]

where \( \mathcal{N} \) is the normal distribution and \( M \) is the number of clusters.

A GMM can be set up as

1) \( Y = \{ y_1, y_2, ..., y_T \} \): hidden data label set.

2) \( X = \{ x_1, x_2, ..., x_T \} \): feature data set.

3) \( p = \{ p_1, p_2, ..., p_K \} \): mixture coefficients of \( N \) Gaussian components.

4) \( \theta_G = \cup \{ p_i, \mu_i, \Sigma_i \} \): model’s parameters, where \( \mu_i \) and \( \Sigma_i \) are the mean and covariance of \( i^{th} \) components.

The association likelihood of a new feature \( x \) upon the GMM is

\[
p(x|\theta_G) = \sum_{i=1}^{N} p_i \mathcal{N}(x|\mu_i, \Sigma_i).
\]

45
The hidden discrete label $y$ of one feature $x$ is

$$y = \arg \max_i \mathcal{N}(x|\mu_i, \Sigma_i). \quad (2.16)$$

LRM and GMM differ in that the former has a linear regression vector obtained from a supervised training process and the latter has a mixture coefficient vector acquired from an unsupervised training process.

An HMM can be set up as

1) $Y = \{y_1, y_2, ..., y_T\}$: hidden state sequence.

2) $X = \{x_1, x_2, ..., x_T\}$: feature sequence.

3) $A = \{a_{ij}\}$, $a_{ij} = P(y_{t+1} = j|y_t = i)$: state transition probability matrix.

4) $B = \{b_{ij}\}$, $b_{ij} = P(x_t = m|y_t = i)$: emission probabilities.

5) $\pi = \{\pi_i\}$: initial state distribution.

6) $\theta_H = \{A, B, \pi\}$: model’s parameters.

For given parameters $\theta_H$, the likelihood of the hidden state sequence $Y$ and the observed data $X$ to associate with the model is

$$p(X, Y|\theta_H) = \left[ \prod_{t=1}^{T-1} a_{y_t y_{t+1}} \right] \left[ \prod_{t=1}^{T} b_{y_t x_t} \right] \pi_{y_1} \quad (2.17)$$

To estimate the membership of one sequence with respect to one model, we estimate the hidden state sequence first and use that to calculate the association likelihood.
Data Learning

Given the conditional probability density \( p(X, Y|\theta) \), the maximum likelihood (ML) estimation of the parameter vector \( \theta \) from the training data \( X \) and their labels \( Y \), known or hidden, is

\[
\hat{\theta} = \arg \max_{\theta} \ln p(X|\theta) = \arg \max_{\theta} \ln \sum_{Y} p(X, Y|\theta) \\
= \arg \max_{\theta} F(q(Y), \theta),
\]

where \( q(Y) \) is the probability distribution over \( Y \) and the free energy function \( F( ) \) is

\[
F(q(Y), \theta) = \sum_{Y} q(Y) \ln \frac{p(X, Y|\theta)}{q(Y)} \\
= \sum_{Y} q(Y) \ln p(X, Y|\theta) - \sum_{Y} q(Y) \ln q(Y).
\]

For the supervised learning problem, the preset data label distribution is known as \( q(Y) = \delta(Y - Y_0) \), so \( X \) is “complete data” for model training. Such is the case for LRM, and the problem is then reduced to

\[
\hat{\theta} = \arg \max_{\theta} \ln p(X|Y_0, \theta).
\]

Thus the linear regression vector can be achieved through straightforward linear least square error (LSE) methods.

For the unsupervised learning problem, the data labels \( Y \) are hidden variables, the expectation-maximization (EM) algorithm is hence developed. In each E step, the distribution over the hidden variables \( q(Y) \) is obtained, given an estimation of
model parameters $\theta^t$.

$$q^{t+1}(\cdot) = \text{arg} \max_{q(Y)} \mathcal{F}(q(Y), \theta^t).$$

(2.21)

In each M step, the model parameters $\theta$ is optimized, given the updated hidden variable distribution $q^{t+1}(Y)$.

$$\theta^{t+1} = \text{arg} \max_{\theta} \mathcal{F}(q^{t+1}(Y), \theta).$$

(2.22)

Hypothesis Testing

For an unlabeled spectral feature, or an event sequence, $x$, we will have $K + 1$ hypothesis for $K$ registered objects, $\{H_0, ..., H_K\}$, to test. The hypothesis $H_0$ represents “none-of-the-above”. The decision rule then is

$$x \in \begin{cases} 
H_0, & \text{if } \max_i p(x|H_i) < \gamma \\
H_i: i = \text{arg} \max_i p(x|H_i), & \text{otherwise}
\end{cases}$$

(2.23)

where $p(x|H_i)$ is the likelihood of $x$ to associate with the $i^{th}$ hypothesis and $\gamma$ is a selected acceptance/rejection threshold. There are several special cases of this general statement. For the verification problem, $i = 1$; for the closed-set identification problem, $\gamma = -\infty$. 48
Chapter 3

Event Capture and Feature Extraction

This chapter introduces the general strategies of event-based tracking and feature-based identification. Both the event and feature are defined in the pyroelectric signal space. Signal processing techniques such as Kalman filtering, FFT, and band-pass sine filtering are proposed to detect the events embedded in the sensory data. A Gaussian mixture model (GMM) clustering method fueled with a Markov chain Monte Carlo simulation is developed to model noises in signals, so as to choose the event acceptance/rejection thresholds accordingly. Event sequences can be further smoothed by a hidden Markov model (HMM) filter and validated by the prior knowledge on the geometry of sensor configuration and deployment. Two kinds of features can be extracted from pyroelectric sensor (array) signals, the binary event index sequence of a sensor array and the spectral segment of the data of one event. The first one is defined as the digital feature; the second one as the analog feature.

3.1 Tracking and Identification Strategies

In general, as shown in Fig. 3.1, the tracking strategy for distributed sensors includes: event detection, to alert system to the presence of targets; event digitization,
Figure 3.1: A general tracking scheme for distributed pyroelectric sensors.

to convert sampled signals into event indexes; event registration, to reject the errors of event indexes by a simple HMM event smoother; motion inference, to fuse the measurements from all the sensor modules and convert them into measured positions of targets; and trajectory smoothing, to estimate/predict the state trajectory of targets in configuration space by using Bayesian tracking techniques.

In our implementation, we have assigned the procedures of event detection and event digitization to the embedded computation of slave sensor modules, not only to make a better use of distributed computation resources, but to improve the wireless communication efficiency and fidelity as well. The message between each slave and master only contains 2 bytes, one for the slave ID, another for the on/off status of 8 sensors, aside from the message head and tail. The master module synchronizes the communications with those slave modules. As there are only a limited number of 8-bit event indexes according to visibility design and there also exists a dynamic coherence among those event indexes, we have employed a hidden Markov model filter in the master module to register and smooth the received event indexes, while rejecting the errors.

On the other hand, as shown in Fig. 3.2, the identification strategy for the distributed sensors includes: event detection; event data acquisition, to acquire the data inside a event window after that event is validated; feature extraction, to extract the feature, digital or analog, out of the event data based on the predefined
Figure 3.2: A general identification scheme for distributed pyroelectric sensors.

feature-to-event association; membership estimation, to estimate the membership of one feature with respect to all the feature models of enrolled individuals; and object identification, to make a decision on the identity of the human object under testing according to the estimated membership likelihood vector and the inference rule.

3.2 Event Detection and digitization

An event could be defined as an occurrence of interest in spatial-temporal space distinguishable from its environment and repeatable in multiple trials. In our case, it is an instance in which the thermal flux collected by a pyroelectric sensor is above a threshold and its response data can be associated with one or several specific human motions, such as moving across one detection region.

As we have obtained an impulse response model of the pyroelectric sensing system, in Eq. (2.3), it is reasonable to consider estimating the thermal flux upon each sensor by inversion of this model in a mean square sense. This linear model based inversion process poses itself as a Kalman estimation problem and its solution yields a low-pass IIR filter. We also consider two additional signal processing techniques: windowed FFT and band-pass sine filter, and compare the performances and computation costs of these three methods. To digitize the processed signal into
an 8-bit event index, we have to select a suitable threshold value to be tested. In practice, we have used the expectation maximization (EM) algorithm to estimate signals distribution, collected from random walks of one human target, in terms of Gaussian mixture models (GMMs), and then get the covariance of process noise.

### 3.2.1 Kalman Filtering

The Kalman filter can estimate a process by using a form of feedback control: the filter makes a priori estimate of the process state, initialized with a guessed value, with a linear process model and then obtains an improved posteriori estimate with a feedback of noisy measurements [84].

In this project, since we need to estimate the input of thermal flux from the response signal of pyroelectric sensors, the sensor model has to be extended to include input as a process state. However, such a state augmentation requires a pre-known input dynamics. As the thermal flux of a moving source is modulated by the Fresnel lens array into rectangular pulses, the dynamics of input can thus be assumed as a constant in a short period. This approach yields a filter with a high gain property in the low frequency range of an integrator. The estimate of the input can track the real value effectively with some acceptable delay if the change of the thermal flux is not too fast.

The discrete sensor model augmented with the process disturbance \( w(k) \) and measurement noise \( v(k) \) is given by

\[
\begin{align*}
\alpha(k + 1) &= \Phi \alpha(k) + \Gamma \Psi(k) + w(k), \\
\gamma(k) &= H \alpha(k) + J \Psi(k) + v(k),
\end{align*}
\]

(3.1)

where \( \Phi \in \mathcal{R}^{n \times n}, \Gamma, H^T \in \mathcal{R}^n, J \in \mathcal{R} \) is the discrete dynamics model of pyroelectric
sensors derived from sensor model in state space with a sampling rate of $t_s$; $n$ is the order of the sensor model; $\alpha(k) \in \mathcal{R}^n$ is sensor state; $s(k) \in \mathcal{R}$ is measured voltage signal; $w(k) \in \mathcal{R}^n$, $v(k) \in \mathcal{R}$ are the process and measurement noises respectively.

To estimate the thermal flux $\Psi(k)$, the process model has be rewritten as

$$
\begin{bmatrix}
\alpha(k+1) \\
\Psi(k+1)
\end{bmatrix} = \Phi_e \begin{bmatrix}
\alpha(k) \\
\Psi(k)
\end{bmatrix} + w_e(k),
$$

$$
s(k) = H_e \begin{bmatrix}
\alpha(k) \\
\Psi(k)
\end{bmatrix} + v(k),
$$

(3.2)

where $\Phi_e = \begin{bmatrix} \Phi & \Gamma \\ 0 & 1 \end{bmatrix}$; $H_e = \begin{bmatrix} H & J \end{bmatrix}$; $w_e \in \mathcal{R}^{n+1}$ is the augmented process disturbance.

If the covariance of $w_e(k)$ and $v(k)$ are $Q \in \mathcal{R}^{(n+1) \times (n+1)}$ and $R \in \mathcal{R}$ respectively, we have the Kalman filter, which can achieve the optimal estimate of $\Psi(k)$ in the sense of minimum mean square error, in the form of

$$
\begin{bmatrix}
\dot{\alpha}(k+1) \\
\dot{\Psi}(k+1)
\end{bmatrix} = \Phi_e - LH_e \begin{bmatrix}
\dot{\alpha}(k) \\
\dot{\Psi}(k)
\end{bmatrix} + Ls(k),
$$

$$
\dot{\Psi}(k) = H_0 \begin{bmatrix}
\dot{\alpha}(k) \\
\dot{\Psi}(k)
\end{bmatrix},
$$

(3.3)

where $H_0 = [0_{1 \times n} \ 1]$; $L = MH_e^T(H_eMH_e^T + R)^{-1}$ is the feedback gain of measurement; $M$ is the solution of discrete arithmetic Racati equation of $M = \Phi_e(M - MH_e^TR^{-1}H_eM)\Phi_e^T + Q$. By using the MATLAB control toolbox function `kalman`(\Phi_e, H_e, Q, R), we can obtain $L$ right away.

53
The resultant Kalman filter based on the sensor model at a sampling rate of 20Hz is

\[ K(z) = \frac{0.2683z^{-1} - 0.6804z^{-2} + 0.6465z^{-3} - 0.2920z^{-4} + 0.0600z^{-5}}{1 - 3.2679z^{-1} + 4.2386z^{-2} - 2.8129z^{-3} + 1.007z^{-4} - 0.1648z^{-5}}. \]  

(3.4)

The bode plot of the filter is shown in Fig.3.3. The low-mid-frequency part of the Kalman filter is roughly an inversion of the sensor impulse response function.

---

**Figure 3.3:** Bode plot of the Kalman filter based on the sensor model.

When a thermal flux value is larger than its threshold, the corresponding bit of the event index will be set as ‘1’; otherwise it is ‘0’. The whole filter architecture is shown in Fig. 3.4.

We select a threshold based on the covariance of the measured noise signal as

\[ C_{threshold} = \gamma \sigma \{s(k)\} ||K(z)||_2, \]

(3.5)

where \( \gamma \) is a constant; \( K(z) \) is the kalman filter; \( ||*||_2 \) is the \( \mathcal{H}_2 \) norm of discrete
linear systems.

### 3.2.2 Robust Gaussian Mixture Estimation of Noise

The histogram of the response signal generated by a random walk of one human target is illustrated in Fig. 3.5 (a). It is notable that the process noise is Gaussian. By using the Gaussian mixture model (GMM) estimation through expectation maximization (EM) scheme, we can obtain the distribution model of noise, as shown in Fig. 3.5 (b) and then obtain the value of $\sigma \{s(k)\}$. In order to estimate the noise distribution embedded in the sensor response signals, we develop a variational Bayesian Gaussian mixture model (VBGMM) clustering method fueled with a Markov chain Monte-Carlo (MCMC) simulation, for the algorithmic simplicity and numeric robustness of the MC simulation and the global optimality of the variational Bayesian scheme.

The data distribution is written as the linear superposition of Gaussians,

$$p(y) = \sum_{k=1}^{K} \pi_k \mathcal{N}(y | \mu_k, \Sigma_k), \tag{3.6}$$

where $y$ is the data set in size of $N$, $\pi_k$ is the $k^{th}$ mixing coefficients, $\mu_k$ and $\Sigma_k$ are the mean and covariance of the $k^{th}$ Gaussian component. Besides, the responsibility $z_k$, a hidden variable, describes the membership between the data set $y$ and the $k^{th}$
Gaussian component and its conditional distribution is defined as

$$p(z_k|\pi, \mu, \Sigma, y) = \delta(z_k - \pi_k \mathcal{N}(y|\mu_k, \Sigma_k)) \sum_{j=1}^{K} \pi_j \mathcal{N}(y|\mu_j, \Sigma_j).$$ \hspace{1cm} (3.7)$$

In the variational Bayesian inference schemes [85], all the model parameters are initialized with conjugate priors for the tractability. Usually, Dirichlet distribution is chosen for the prior on mixing coefficients $\pi$, normal distribution on the means $\mu$, and Wishart distribution on the precisions $\Sigma^{-1}$, inverse covariances. Thus, the prior distribution on the GMM parameters is

$$p(\pi, \mu, \Sigma) = D(\pi) \prod_{k=1}^{K} \mathcal{N}(\mu_k) \mathcal{W}(\Sigma_k^{-1}),$$ \hspace{1cm} (3.8)$$

where $D(\pi|\lambda)$, $\mathcal{N}(\mu_k|m_k, \beta_k \Sigma_k^{-1})$, and $\mathcal{W}(\Sigma_k^{-1}|S_k, \alpha_k)$ are initialized with values of $\lambda^0$, $m_k^0$, $\beta_k^0$, $\alpha_k^0$, and $S_k^0$ and can be simulated by the MATLAB statistics toolbox functions $\text{gamrnd}$, $\text{mvnrnd}$, and $\text{wishrnd}$. The variational update formulas for those super-parameters are

$$\begin{align*}
\lambda_k^{\text{new}} &= \lambda_k^0 + N_k \bar{y}_k, \\
m_k^{\text{new}} &= \frac{1}{\beta_k^{\text{new}}} (\beta_k^0 m_k^0 + N_k \bar{y}_k), \\
\beta_k^{\text{new}} &= \beta_k^0 + N_k, \\
\alpha_k^{\text{new}} &= \alpha_k^0 + N_k, \\
S_k^{\text{new},-1} &= S_k^{0,-1} + N_k Y_k \\
&\quad + \frac{\beta_k^0 N_k (\bar{y}_k - m_0)(\bar{y}_k - m_0)^T}{\beta_k^{\text{new}}},
\end{align*}$$ \hspace{1cm} (3.9)-(3.13)$$

56
where

\[ N_k = \sum_{i=1}^{N} z_{k,i}, \]  
(3.14)

\[ \bar{y}_k = \frac{1}{N_k} \sum_{i=1}^{N} z_{k,i} y_i, \]  
(3.15)

\[ Y_k = \frac{1}{N_k} \sum_{i=1}^{N} z_{k,i}(y_i - \bar{y}_k)(y_i - \bar{y}_k)^T, \]  
(3.16)

are the sufficient statistics of data. The posterior distribution of the GMM parameters is

\[ p(z, \pi, \mu, \Sigma | y) = p(z | \pi, \mu, \Sigma, y)p(\pi, \mu, \Sigma | y). \]  
(3.17)

The Gibbs sampling scheme is then given by

1) Initialize \( \lambda^0, m^0, \beta^0, \alpha^0, \) and \( S^0. \)

2) Generate \( [z^0, \pi^0, \mu^0, \Sigma^0], \) set \( t = 1. \)

3) Update \( \lambda, m, \beta, \alpha, \) and \( S. \)

4) Generate \( [\pi^t, \mu^t, \Sigma^t] \) from \( p(\pi, \mu, \Sigma). \)

5) Generate \( z^t \) from \( p(z | \pi, \mu, \Sigma, y). \)

6) Set \( t = t + 1 \) and go to step 3).

The procedure repeats until convergence.

Fig. 3.6 illustrates how the event time windows are generated from the sensory data by using signal processing architecture of Fig. 3.4.
Figure 3.5: Response signals of one multiplex sensor from a random walk by a human target. (a) the histogram, (b) noise extraction by mixture Gaussian model estimation.

Figure 3.6: Event detection through a pair of sensors with similar FOVs by using a Kalman filter. (a) Raw data. (b) Filtered signals. (c) Digitized signals. (d) Logic signals. (e) Event windows.

3.2.3 Windowed FFT

Another alternative approach for event detection is windowed discrete Fourier transform (WDFT). Fig. 3.7 shows the windowed power spectrum density of the sensory
Figure 3.7: Event detection from windowed power spectrum density of sensory data. (a) Raw data. (b) Data windowed spectra. (c) Digitized signals. (d) Event windows.

data and the derived event time windows. Compared with Kalman filtering, real-time power spectrum estimation requires much more computation resource.

3.2.4 Band-Pass Sine Filtering

When a human target passes through the Fresnel lens modulated FOV, illustrated in Fig. 3.8, the response signals shown in Fig. 3.9 are generated. Based on the pre-known visibility function, we can read out the angular measurements for that specific sensor. More specifically, by using band-pass filtering, \( i.e. \sin[2\pi(1 : N)/N] \), we can capture and amplify the features of interest. Then, the signals can be further digitalized by threshold testing. Fig. 3.10 and Fig. 3.11 show the filtered and digitalized results, when \( N \) was chosen as 12. Fig. 3.12 shows the the processed sensory data by a sine filter and the derived event time windows.
Figure 3.8: Human target trajectory with two sensor modules set up.

Figure 3.9: Sensor response signals of one sensor pair of two sensor modules.
Figure 3.10: Filtered signals of sensors after band-pass filtering.

Figure 3.11: Digitalized signals of sensors after threshold testing.
Figure 3.12: Event detection by matched filtering, threshold testing, and low-pass filtering. (a) Raw data. (b) Filtered signals. (c) Digitized signals. (d) Logic signals. (e) Event windows.

3.3 Event Registration

As shown in Table 2.1, there are only 16 unique codes, among 256 possible 8-bit event indexes. In implementation, they can be further reduced, due to the specific sensor FOV. To reject false alarms, the event indexes generated by each slave module, as the human target moves through the object space, can be modeled as a two-layer first-order finite-state discrete-time Markov processes. The two-layer Markov process is a stochastic process, in which an underlying process \( \{ ..., s_{t-1}, s_t, s_{t+1}, ... \} \) is not observable but only can be inferred through another set of processes \( \{ ..., o_{t-1}, o_t, o_{t+1}, ... \} \). Fig. 3.13 shows a 4 state hidden Markov model, each with two possible observations. Because of the spatial coherence of the events, one event is restricted to follow only events in its spatial neighborhood.

A hidden Markov Model (HMM) with \( N \) possible states and \( M \) possible obser-
where $\pi \in \mathbb{R}^{N \times 1}$ is the state initial probability distribution for $N$ states, $\pi_i = P\{s_1 = i\}$ is the likelihood that the initial state is the $i^{th}$ state, $A \in \mathbb{R}^{N \times N}$ is the state transition probability distribution, $a_{ij} = P\{s_{t+1} = i|s_t = j\}$ is the probability of that the $i^{th}$ state could transit to $j^{th}$ state, $B \in \mathbb{R}^{N \times M}$ is the state emission probability distribution, $b_{ij} = P\{o_t = \chi_j|s_t = i\}$ is the probability of that the observation of $i^{th}$ state is $j^{th}$ event index.

Since the HMM accommodates multiple observations for each state, one could consider each true alarm with multiple event indexes for the errors in event indexes due to system noise. One can also define an “uncertainty” observation, when a motion generates some un-calibrated event indexes or some event indexes which make no geometric senses. In this case, the hidden Markov model will be augmented as

$$
\tilde{B} = \begin{bmatrix}
0_{N \times M} & 0 \\
B & \epsilon_{N \times 1}
\end{bmatrix},
$$

$$
\tilde{A} = \begin{bmatrix}
0 & \pi \\
0_{N \times 1} & A
\end{bmatrix},
$$

where $0_{N \times M}$ is a zero matrix of size $N \times M$; $\epsilon_{N \times 1}$ is the assumed probability distribution of the “uncertainty” observation and the “initial” state is also augmented into the state transition matrix with the predefined probability distribution of $\pi$.

The procedure to estimate most likely state sequence from the hidden Markov model, i.e. the forward part of Viterbi algorithm [86], is given in Section 4, where it is used for HMM based human tracking.
Figure 3.13: Block diagram of a hidden Markov model with four states $s_1, s_2, s_3, s_4$ and each state has two possible observations as an example.

Fig. 3.14 (a) shows an observation sequences with and without false alarms, with a flip probability of 0.1 for each bit. By using HMM smoothing, the mis-registration rate can be reduced from 58% to 17%. The estimated and true state sequences are shown in Fig. 3.14 (b).

3.4 Event Validation and Interpretation

The master node synchronizes the communication of slave nodes and collects the data in turn. In one cycle, it gets the events from all the sensor nodes, validates their feasibility, removes ineffective events according to the geometry of sensor visibilities, and frames a new composite event index before sending it to the host.

Fig. 3.15 (a) shows the structure of the data packet transmitted from slaves to the master. The packet contains two bytes of $FF$ as “header”, two bytes of 10 as “tail”, one byte for node ID, and another byte for event index. In our geometric
Figure 3.14: An observation sequence with false alarms is smoothed by 8 state HMM filtering.

Figure 3.15: The structure of event packets, (a) transmitted by slaves via wireless channels, (b) transmitted between the master and the host via a cable.
design, four local detection areas are formed, shown in Fig. 2.10. If an object moves inside one of these areas, the sensors associated with that area should fire, forming a valid event. For example, for an object moving in area I of Fig. 3.16., a legal event has the non-zero four higher bits of the event reported by sensor node 1 and the non-zero four lower bits by sensor node 2. If only one of these two sensors reports a non-zero signal, the master will regard it as invalid and clear those bits to zero. After the event validation, the master will package all four event bytes into a single message, as illustrated in Fig. 7.1 (b), and send it to the host. The host will localize the objects by interpreting this composite event in an expectation-maximization (EM) way.

Once we have obtained the 8-bit event index of a sensor module, we can convert it into an angular displacement with respect to this node, according to its visibility.
Figure 3.17: An event sequence from one sensor node and its interpretation as angular displacements.

function, modulated by Fresnel lens arrays. Fig. 3.17 illustrates a sequence of events transmitted by one sensor node and its translation in terms of angular displacements at the host side. As a result, each sensor module turns out to be a thermal bearing sensor.

3.5 Feature Extraction

A feature characterizes patterns of phenomena with quantitative measures. Any functional feature in recognition should be characterized by its universality over all objects under testing, distinctiveness between any two individuals, invariance over a period of time, and feasibility for quantitative measures. In speaker recognition, among the most successful features are those based on the cepstral coefficients, because irrespective of the technique (linear predication or filter bank analysis) used in their computation [80]. Pyroelectric sensor signals, by contrast, are event based. Our research effort upon feature representation bifurcates into two branches:
spectral analysis of event signals and statistics of digital event index sequences.

Figure 3.18: Two 4-bit event index sequences generated by two objects walking along the same path.

Once we have obtained the event index sequence of a sensor module, Fig. 3.18 illustrates a sequence of events after 10 times down-sampling transmitted by one sensor node, whose visibility is designed in Table 2.3. Human objects have different thermal distributions over space and habitual walking styles and will trigger different visibility patterns, which can be statistically modeled in HMMs. For each individual, one HMM is trained with a long training sequence generated by him/her.

The event data, on the other hand, can also be used to extract human object features for identification. Fig. 3.19 shows the sensor data spectra of two human objects walking at different speed levels, high, moderate, and low, among which spectral distinctions can be observed. This can be exploited for human identification purposes, as those spectral features can be collected repeatably with small variances, for a fixed sensor configuration and a fixed walking path as well.

Fig. 3.20 and Fig. 3.21 summarize the procedures of digital and analog feature extraction. The length of each digital feature for real-time object recognition is
Figure 3.19: Event data spectra generated by two objects walking at different speeds along the same path. Solid line: object A; dashed line: object B. (a) High speed. (b) Moderate speed. (c) Low speed.

fixed. Once it reaches the preset length, the algorithm resets itself awaiting next batch of event indexes coming in. Analog feature is event based. As one event happens, its data will be retrieved at once. The length of the event data is checked first to reject those trivial events. Its low-to-mid-band spectral segment is also checked against the universal background model, a linear regression model of invalid spectral features, to make sure its validity before being tested against all the hypotheses.
Figure 3.20: Digital feature extraction.

Figure 3.21: Analog feature extraction.
Chapter 4

Single Human Tracking

This chapter describes one human tracking scheme with a wireless pyroelectric sensor system. Based on the sensor module illustrated in Fig. 2.6, three Bayesian tracking techniques, namely Kalman filters, HMM filters, and particle filters, are tested and their performances are compared. The experimental tracking results of a four node prototype sensor system are also given.

4.1 Data Fusion and Tracking Synthesis

Human object tracking with multiple sensors is an intrinsic multi-sensor data fusion problem. Multi-sensor data fusion is such a process through which we combine readings from different sensor nodes, remove inconsistencies, and pull all the information together into one coherent structure. Various data fusion schemes and techniques have been proposed for combining measurements from many sensing nodes with limited accuracy and reliability, to achieve better accuracy and more robustness [82]. Fig. 4.1 shows the architecture used in our prototype sensor system. The slave nodes work at the raw data level. The master node works at the event level. The host carries out the data fusion at the feature and tracking level.
4.1.1 Motion Inference

Once we obtain the 8-bit event index of a sensor module, we can convert it into an angular displacement with respect to the node, according to the visibility code table. We even can improve the sensor resolution because of the visibility modulation by the Fresnel lens array. Specifically, using the data shown in Fig. 3.11, from two pairs of adjacent sensors. We know there are 11 sub-regions of the roughly $\frac{\pi}{2}$ FOV associated with each sensor, each sub-region thus having a FOV of $\frac{\pi}{22}$, as illustrated in Fig. 3.8. We can interpret the signals in Fig. 3.11 as target crossing over those sub-detection-regions. The results are shown in Fig. 4.2.

However, only after the human target leaves the FOVs of sensors, can the ambiguity of those angular displacements be broken. For simplicity, in our real-time implementation, the event indices are translated into the angular displacements at a lower resolution of $\frac{\pi}{8}$ by the visibility coding scheme in Table 2.1. The results are shown in Fig. 4.3. To improve the sensing resolution, we need to deploy more sensor modules.

For a distributed angular sensor system, a simple way for measurement linearization is grid approximation, shown in Fig. 4.4. That is to calculate the angular
Figure 4.2: Estimated angular displacements of two sensor modules for one human target in high resolution.

Figure 4.3: Estimated angular displacements in low resolution.
Figure 4.4: Grid approximation to linearize the measurements of angular displacements with respect to four sensor nodes.

displacements, with respect to each sensor node, of the grid points in the object space. When a set of angular displacement measurements are available, the Cartesian coordinates of nearest grid point can be chosen to estimate dynamic states of targets by Bayesian tracking schemes.

4.1.2 Trajectory Smoothing by Bayesian Tracking

The single-target tracking problem is usually posed as a dynamic estimation of a partially observable Markov process, and in most cases can be solved by Bayesian recursive filtering schemes in the general form of (2.8). For the linear and Gaussian tracking problem, Kalman tracking offers the optimal solution. HMM tracking provides the optimal recursion of the posterior density estimation, if the state space only consists of a finite number of discrete states. In many situations, linear, Gaussian, or finite discrete state assumptions do not hold. Gaussian approximation,
grid-based approximation, and particle filters are proposed to achieve suboptimal solutions [57].

If the true density is heavily skewed, the Gaussian approximations will have poor performance. Disadvantages of grid-based approach include that its state space need to be predefined, which consume much data storage space, and that each grid point is usually given the same weight, without making good use of target dynamics prior. The particle filter technique has heavy computation needs. For real-time implementation, the computation complexity and practical functionality of a tracking algorithm is more valued than its mathematical rigor. Thus, particle filter tracking is indeed least favorable in our case, but they still can serve as a benchmark to evaluate the performance of the whole system.

In the following paragraphs, we summarize the three Bayesian tracking strategies, namely Kalman, HMM, and Gaussian particle filters. Among the various particle filters, the Gaussian particle filter is chosen for its algorithmic simplicity, guaranteed convergence with small samples, and performance superiorities in tracking over extended Kalman filters and unscented Kalman filters [58]. The comparison of the performances and computation costs is given in next Section. Based upon those results, we have selected a Gaussian tracking scheme in real-time implementation, with grid approximations to linearize the angular displacement measurements.

### 4.1.3 Kalman Tracking

The Kalman tracking assumes that the posterior density at each step is Gaussian and can be parameterized by a mean and covariance. The corresponding prior on
state dynamics and likelihood on measurements will be

\[
p(x_k|x_{k-1}) = \mathcal{N}(x_k; \Phi x_{k-1}, Q),
\]
\[
p(y_k|x_k) = \mathcal{N}(y_k; Hx_k, R).
\]

where $\Phi$ and $H$ are known matrices defining the linear functions. The covariances of process and measurement noises are, respectively, $Q$ and $R$.

The Kalman tracking algorithm, derived from (2.8), can be viewed in the term of recursive pdf as

\[
p(x_{k-1}|y_{1:k-1}) = \mathcal{N}(x_{k-1}; m_{k-1|k-1}, P_{k-1|k-1}),
\]
\[
p(x_k|y_{1:k-1}) = \mathcal{N}(x_k; m_{k|k-1}, P_{k|k-1}),
\]
\[
p(x_k|y_{1:k}) = \mathcal{N}(x_k; m_{k|k}, P_{k|k}),
\]

where

\[
m_{k|k-1} = \Phi m_{k-1|k-1},
\]
\[
P_{k|k-1} = Q + \Phi P_{k-1|k-1} \Phi^T,
\]
\[
m_{k|k} = m_{k|k-1} + L_k(y_k - Hm_{k|k-1}),
\]
\[
P_{k|k} = P_{k|k-1} - L_kHP_{k|k-1},
\]

and $\mathcal{N}(x; m, P)$ is a Gaussian density with argument $x$, mean $m$, and covariance $P$, and

\[
S_k = HP_{k|k-1}H^T + R,
\]
\[
L_k = P_{k|k-1}H^T S_k^{-1},
\]

76
are the covariance of \( y_k - H m_{k|k-1} \), and the Kalman gain, respectively. For the Guassian posterior pdf, the mean, median and mode are identical. Therefore, the optimal estimation trajectory is \( \{ m_{k|k} \} \).

4.1.4 Hidden Markov Model Tracking

If a finite-state and finite-observation system at time \( k \) consists \( N_s \) discrete states \( \{ x^i_k, i = 1, ..., N_s \} \) and \( N_m \) discrete observation \( \{ y^l_k, l = 1, ..., N_m \} \), then the whole system can be described, via a series of probabilities, as

\[
p(x^i_k | x^j_{k-1}) = a_{ij},
\]

\[
p(y^l_k | x^i_k) = b_{li},
\]

where \( \sum_i a_{ij} = 1 \) and \( \sum_l b_{li} = 1 \).

Then the corresponding pdfs will be

\[
p(x_{k-1} | y_{1:k-1}) = \sum_{i=1}^{N_s} w^i_{k-1|k-1} \delta(x_{k-1} - x^i_{k-1}),
\]

\[
p(x_k | y_{1:k-1}) = \sum_{i=1}^{N_s} w^i_{k|k-1} \delta(x_{k-1} - x^i_k),
\]

\[
p(x_k | y_{1:k}) = \sum_{i=1}^{N_s} w^i_{k|k} \delta(x_k - x^i_k),
\]

where

\[
w^i_{k|k-1} = \sum_{j=1}^{N_s} w^j_{k-1|k-1} a_{ij},
\]

\[
w^i_{k|k} = \frac{w^i_{k|k-1} \sum_{l=1}^{N_m} b_{li} \delta(y_k - y^l_k)}{\sum_{j=1}^{N_s} w^j_{k|k-1} \sum_{l=1}^{N_m} b_{li} \delta(y_k - y^l_k)}. \tag{4.7}
\]
Therefore, at time $k$ the optimal estimation of state $x_k$ will be the state associated with the maximum component of $w_k|k$.

### 4.1.5 Gaussian Particle Filter Tracking

In general, a dynamic system can be described by the prior on dynamics $p(x_k|x_{k-1})$ and likelihood on measurements $p(y_k|x_k)$. Most complex systems cannot construct closed-form pdfs of posterior and prediction, $p(x_k|y_{1:k})$ and $p(x_{k+1}|y_{1:k})$. Therefore, Monte Carlo importance (MCI) sampling technique, as opposed to those dependence sampling schemes such as Gibbs sampling mentioned in Appendix II, is used to implement a recursive Bayesian filter by MC simulations.

Gaussian particle filters technique assumes that all densities are Gaussian to reduce the complexity [58]. The whole procedure includes two parts, measurement update and state prediction, to estimate $p(x_k|y_{1:k})$ and $p(x_{k+1}|y_{1:k})$, in the forms of $\mathcal{N}(x_k; \mu_k, \Sigma_k)$ and $\mathcal{N}(x_{k+1}; \bar{\mu}_{k+1}, \bar{\Sigma}_{k+1})$ respectively.

**Measurement update:**

1) Sampling: draw \( \{x_i^j\}_{i=1}^{N_s} \) from $\mathcal{N}(x_k; m_{k|k}, P_{k|k})$, where $m_{k|k}$ and $P_{k|k}$ are obtained from Kalman update;

2) Weight updating:

\[
\bar{w}_k^i = \frac{p(y_k|x_i^j)\mathcal{N}(x_k = x_i^j; \bar{\mu}_k, \bar{\Sigma}_k)}{\mathcal{N}(x_k = x_i^j; m_{k|k}, P_{k|k})};
\]  

(4.8)

3) Weight normalization:

\[
w_k^i = \frac{w_k^i}{\sum_{j=1}^{N_s} \bar{w}_k^j};
\]  

(4.9)
4) Mean and covariance updating:

\[ \mu_k = \sum_{i=1}^{N_s} w_k^j x_k^j \]
\[ \Sigma_k = \sum_{i=1}^{N_s} w_k^j (x_k^i - \mu_k)(x_k^i - \mu_k)^T, \]  \hspace{1cm} (4.10)

where \( p(y_k|x) \) is also Gaussian and given in (2.7).

State predication:

1) Sampling: draw \( \{x_k^i\}_{i=1}^{N_s} \) from \( N(x_k; \mu_k, \Sigma_k) \);

2) State evolution: draw \( \{x_{k+1}^i\}_{i=1}^{N_s} \) from \( p(x_{k+1}|x_k = x_k^i) \);

3) Mean and covariance updating:

\[ \bar{\mu}_{k+1} = \frac{1}{N_s} \sum_{i=1}^{N_s} x_{k+1}^i \]
\[ \bar{\Sigma}_{k+1} = \frac{1}{N_s} \sum_{i=1}^{N_s} (x_{k+1}^i - \bar{\mu}_{k+1})(x_{k+1}^i - \bar{\mu}_{k+1})^T. \]  \hspace{1cm} (4.11)

### 4.2 Results and Discussion

The whole tracking scheme was implemented in a 9 m × 9 m room. We also apply it to numerical simulations, not only to pretest its feasibility, but to investigate the correlations between sensor resolution, sensor deployment and tracking precision, as well as performance and robustness of the tracking system with different algorithms.
4.2.1 Simulation Results

Fig. 4.5 displays a snapshot of one target tracking with four sensing nodes. The empty circle represents the target, the shaded circle the estimated. Each sensor node’s resolution is $\pi/8$. It can be seen that despite the measurement error of one sensor node, the target still can be tracked according to the angular displacement measurements of other three nodes. Fig. 4.6 shows the true trajectories, in solid line, and estimated trajectories by HMM, Kalman, and particle filter tracking techniques, at a low false alarm rate, respectively. The histograms of particle filter tracking errors in x and y directions are given in Fig. 4.7. The standard deviation of the tracking errors are 1.2$m$ and 1.1$m$ respectively.

Fig. 4.8 shows the performance comparison of three different tracking schemes at different false alarm rates. We can see that given a low false alarm rate, all the tracking schemes perform well, and HMM tracking can even achieve better performance than Kalman tracking. When the false alarm rate is increased, the performance of HMM tracking degrades much more than do those of the other two.
Figure 4.6: The simulation results of tracking a human target with four radial sensor modules.

Figure 4.7: The histograms of tracking errors in x and y directions respectively.
Figure 4.8: Tracking errors of three tracking schemes given different false alarm rates.

Table 4.1: Comparison of Bayesian tracking algorithms

<table>
<thead>
<tr>
<th>Bayesian Tracking Algorithm</th>
<th>Computation Complexity</th>
<th>Robust Performance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kalman Filter</td>
<td>$O(N^2)+O(M)$</td>
<td>Moderate</td>
</tr>
<tr>
<td>HMM Filter</td>
<td>$O(M)$</td>
<td>Poor</td>
</tr>
<tr>
<td>Particle Filter</td>
<td>$P \times O(N^2)+O(P)+O(M)$</td>
<td>Best</td>
</tr>
</tbody>
</table>


Its poor tracking robustness should be ascribed to the lack of the velocity estimation in its algorithm. Particle filter tracking outperforms the other two schemes, but its performance improvement is less than 10% of that of Kalman tracking, due to the limitation of system sensing resolution.

Table 4.1 summarizes the computation complexity and performances of the three tracking schemes. Note that although in theory the computation complexity of particle filter is higher than that of Kalman tracking roughly by times of the number of particles, in MATLAB implementation this computation cost gap appears to be less,
### Table 4.2: Comparison of embedded signal processing algorithms

<table>
<thead>
<tr>
<th>Event Detection Algorithm</th>
<th>Computation Complexity</th>
<th>Performance</th>
<th>Mean Calculation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kalman Filter</td>
<td>$O(N_1)$</td>
<td>Moderate</td>
<td>Yes</td>
</tr>
<tr>
<td>Windowed FFT</td>
<td>$O(N_2 \log(N_2))$</td>
<td>Best</td>
<td>Yes</td>
</tr>
<tr>
<td>Sine Filter</td>
<td>$O(N_3)$</td>
<td>Moderate</td>
<td>No</td>
</tr>
</tbody>
</table>

$N_1$: order of the sensor impulse transfer function plus one.
$N_2$: length of the data window. $N_3$: order of the sine filter.

**Figure 4.9:** The computation time per echo in MATLAB with respect to different tracking algorithms and signal processing filters.

...only 7 times for 500 particles as shown in Fig. 4.9 (a), mainly due to the MATLAB’s intrinsic forte in parallel computation. Of course, we also can conclude the grid approximation in the Kalman tracking takes most of its computation. The Gaussian particle filter tracking performance is pretty sensitive to tuning parameters, from number of particles to the observation probability density model to the input noise level. To some extent, tuning up a proper observation distribution model turns out to be the key factor in obtaining a decent particle filter. Therefore, in the real-time implementation, we choose Kalman tracking scheme with grid approximation.

83
4.2.2 Experimental Results

Having the above simulated sensor configuration set up in a $9 \times 9$ m room, we built a prototype pyroelectric tracking system for one human target tracking. In the real-time tracking demo, only the packets of events in 6 bytes, illustrated in Fig. 7.1, are transferred through either the wireless channels, between slave sensor nodes and the master node, or the serial cable, between the master node and the host. But to evaluate the system performance with respect to different embedded signal processing techniques, we collected raw response signals via serial cables from all sensor nodes. Fig. 4.10 shows the response signals of those triggered channels of four nodes, when a human target walked across the room along its diagonal. After the event digitalization and registration, we can convert those temporal-spatial signals into angular displacements with respect to each sensor node, shown in Fig. 4.11. By using the Kalman tracking scheme with grid approximation, the tracking results with three event detection techniques for the embedded computation, namely windowed FFT, Kalman IIR filter, and sine FIR filter, are given in Fig. 4.12. The standard deviations of tracking errors for these three event detection schemes are 0.75 m, 0.85 m, and 0.82 m, respectively.

Table 4.2 shows the comparison of these three signal processing algorithms in their performance and costs. The windowed FFT has the best performance in event detection at the expense of $N\log_2(N)$ computation cost, not to mention the extra calculation of exponents. Besides, both Kalman filtering and windowed FFT also need to remove the mean values of the signals before processing. The computation time of these three techniques in MATLAB is shown in Fig. 4.9. It can be seen that the 12 point sine FIR filter has the lowest computation time. Therefore, we chose the band-pass sine filter for real-time embedded computation in a micro-controller.
Figure 4.10: The response signals at different channels of four sensor nodes for event detections.
Figure 4.11: The measured angular displacements from the four nodes.

Figure 4.12: The tracking results with real sensory data with different event detection schemes. circle: Kalman IIR; square: windowed FFT; diamond: sine FIR.
Chapter 5

Multiple Human Tracking

5.1 Introduction

Multiple human tracking is desirable yet challenging for many applications, from surveillance, intelligent space, smart video conference, computer vision, to robotics [6], [8], [9], [10], [11]. The dynamics of multiple targets can be modeled as coupled hidden Markov chains, show in Fig. 5.1. It tends to be ambiguous and confusing in perception and interpretation of sensory data as some objects are often occluded by other objects in the field of view (FOV) of sensors. From the perspective of tracker, the tracking errors of one object likely propagate into those of others. Object identification has been proposed to help reduce such a mutual interference in tracking [87], [19]. Besides, the varying number of objects within observation space gives rise to the issue of object number determination [88].

Most trackers consist of four components, namely object representation, localization, data-object-association, and motion filtering [10]. The way to implement all these parts in tracking and put them together is application dependent and yields different performances in robustness and efficiency of a tracker. For real-time applications, the system resource for tracking is limited and the computational complexity for a tracker is expected to be maintained as low as possible. Such
that those components are indeed put on various weights for different sensor systems. In video based applications, where the nonrigid aspect of objects are more interested, object feature extraction and representation play a more important role and consume most computation resource in processing data collected from the intensive sensor array on the focal plane [9], [10], [15]. For acoustic sensor tracking systems, by contrast, in which objects are assumed rigid and signal-to-noise ratios (SNRs) are assumably low due to the noisy and clutter environments, numerous data-object-association and motion filtering techniques, from conceptually optimal multiple hypothesis trackers (MHTs) [89], [90] to theoretically strict joint probability data association filters (JPDAFs) [91], [92], [93] to computationally intensive sequential Monte Carlo simulations [94], [95], apart from all kinds of variations of Bayesian filtering schemes [96], have been proposed and developed.
5.2 EM-Bayesian Tracking

In general, as shown in Fig. 5.2, the tracking strategy for distributed sensors includes: event detection, to capture the sensor response signals inspired by object motion, digitalize them into event indexes, and reject those false alarms; object localization, to decompose events into groups and associate different groups of events with different objects based on the prediction of Bayesian tracking and measurement gating; and motion tracking, to estimate the state trajectory of objects in configuration space by Bayesian tracking techniques.

In our implementation, we assign the procedures of event digitalization, including matched filtering, threshold testing, and low-pass filtering, to the embedded computation of slave sensor modules. The message between each slave and master only contains 2 bytes, one the slave ID, another the on/off status of 8 sensors, aside from message head and tail. The master module synchronizes the wireless communication and collects event indexes transmitted from slave modules. and rejects those false alarms, i.e. event validation.

5.3 EM Object Localization

The object localization in bearing sensor systems is mainly resolved in terms of least square solutions, closed-form or constrained, of a set of linear equations, formulated by spherical/hyperbolic/linear intersection of measurements. It relies on the prior of sensor observation statistical model only, without taking object motion dynamics into account, and localizes sources to fit the measurements in the maximum likelihood within the Cramér-Rao estimation bound. On the other hand, grid-approximation can offer a simple way to translate the angular displacement
Figure 5.2: The proposed EM Bayesian tracking scheme.

measurements into Cartesian coordinates.

Compared with the single object tracking, the arresting and challenging aspect of the multiple object tracking with multiple sensors is the data-object-association problem. It turns more intractable for motion sensing systems which only respond to object motions; When some objects stand still, or when some false alarms fire, it poses a challenge to manage the number of trackers, as the number of objects is unknown and varying. Therefore, we propose an EM scheme for multiple object localization. In each $E$ step, we assume different numbers of objects to decomposite a joint event of all the sensor nodes. In each $M$ step, we associate measurements to those assumed numbers of objects respectively, according to the their predicted positions of last iteration during the tracking stage. In other words, we maintain several tracking templates at the same time. Each of them describes a dynamic context of a possible number of objects. Over a period, we evaluate all the templates
and pick out the one with the maximum likelihood for rendering.

5.3.1 E Step: Event Decomposition

The composite event received by the host contains the status of all the sensors. Based upon the geometry of the sensor visibilities, the on/off status of the sensors of one tracking slave can be interpreted as the angular displacement in one of its two sub-detection regions. The whole object space is divided into four regions, each of which is associated with half sensors of two neighboring slave nodes. If we obtain two angular displacements associated with one detection region, a simple way for measurement linearization is grid approximation, shown in Fig. 5.3. That is to calculate the angular displacements, with respect to each sensor node, of the grid points in the divided object space. When a set of paired angular displacement measurements are available, the Cartesian coordinates of nearest grid points can be chosen to assign some of them to different objects, or reject some of them as false alarms, by the process of measurement-to-object-association. The number of objects, however, is unknown and needs an assumption on it. We assume all the possible numbers of objects for localization in our sensor system, from one to four.

5.3.2 M Step: Measurement to Object Association

Three main approaches that have been offered and studied for a reliable and effective data association are multi-hypothesis filtering (MHF), joint probabilistic data association (JPDA), and probabilistic multi-hypothesis filtering (PMHF). The underlying motivation of those probabilistic schemes is that the state is assumed to be normally distributed according to the latest estimate and covariance matrix, such that only the measurements falling inside the validation gate of a object are
Figure 5.3: Grid approximation to linearize the measurements of angular displacements with respect to four sensor nodes.

Alternative approaches based on Bayesian inference have also been proposed, which offers an integrated tracking scheme with implicit data association. These techniques are typified by the structure of a prior likelihood of the object state over a discrete state space and a weighting of that prior by the quantized spatial spectrum in form of a likelihood to produce a posterior likelihood of the object state. They have been constructed as recursive methods and produce optimal estimates, in the discrete state space, of the current state. However, they do not generate optimal track estimates over a batch of observations. Multiple pass extensions have been formulated at the expense of more computation intensity [97].

In between, there are several techniques that the data-association process is followed by a tracking process, where the tracking process provides predictions to initialize or aid the data-to-object-association in a sequential fashion [81].
our tracking scheme, for the simplicity, we build several templates at the same
time. Within each template, the number of targets is fixed. After a period, the
performance of all the templates are evaluated and the best one will be chosen as
the active one for rendering.

The key concepts of measurement-to-object association for a number of targets
are the joint event and and the validation matrix [92]. The joint event is denoted
as
\[ \chi = \bigcap_{j=1}^{m_k} \chi^{j l_j}, \]  \hspace{1cm} (5.1)
where \( \chi^{j l_j} \) is the event that the measurement \( j \) originated from object \( l_j, 0 \leq l_j \leq L, \)
and \( l_j \) is the index of the object to which measurement \( j \) is associated, and \( m_k \) is
the number of a new batch of validated measurements at time \( k \).

The validation matrix for an joint event \( \chi \) is defined
\[
\Omega(\chi) = [\omega_{jl}(\chi)],
\]  \hspace{1cm} (5.2)
with binary elements to indicate if measurement \( j \) lies in the validation gate for
target \( l \). Thus those elements corresponding to the associations assumed in event
\( \chi \) are given by,
\[
\omega_{jl}(\chi) = \begin{cases} 
1 & \text{if } \chi^{j l_j} \text{ occurs} \\
0 & \text{otherwise.}
\end{cases}
\]  \hspace{1cm} (5.3)
The construction of each \( \Omega(\chi) \) follows the two rules:

1) There are can be only one origin for a measurement.

2) At most one measurement could have originated from a target.

Those two rules might lead to several feasible joint event and validation matrices
by enumerations. To reduce the number of feasible joint events and computation
time, an individual validation gate can be assumed for each tracker. Only those measurements falling inside the gates of trackers will be counted in the procedure of association. Fig. 5.4 illustrates of a typical set of validation gates and validation matrices, where \( \hat{y}_1 \) and \( \hat{y}_2 \) are estimated object positions, \( y_1, y_2, \) and \( y_3 \) are position measurements falling inside validation gates, \( y_4 \) a position measurement outside gates.

By using Bayes’ rule, the probability of one joint event conditioned on all the measurements up to the present time \( k \) is obtained as [93],

\[
P\{\chi_k|z_{1:k}\} \propto p[z_k|\chi_k, z_{1:k-1}]P\{\chi_k|z_{1:k-1}\}
\]

(5.4)

where \( p[z_k|\chi_k, z_{1:k-1}] \) is the likelihood of the predicted measurements \( z_k \) for the joint event \( \chi_k \), derived from object dynamics and sensor observation model, and \( P\{\chi_k|z_{1:k-1}\} \) is the prior probability of the joint event, derived from the probability distributions of false measurements of and target detection rates [92].
The association probability $\beta^j_{kl}$ that measurement $j$ belongs to object $l$ at time $k$ may be obtained by summing over all feasible events for which this condition is true

$$\beta^j_{kl} = \sum_{\chi_k} P\{\chi_k|z_{1:k}\} \omega_{jl}(\chi_k).$$

(5.5)

The states of each object can be updated with the measurements weighted by those association probabilities. Alternatively, for the simplicity, we can also just use the measurements with the highest probability to update the object state estimation.

### 5.4 Data Fusion and Tracking Synthesis

Multi-sensor data fusion is such a process through which we combine readings from different sensor nodes, remove inconsistencies, and pull all the information together into one coherent structure. Various data fusion schemes and techniques have been proposed for combining measurements from many sensing nodes with limited accuracy and reliability, to achieve better accuracy and more robustness. Fig. 5.5 shows the architecture used in our prototype sensor system. Both the slave and master nodes work at the raw data and event level for event validation and smoothing. The host carries out the data association and fusion at the object and template level for the purposes of tracking and identification.

#### 5.4.1 Template Operation

The whole system running is template based. Each template represents a pattern of multiple object dynamics context, such as how many persons in the object space and who of them are registered beforehand. One template is initialized and terminated upon certain conditions to save the computation resource. For example, if over
Figure 5.5: Data fusion architecture.
quite a while only two objects are detected, the templates for four or more object tracking should be terminated; likewise, if only one object is detected active, the templates for one and two targets are kept running in case another object will come into observation space at any time. Over a period, the template suited for measurements with highest likelihood is chosen as the active one. Such is the case for identification, each template contains the feature statistical model for one enrolled object. The best fitted template will report the ID of enrolled objects, if they happen to be there, to help break through the multiple object tracking ambiguities.

5.4.2 Bayesian Filtering and Prediction

The single-object tracking problem is usually posed as a dynamic estimation of a partially observable Markov process, and in most cases can be solved by Bayesian recursive filtering schemes in the general form of (2.8). For the linear and Gaussian tracking problem, Kalman tracking offers the optimal solution. HMM tracking provides the optimal recursion of the posterior density estimation, if the state space only consists of a finite number of discrete states. In many situations, linear, Gaussian, or finite discrete state assumptions do not hold. Gaussian approximation, grid-based approximation, and particle filters are proposed to achieve suboptimal solutions [57].

If the true density is heavily skewed, the Gaussian approximations will have poor performance. Disadvantages of grid-based approach include that its state space need to be predefined, which consume much data storage space, and that each grid point is usually given the same weight, without making good use of object dynamics prior. The particle filter technique has heavy computation needs. For real-time implementation, the computation complexity and practical functionality
of a tracking algorithm is more valued than its mathematical rigor. Thus, particle filter tracking is indeed least favorable in our case, but they still can serve as a benchmark to evaluate the performance of the whole system.

In [51], we summarize the three Bayesian tracking strategies, namely Kalman, HMM, and Gaussian particle filters. Among the various particle filters, the Gaussian particle filter is chosen for its algorithmic simplicity, guaranteed convergence with small samples, and performance superiorities in tracking over extended Kalman filters and unscented Kalman filters [58]. The comparison of the performances and computation costs is given in Section VI. Based upon those results, we have selected a Gaussian tracking scheme in real-time implementation, with grid approximations to linearize the angular displacement measurements.

5.5 Results and Discussion

The whole tracking system has been implemented in a 9 m × 9 m room, shown in Fig. 5.6. First, we tracked one human object walking along a prescribed rectangular path. Then, two human objects were tracked in walking modes of one-follow-another and cross-over. The results of five walker identification in path-independent way are also given.

5.5.1 Case I: One object Tracking

Fig. 5.7 displays a snapshot of one human object tracking with four sensing nodes. The lightly and heavily shaded circles represent respectively the grid approximation and Kalman estimated positions of the object, and the shaded beams the bearing measurements by different nodes. Each sensor node’s angular resolution is roughly π/22. The four sensing nodes generated 8-bit event sequences shown in 5.12.
Figure 5.6: The experiment setup.

Figure 5.7: A snapshot of the one object tracking.
Figure 5.8: The 8-bit event sequences of four nodes for one object tracking.

Figure 5.9: The tracked trajectory of the human object, represented by circles, along the prescribed route (solid line).
tracked trajectory of the human object along a prescribed rectangular route is illustrated in Fig. 5.9. It can be seen that the initial position of the tracker is set as [0 0]. The tracking errors and its histogram are given in Fig. 5.10. The standard deviation of the tracking errors is 0.46 m for one object tracking.

### 5.5.2 Case II: Two object Tracking

Having the same sensor configuration, we tested the system for two human object tracking. Fig. 5.11 displays a snapshot of one human object tracking with four sensing nodes. The four sensing nodes generated 8-bit event sequences shown in 5.12. The tracked trajectories of two human objects following each other along a prescribed rectangular route are illustrated in Fig. 5.13. It can be seen that both initial positions of two trackers are set as [0 0]. The tracking errors and their histograms are given in Fig. 5.14. The standard deviations of the tracking errors are 0.45 m and 0.43 m for two objects respectively.

A more challenging scenario for multiple object tracking is the case when they
Figure 5.11: A snapshot of the two object tracking.

Figure 5.12: The 8-bit event sequences of four nodes for two object tracking.
Figure 5.13: The two tracked trajectories of two human objects, represented by circles and crosses, along the prescribed route (solid line).

Figure 5.14: The tracking errors of two trackers and their histogram.
walk along different paths and having a cross-over. There are indeed no effective solutions for the data association problem in general cases without discriminatory characteristics available about each object. But for some specific cases, such as when objects don’t change the directions of their velocities too much during the cross-over, we can resolve the data-object-association problem by using the prior on speeds and their predictions from Bayesian prediction. The four sensing nodes generated 8-bit event sequences shown in Fig. 5.15. Fig. 5.16 displays the tracking results when two objects walk along two different diagonals of the room. By using the predicted speeds, the trackers can follow the targets after the cross-over. The tracking errors and their histogram are given in Fig. 5.14. The standard deviations of the tracking errors are 0.38 m and 0.7 m for two objects respectively.
Figure 5.16: The estimated crossing trajectories of two human objects walking in three rounds. The tracking results are represented by circles and crosses and the prescribed crossing route by solid lines.

Figure 5.17: The tracking errors of two trackers and their histograms.
Chapter 6

Walker Recognition

6.1 Introduction

The human body is actually a black body and a good infrared source. The average human body radiates about 100 W/m² of power, which peaks at the wavelength of 9.55 µm [12]. The pyroelectric infrared (PIR) sensor used in this experiment is able to detect humans walking at moderate speeds [13]. From the thermal perspective, each person acts as a distributed infrared source; by properly sampling the IR field, the idiosyncrasies in how an individual carries himself/herself and the habits of how he/she moves can generate a statistically unique signature in the signal space.

The low cost passive infrared (PIR) motion detector, LiTaO₃, has attracted increasing applications as a low cost alternative to infrared cameras [29], [28], [98], [3], [26]. Using a Fresnel lens array as a spatial filter, to produce visibility modulation, allows pyroelectric sensors to capture the source motions and convert them into a temporal response. The response data spectra of a single detector can discriminate between two individuals, possibly due to the difference in the motions of their arms and legs [48]. The modulated visibilities of multiple sensors segment object space into many cells, each characterized by a unique visibility pattern of sensors. When a human walks through the visibility modulated object space, the sensors will gen-
erate a binary signal sequence [51]. Such visibility pattern sequences can be utilized to locate and recognize human objects.

The information obtained by pyroelectric temporal-spatial signal processing techniques can be used to extract the following information from the walker:

1) Walker detection: *Is there someone moving?*

2) Walker tracking: *What is his/her location?*

3) Walker recognition: *Who is the walker?*

Much research attention has been paid to the topics, [98], [3], [48], [51]. Here, we focus on the third topics.

There is a parallelism between pyroelectric spatial-temporal signal processing and speech signal processing, from the perspective of signal inference. Pyroelectric spatial-temporal signals are processed for human tracking and walker recognition; speech signals for word recognition and speaker recognition [60]. Both human motions and spoken words can be modeled as hidden Markov chains for tracking and recognition [61]. Multiple human tracking bears a resemblance to tracking words of multiple speakers, both utilize the concept of coupled multiple hidden Markov chains [99], [62]. For speaker recognition, there are text-dependent and text-independent solutions [63]; for walker recognition, path-dependent and path-independent. It is reasonable and attractive to borrow numerous techniques developed for speech signal processing, from signal feature representation to feature statistical model building, for the solutions of pyroelectric spatial-temporal signal processing and inference problems.

Similar to speaker recognition [63], walker recognition can be performed in two ways:
1) **Walker Identification.**

The automatic system must determine who is walking. If the walker belongs to a predefined set of known walkers, it is referred to as “closed-set walker identification”. However, if the set of known walkers is smaller than the potential number of objects, it is referred to as “open-set walker identification”. Adding a “none-of-the-above”, or “others”, option to closed-set identification gives open-set identification. The system performance is evaluated using identification rate.

2) **Walker Verification.**

In this approach walker identifies himself/herself and the system must determine if the person is who he/she claims to be. The system accepts or rejects the users according to a successful or unsuccessful verification. Sometimes this operation mode is also called as authentication. The system performance is evaluated using the False Acceptance Rate (FAR, those situations where an impostor is accepted) and the False Rejection Rate (FRR, those situations where a walker is incorrectly rejected), also known in detection theory as False Alarm and Miss, respectively. This framework gives us the possibility of distinguishing between the discriminability of the system and the decision bias. [63]. The performance can be plotted in a Receiver Operator Characteristic (ROC) plot, where the Detection Rate (DR = 1-FRR) is instead used in most cases.

Despite the different requirements of verification and identification, they are intrinsically related to one another. For an open-set identification system, the optimal decision threshold has to be obtained from the ROC curves generated from verification testing results. For the verification problem, the unknown walker’s data sample is compared to the database. If the claimed walker is among the $K$ best
matches, the walker is accepted and otherwise rejected [64].

In both cases (identification and verification), walker recognition techniques can be split into two main modalities:

1) **Path Independent.**

This is the general case, where the system does not know the path walked by the person. This operation mode is mandatory for those applications where the user does not know that he/she is being evaluated. This allows more flexibility, but it also increases the difficulty of the problem. In this mode one implicitly uses the typical path and speed of the walker derived from the histograms of locations and speeds. From the signal processing viewpoint, one must extract a specific statistical pattern for each person.

2) **Path Dependent.**

This operation mode implies that the same path is taken by everyone. The recognition relies on the comparison of the measured signals. Given that the response signals are usually speed dependent, there are two solutions. One is to collect the data of objects walking at different speeds, in the training phase. Another is to develop features that are less sensitive to the speed. Both improve with more data. This mode is useful for those applications with strong control over user input similar to its text dependent speaker recognition counterpart. [63].

We selected two types of features, upon which our walker recognition algorithms were based: the spectral vectors gathered from the analog event data and the digital event sequences produced as a walker crossed individual beams in the modulated object space. For path-dependent walker recognition purposes, we used linear regression models (LRMs) and hidden Markov models (HMMs) to cluster the training
data and to test new data. For path-independent walker recognition, an HMM was derived for each person from the training data and this set of enrolled models are used to estimate the identity likelihoods of newly generated signals. The experimental results demonstrate the effectiveness of the proposed methodology.

6.2 LRM and HMM Training

The model setup of LRM and HMMs are given in Chapter 2. Here, we describe the supervised learning algorithm, linear principle components regression, for LRM training and the unsupervised learning algorithm, Baum-Welch optimization, for HMM training.

6.2.1 Linear Principle Components Regression

We use linear principal components regression (LPCR) to find a linear regressive vector $F$, such that the label of a unknown spectral vector can be estimated, by an inner product of vector $F$ and the spectral vector $S$, i.e.,

$$ I = S \cdot F. \quad (6.1) $$

PCR uses the full spectrum and is factor-based. The spectrum information is not directly used in training, but first is subject to factor analysis to find those factors that have the largest influence on data variations. PCR can be divided into two steps: principle components analysis (PCA) followed by standard multiple linear regression (MLR). In the multiple hypothesis testing (MHT), the label of an unknown spectrum is estimated by Eq. 6.1 and then is checked against the clusters and their distribution obtained from the training process.
Principle Components Analysis (PCA).

PCA is essentially a spectrum decomposition of the spectra matrix $S$, retaining only those factors that have large singular values. The remaining factors associated with small singular values are assumed to be from noise, and therefore left out in the later regression phase. The singular value decomposition (SVD) of a spectral matrix $S$ can be represented by

$$S_{m \times n} = U_{m \times m} \Sigma_{m \times n} V_{n \times n}^T,$$  \hspace{1cm} (6.2)

where the $U$ and $V$ are orthogonal matrices, $m$ is the number of samples, $n$ is the number of spectral points in one spectrum of sensor data trace.

$$\Sigma = \begin{bmatrix}
\sigma_1 \\
\sigma_2 \\
\vdots \\
\sigma_r \\
0 \\
\vdots \\
0
\end{bmatrix} \hspace{1cm} (6.3)$$

$\Sigma$ is diagonal with nonnegative singular values in descending order. Thus the spectrum matrix $S$ can also be written as

$$S = \Sigma_{i=1}^{r} \sigma_i u_i v_i^T = \sigma_1 u_1 v_1^T + \sigma_2 u_2 v_2^T + \cdots + \sigma_r u_r v_r^T.$$  \hspace{1cm} (6.4)

$S$ can be approximated by its first $k$ singular values, assuming singular values for larger $k$ are negligible.
\[ S = \sum_{i=1}^{k} \sigma_i u_i v_i^T \]

\[ = \tilde{U}_{m \times k} \tilde{\Sigma}_{k \times k} \tilde{V}_{k \times n}, \]  

(6.5)

with \( k \ll m, n \).

The spectrum matrix \( S \) also can be defined as

\[ S \approx TP^T, \]  

(6.6)

where

\[ T_{m \times k} = \tilde{U}_{m \times k} \tilde{\Sigma}_{k \times k} \]

, 

\[ P = \tilde{V}_{n \times k} \]

, 

\[ SP = T \]

. \( T \) is called the score matrix, and \( P \) is called the factor matrix. Geometrically, \( P \) can be viewed as a new set of orthogonal coordinates spanning the inherent (true) dimensionality of the spectrum data matrix \( S \), and \( T \) is the projection (scores) of \( S \) onto the new coordinates systems. For convenience, we will call it k-space.

**Multiple Linear Regression (MLR)**

Once we obtain the underlying factors and their corresponding scores, MLR is performed to regress those scores. In the classification process, the Fourier spectrum is first projected onto those factors obtained during training, and the resulting scores are correlated with the calibration vector obtained in MLR. MLR is done in k-space.
We regress the spectrum vector against the score matrix $T_{m \times k}$, to get the regression vector $f_{k \times 1}$ in k-space, i.e., we find the least-squares solution of equation

$$I_{m \times 1} = T_{m \times k} f_{k \times 1}. \quad (6.7)$$

The least-squares solution for $f_{k \times 1}$ is

$$f_{k \times 1} = (T^T T)^{-1} T^T I = \tilde{\Sigma}^{-2} T^T I, \quad (6.8)$$

where

$$\tilde{\Sigma}^{-2} = \begin{bmatrix} \frac{1}{\sigma_1^2} \\ \sigma_1^2 \\ \frac{1}{\sigma_2^2} \\ \sigma_2^2 \\ \cdots \\ \frac{1}{\sigma_k^2} \end{bmatrix} \quad (6.9)$$

Since the regression vector $f_{k \times 1}$ is in the k-space, in the prediction process, we first must get the score (projection) of a measurement spectrum, and then find the inner product of $f_{k \times 1}$ and the score.

Finally, from Eq. 6.7, Eq. 6.8, and Eq. 6.6 the regression vector can be written as follows

$$F_{n \times 1} = P_{n \times k} f_{k \times 1},$$

$$= P_{n \times k} \tilde{\Sigma}^{-2} T^T I,$$

$$= \tilde{V}_{n \times k} \tilde{\Sigma}^{-2} (\tilde{U}_{m \times k} \tilde{\Sigma}_{k \times k})^T I,$$

$$= \tilde{V}_{n \times k} \tilde{\Sigma}^{-1} \tilde{U}_{m \times k}^T I. \quad (6.10)$$

The advantage of this method is that, it is easy to understand and eliminate the
problem of overfitting (since \( m >> k \), and columns of are orthogonal). The number of factors to include can be decided by calculating the PRESS (prediction residual error sum of squares) with cross-validation as a function of the number of factors included. This is usually a convex function, and we are looking for the number of factors where a minimum occurs [48].

### 6.2.2 Baum-Welch EM Optimization

The traditional training algorithm for hidden Markov model is an expectation-maximization (EM) algorithm known as the Baum-Welch algorithm. It is a maximum likelihood method, or, with a simple modification, a penalized maximum likelihood method, which can be viewed as maximizing a posterior probability density over the model parameters. The Baum-Welch algorithm is an iterative algorithm that increase posterior likelihood at each iteration until a maximum is reached. In each iteration, a forward-backward (\( E \)) step computes the probabilities of state sequence \( S \) conditioned on the current parameters \( \theta \), then an \( M \)-step updates the parameter \( \theta \).

The forward and backward probabilities \( \alpha^{(t)} \) and \( \beta^{(t)} \) are given by [61]

\[
\begin{align*}
\alpha_i^{(1)} &= \pi_i b_{ix_1}, \\
\alpha_j^{(t+1)} &= \left[ \sum_{i=1}^{N} \alpha_i^{(t)} a_{ij} \right] b_{jx_{t+1}}, \\
\beta_i^{(T)} &= b_{ix_T}, \\
\beta_i^{(t)} &= b_{ix_t} \left[ \sum_{i=1}^{N} a_{ij} \beta_j^{(t+1)} \right]. (6.11)
\end{align*}
\]

The \( M \)-step is expressed in terms of \( n_{ij}^{(t)} \), the posterior probability that there
was a transition between state $i$ and state $j$ at time step $t$ given $X$ and $\theta$,

\[ n_{ij}^{(t)} = \sum_{s} P(S|X, \theta) \delta(s_t = i, s_{t+1} = j) \]

\[ = \frac{1}{Z_n} \alpha_i^{(t)} \alpha_{ij} \beta_j^{(t+1)}, \quad (6.12) \]

where $Z_n$ is a normalizing constant such that $\sum_{i,j} n_{ij}^{(t)} = 1$. Then the $M$-step is [61]

\[ \pi_i' = \frac{\sum_{j=1}^{I} n_{ij}^{(1)}}{\sum_{t=1}^{T-1} \sum_{j'=1}^{I} n_{ij'}^{(t)}}, \]

\[ a_{ij}' = \frac{\sum_{t=1}^{T-1} \sum_{j'=1}^{I} n_{ij'}^{(t)}}{\sum_{t=1}^{T-1} \sum_{j'=1}^{I} n_{ij'}^{(t)}}, \]

\[ b_{im}' = \frac{\sum_{t=1}^{T} \sum_{j=1}^{I} n_{ij}^{(t)} | s.t. x_t = m}{\sum_{t=1}^{T} \sum_{j=1}^{I} n_{ij}^{(t)}}. \quad (6.13) \]

### 6.3 LRM and HMM Testing

Given the model setup in Chapter 2, the likelihood of a new feature $x$ to associate with the LRM is

\[ p(x|\theta_L) = \sum_{i=1}^{M} \mathcal{N}(wx|\mu_i^Y, \Sigma_i^Y), \quad (6.14) \]

where $\mathcal{N}$ is the normal distribution, $M$ is the number of clusters, $\mu_i^Y$ and $\Sigma_i^Y$ are the mean and covariance of labels $Y$ in the $i^{th}$ cluster.

For given parameters $\theta_H$, the likelihood of the hidden state sequence $Y$ and the observed data $X$ to associate with the model is

\[ p(X, Y|\theta_H) = \prod_{t=1}^{T-1} a_{y_{t}y_{t+1}} \prod_{t=1}^{T} b_{y_{t}x_{t}} \pi_{y_1}, \quad (6.15) \]

To estimated the membership of one sequence with respect to one model, we es-
timate the hidden state sequence first and use that to calculate the association likelihood. The posterior probability of the hidden variables $Y$ given $X$ and $\theta$ is given by

$$p(Y|X, \theta_H) = \frac{1}{\sum_Y p(X, Y|\theta_H)} \left[ \prod_{t=1}^{T-1} a_{y_t,y_{t+1}} \right] \left[ \prod_{t=1}^{T} b_{y_t|x_t} \right] \pi_{y_1} \quad (6.16)$$

### 6.4 Experimental Results and Discussion

The two recognition schemes, based on analog and digital features respectively, were implemented by using the TI's micro-controller and RF transceiver combo of MSP430149 and TRF6901. More details about the computation and communication platform implementation can be referred in [51]. Fig. 6.1 shows the experiment setup. First we collect the event data and event index sequences and train the feature models. Then the (log)likelihood of each feature is evaluated against different models in real time. Once the decision is made, an associated pre-recorded 16-bit voice file will be played by WAVPLAYER to announce the recognition result.
6.4.1 Feature Model Training

Figure 6.2: The feature clusters of one LRM with respect to different walking speeds.

Linear regression models map the analog features of signals into clusters in label space. For each registered object, we collect three sets of training data at high, moderate, and slow walking speeds. Through linear regression approach, we can obtain the parameters $\theta_L$ for each model by supervised learning, where we assign a nominal labels for each feature and obtain the regression vector through solving a linear inversion problem. Fig. 6.2 shows the regression results of training data for one human object. The clusters, each with 20 sets of analog features, are centered around the three nominal labels: $(0.867, -0.5)$, $(0, 1)$, and $(-0.867, -0.5)$.

Hidden Markov models, by comparison, characterize the statistics of a finite-state sequence for training. Their model parameters $\theta_H$ are obtained by random initialization and updated after the iterations of expectation-maximization in light of likelihood of how well the data fit models. By using MATLAB Statistics Toolbox function `hmmtrain`, we can estimate the transition and emission matrices, $A$ and
B, from an initial guess as to their values. With another function *hmmdecode*, we can compute the posterior state probabilities of testing sequences generated by different human objects.

One issue about HMM training is to chose the length of training and testing sequences. Fig. 6.3 illustrates the impact of the length of training sequences and testing sequences for training and testing HMMs. It can be seen that the length of training data beyond 1200 provides little improvement of the average log-likelihoods of testing sequences. Then, for a HMM obtained from the training data in the length of 2000, when the length of testing data is longer than 300, their average log-likelihoods become nearly linear to the sequence length. Therefore, in this case we chosen the length of 2000 for training sequences, 300 for testing sequences.

**Figure 6.3**: The average log-likelihoods of testing sequences against trained HMMs for (a) training sequences in different length; (b) testing sequences in different length.

Another issue about HMM training is to selection of its order, that is, the size of the square transition matrix, A. Fig. 6.4 shows the average log-likelihoods of testing data of five persons against pre-trained HMMs in different orders of one
Figure 6.4: The average log-likelihoods of testing sequences of five persons against HMMs in different orders of Bob.

person, Bob. Each set of testing data includes 20 event index sequences generated by one object. It can be seen that, though all four models can be used to identify Bob, the 6th order model yields the maximum log-likelihood margin between acceptance and rejection. In this way, we can choose the order of feature statistical model, which yet is not necessarily identical for all enrolled objects, for real-time implementations. One well known problem about HMM training is that its EM based learning algorithm is gradient based and hence susceptible to stuck into local optima. In practice, different guesses of initial model parameter values are tried for training before choosing the final model by comparison.

6.4.2 Path-Dependent Recognition

In this operation mode, objects walk along a prescribed path back and forth. The sensor system then extracts analog or digital features out of response signals, and checks them against the feature models, LRM and HMMs respectively.
**Figure 6.5**: ROC curves of walker verification of two objects using LRM

Fig. 6.5 shows the ROC plots of two objects, Bob and Jason, for a walker verification system based on LRM. After choosing different decision thresholds \( \gamma \) in Eq. (2.23), we obtain different false alarm rates and detection rates. When the decision threshold is reduced, higher detection rates as well as higher false alarm rates are achieved. If the system curve is moved toward the upper left zone, smaller error rates are achieved (better performance). The trade-off \( \gamma \) can be chosen along the equal error rate (EER) line, where false alarm equals miss probability, that is, the situation of balanced performance. In this case, the values of \( \gamma \) at balanced performance are 0.6 and 0.4 for two persons respectively.

No only suitable for verification, those compromised decision thresholds can also be used for open-set object identification in which the objects under examination could be both registered or unregistered. For those unregistered objects, the recognition system labels them as “Others”. After building the LRM for those two persons, the data of nine other persons walking in the way of mimicking the speeds
and gaits of these two are collected and tested against the two feature models. The results of such an open-set object identification are listed in Tab. 6.1. It should be noted that the system performance can be improved dramatically if these nine objects don’t try to fool the recognition system but instead walk in their natural habits.

Using Hidden Markov Models

![Log-likelihoods](image)

**Figure 6.6**: Log-likelihoods of (a) five walkers’ testing data against one walker’s HMM; (b) one walker’s testing data against five walkers’ HMMs.

By contrast, HMMs are unsuitable for walker verification and open-set identification but instead the close-set identification. Fig. 6.6 (a) shows the verification case where the event sequences of five walkers are tested against one walker’s HMM.
It turns out that the right walker’s data cannot achieve the maximum log-likelihood at most time. However, when the testing data of that walker are checked against five HMMs, his own along with other four walkers’, the expected maximum likelihoods can be achieved, shown in Fig. 6.6 (b), which is the case of closed-set identification. This might be caused by the much information loss with respect to individuals after digitalization of sensor signals. Yet the digital features advantage in their simple processing and high data compression ratio, ideal for wireless transmission [51], [99].

Tab. 6.2 shows the close-set identification results for those five persons. It can be seen that in identification among the five objects the lowest identification rate is 75%, the highest is 95%, and the average is 84%. Fig. 6.7 (a) shows the trend of average identification rate decreases with the increase of number of objects to be recognized.

### 6.4.3 Path-Independent Recognition

Using HMMs in the similar procedure we can identify the walkers in the path-independent way. Each object walks randomly inside a room, until the enough length of event index sequences is reached. During the training phase, a training event sequence of length 2000 was collected and used to derived an HMM for each

<table>
<thead>
<tr>
<th>Results</th>
<th>Bob</th>
<th>Jason</th>
<th>Evan</th>
<th>Andrew</th>
<th>Aiqin</th>
<th>Steve</th>
<th>Zhiya</th>
<th>Fang</th>
<th>Tao</th>
<th>Jingbo</th>
<th>Yunbo</th>
<th>Tianyao</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bob</td>
<td>60%</td>
<td>90%</td>
<td>75%</td>
<td>10%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>20%</td>
</tr>
<tr>
<td>Jason</td>
<td>5%</td>
<td>45%</td>
<td>15%</td>
<td>45%</td>
<td>45%</td>
<td>45%</td>
<td>45%</td>
<td>45%</td>
<td>45%</td>
<td>45%</td>
<td>45%</td>
<td>20%</td>
</tr>
<tr>
<td>Evan</td>
<td>45%</td>
<td>30%</td>
<td>35%</td>
<td>35%</td>
<td>35%</td>
<td>35%</td>
<td>35%</td>
<td>35%</td>
<td>35%</td>
<td>35%</td>
<td>35%</td>
<td>20%</td>
</tr>
<tr>
<td>Andrew</td>
<td>10%</td>
<td>15%</td>
<td>20%</td>
<td>25%</td>
<td>25%</td>
<td>25%</td>
<td>25%</td>
<td>25%</td>
<td>25%</td>
<td>25%</td>
<td>25%</td>
<td>20%</td>
</tr>
<tr>
<td>Aiqin</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>20%</td>
</tr>
<tr>
<td>Steve</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>20%</td>
</tr>
<tr>
<td>Zhiya</td>
<td>10%</td>
<td>20%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>20%</td>
</tr>
<tr>
<td>Fang</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>20%</td>
</tr>
<tr>
<td>Tao</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>20%</td>
</tr>
<tr>
<td>Jingbo</td>
<td>10%</td>
<td>20%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>30%</td>
<td>20%</td>
</tr>
<tr>
<td>Yunbo</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>20%</td>
</tr>
<tr>
<td>Tianyao</td>
<td>10%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>20%</td>
</tr>
</tbody>
</table>

![Table 6.2: Close-set path-dependent identification results using HMMs](image-url)
person. Fig. 6.8 shows the training sequences of five people. During the testing phase, event sequences of length 500 were collected for identity testing. Each person was tested for 20 times. The results of close-set path-independent walker identification is shown in Tab. 6.3. Predictably, with the increase of the registered objects, the identification rate in path-independent recognition also drops off, as shown in Fig. 6.7 (b).

Figure 6.7: Average identification rates with respect to numbers of objects. (a) path-dependent; (b) path-independent.
Figure 6.8: The 4-bit training sequences of five people who randomly walk inside the room.

Table 6.3: Close-set path-independent identification results using HMMs

<table>
<thead>
<tr>
<th>Results</th>
<th>Bob</th>
<th>Jason</th>
<th>Andrew</th>
<th>Aiqin</th>
<th>Zhiya</th>
<th>Fang</th>
<th>Tao</th>
<th>Yunbo</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bob</td>
<td>100%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Jason</td>
<td>100%</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Andrew</td>
<td></td>
<td>75%</td>
<td>15%</td>
<td>10%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aiqin</td>
<td>30%</td>
<td></td>
<td>65%</td>
<td></td>
<td></td>
<td>5%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Zhiya</td>
<td></td>
<td>20%</td>
<td>25%</td>
<td>50%</td>
<td>5%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fang</td>
<td>5%</td>
<td></td>
<td>20%</td>
<td>50%</td>
<td>5%</td>
<td>65%</td>
<td></td>
<td>30%</td>
</tr>
<tr>
<td>Tao</td>
<td>45%</td>
<td></td>
<td>5%</td>
<td></td>
<td>50%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yunbo</td>
<td></td>
<td>15%</td>
<td></td>
<td></td>
<td></td>
<td>85%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Chapter 7

System Implementation

The complete system consists of slave nodes, containing pyroelectric sensors and Fresnel lens arrays, a master node, and a host computer. We use the combination of TRF6901 and MSP430149 as the computation and communication platform. Each slave node employs an amplifier board that amplifies eight channel signals in the frequency band 1~10 Hz. Each MSP43019 contains an 8-channel, 12-bit A/D converter, whose sampling rate is set as 19.5 kHz, conversion rate as 96 kHz, with 5 MHz internal ADC clock. One data package for an event includes 48 bits, i.e. 6 bytes. It means that the sampling rate of wireless sensor system, from 10 to 80 Hz in terms of events, is virtually determined by the embedded computation time.

The RF data rate itself is set as 32 kbps, despite its maximum of 76 kbps, for the communication stability. With the same consideration, the UART serial port transmission rate between master and host is set as 57.4 kbps, less than its maximum of 69.4 kbps.

Table 7.1 shows the feature comparison between our computation & communication platform [100] and the well-known MOTE processor/radio module [101]. Our platform has higher computation speed, ADC resolution, and data transmission rate at the expense of higher power consumption. No operating system, such as TinyOS in MOTE, is necessary for our application in the current stage, lowering

125
Table 7.1: Comparison of DISP electronic platform with Mote

<table>
<thead>
<tr>
<th></th>
<th>Duke DISP</th>
<th>Berkeley Mote</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MSP430149+RF9601 (TI)</td>
<td>Atmega128L+NA (Atmel)</td>
</tr>
<tr>
<td>MIPS</td>
<td>8</td>
<td>4</td>
</tr>
<tr>
<td>Data Width (bits)</td>
<td>16</td>
<td>8</td>
</tr>
<tr>
<td>Flash Memory (K)</td>
<td>60</td>
<td>128</td>
</tr>
<tr>
<td>Power (active)</td>
<td>0.28mA/M + 32/18mA</td>
<td>2mA/M + 27/10mA</td>
</tr>
<tr>
<td>(standby)</td>
<td>1.6µA + 0.6µA</td>
<td>15µA + 1µA</td>
</tr>
<tr>
<td>ADC</td>
<td>12bits/8Ch/2.5V/300Ks</td>
<td>10bits/8ch/3.0V/NA</td>
</tr>
<tr>
<td>RF Data Rate</td>
<td>76</td>
<td>50</td>
</tr>
<tr>
<td>TX power (dBm)</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>RX sensitivity(dBm)</td>
<td>-99</td>
<td>-98</td>
</tr>
<tr>
<td>OS</td>
<td>NA</td>
<td>TinyOS</td>
</tr>
</tbody>
</table>

the requirement on flash memory.

The TRF6901 ISM-band RF transceiver is used to establish multichannel, bidirectional wireless communication, link between 888 MHz and 928 MHz, in programmable data transmission rates from 19 to 76 Kbps. The paired baseband processor MSP430F149 FLASH micro-controller comes programmed with baseband communication routines to implement an RS-232 link protocol. As high level users, we have to divide the operational frequency band into many separate channels, each for one sever node. The base channel width is 370.7 kHz, and the bandwidth of carried signal may be up to 1 MHz, that is, four times base channel width; we hence divide the usable base channel number of 105 by 4, and obtain 26 channels without mutual interference.

We employed the master/slave communication mode. Once the master/slave relationship is established, the direction of control is always from the master to the slave(s). Fig. 7.1 shows the data packet structure, including two bytes of FF as “header”, two bytes of 10 as “tail”, and one byte for node ID and another byte for event index. Tab. 7.2 is the summary of the communication interface for slave,
master and host.

The whole computation distribution schematics is shown in Fig. 7.2. Each slave node samples the sensor response signals and convert them into event indexes by band-pass filtering, threshold testing, and low-pass filtering, as shown in Fig. 3.4. After compressing each event index into one byte, the slave node broadcasts the data packages via its unique wireless channel, which are picked up by the master node. The master node synchronizes the communications with all the nodes, removes the false alarms, and frames a new composite event message for tracking. The master sends the messages to the host, which translates those alarms into local angular displacements or digital features and updates the dynamics states or identities of objects with maximum likelihood.
### Table 7.2: Communication interface for slave, master, and host

<table>
<thead>
<tr>
<th>Node</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Slave Node</strong></td>
<td>初始化 RF（包长度，子带宽分配，波特率）</td>
</tr>
<tr>
<td></td>
<td>while(1)</td>
</tr>
<tr>
<td></td>
<td>采样和处理()</td>
</tr>
<tr>
<td></td>
<td>Send RF（子带宽分配，&amp;包，包长度）</td>
</tr>
<tr>
<td></td>
<td>end</td>
</tr>
<tr>
<td><strong>Master Node</strong></td>
<td>同步 RF（包长度，所有带宽分配，波特率）</td>
</tr>
<tr>
<td></td>
<td>while(1)</td>
</tr>
<tr>
<td></td>
<td>for i=1:数量 of 子带宽广播</td>
</tr>
<tr>
<td></td>
<td>接收 RF（子带宽(i)，&amp;包，包长度）</td>
</tr>
<tr>
<td></td>
<td>错误拒绝事件平滑()</td>
</tr>
<tr>
<td></td>
<td>Send Serial（节点 ID，&amp;包，包长度，波特率）</td>
</tr>
<tr>
<td></td>
<td>end</td>
</tr>
<tr>
<td><strong>Host</strong></td>
<td>初始化 Serial（com#，缓冲区长度，波特率）</td>
</tr>
<tr>
<td></td>
<td>while(1)</td>
</tr>
<tr>
<td></td>
<td>找包头()</td>
</tr>
<tr>
<td></td>
<td>读取数据()</td>
</tr>
<tr>
<td></td>
<td>数据融合  目标跟踪()</td>
</tr>
<tr>
<td></td>
<td>end</td>
</tr>
</tbody>
</table>
Figure 7.2: Computation distribution among slave, master, and host.
Chapter 8
Conclusions

This thesis presented a framework and its prototype implementation of a wireless distributed pyroelectric sensor system for multiple human tracking and identification based on TI’s micro-controller and RF transceiver combination of MSP430149 and TRF6901. The prototype system is made up of three kinds of modules: task synthesis hosts, communication synchronization masters, and data collection slaves.

The high variability, volatility of human motion and thermal biometrics necessitate a spectrum of hardware and software techniques that can extract information of interest from human IR radiation under various experimental and theoretical constraints. The concept of distributing the computation to multiple low complexity nodes with local computation and communication capabilities alleviates high complexity requirements upon central processors and data storage. The use of the motion detectors helps maintain the low requirements on data throughput, computational consumption, and communication bandwidth. The characteristics of the pyroelectric sensor give the system the ability to operate under all-illumination environments and the capability to capture human thermal biometrics. The application of the geometric modulation of sensor visibility by Fresnel lens arrays improves the sensing resolution and efficiency, increases the signal discrimination among humans, and facilitates the data-to-object association in multiple object tracking, through
the trade-off between local and global visibilities of sensors.

The techniques proposed and developed in this thesis have addressed problems from visibility modulation and sensor configuration/deployment to event detection and feature capture to embedded computation and wireless communication to data association/fusion and tracking/identification synthesis. The embodiment of the prototype sensor system integrates the knowledge and skills in optics, electronics, and mechanics. The performances of the prototype system have been tested and evaluated by real-time experiments. The results of this thesis herald a rise in the near future of the pyroelectric sensor to a mainstream sensing instrument, beside its video and audio counterparts, offering a new dimension, for extensive applications of human-machine interfaces and human biometrics.

8.1 Thesis Contributions

Chapter 2 presented a mathematical description of the distributed geometric pyroelectric sensors, and a general statement for multiple human tracking and identification problems. A four order band-pass impulse response model of the pyroelectric sensor was estimated from its step response data. Based on the measured sensor visibilities [2] of the dual element pyroelectric sensor, a series of visibility coding schemes were designed, for either tracking or identification of humans. Different sensor modules were made according to the designed visibilities modulated by masked Fresnel lens arrays. The tracking of a single human is usually posed as a dynamic estimation of partially observable Markov process, and in most cases can be solved by Bayesian recursive filtering schemes. The multiple human tracking demands the process of data-to-object association before being decomposed into a set of single object tracking problems. The whole object identification problem includes data
learning and hypothesis testing, given the form of feature representation. This thesis investigated two approaches to data learning: the supervised linear regression and the unsupervised expectation-maximization recursion.

Chapter 3 defined the concept of event and feature from the pyroelectric signal perspective. Signal processing techniques such as Kalman filtering, FFT, and band-pass sine filtering are proposed to detect the occurrence of events embedded in the sensory data. A Markov chain Monte Carlo simulation engined clustering method, based on a Gaussian mixture model, was developed to model noises added to the sensor measurements, to allow the event acceptance/rejection thresholds to be chosen accordingly. Event sequences are further smoothed by a HMM filter and validated by the prior knowledge of the geometry of sensor configuration and deployment to reduce the event mis-registration. This thesis developed techniques to extract two kinds of features from pyroelectric sensor (array) signals in real time: the digital feature and the analog feature. The digital feature exploits the statistical property of the binary event index sequence generated by a sensor array. The analog feature highlights the spectral distinctions among sensor response data of human objects. Examples of both digital and analog features were given to support the choice of such feature selections.

Chapter 4 presented a single human tracking system based on radial pyroelectric sensor modules. Three Bayesian tracking techniques of Kalman filters, HMM filters, and Gaussian particle filters were introduced and tested for the system. Their performances and computation costs were compared. Three event detection techniques of Kalman filtering, FFT, and band-pass sine filtering were applied to the experimental sensory data. From the experimental results, band-pass sine filtering and Kalman tracking, based on a grid approximation, were finally chosen for their simplicity and effectiveness.
Chapter 5 presented a multiple human tracking system based on two-column radial pyroelectric sensor modules. The system consists of one host, one master, four tracking slaves. The whole tracking scheme comprises event detection, object localization, and motion filtering & predication. The local and global visibilities of the sensor system were modulated to validate effective event detections and to facilitate the process of data-object-association. An EM-Bayesian tracking strategy featured with a process of data-object-association was proposed to help localize multiple humans. A more general tracking and identification integrated framework was also proposed, in which the data processing and inference were based upon levels of signals, events, features, objects, and templates.

Chapter 6 presented real-time walker recognition systems by using two kinds of sensor modules. The concept of walker verification and walker identification, as well as two modalities of path-dependent and path-independent in walker recognition, were given. The recognition systems are either based on event signal spectra (analog features) or event index sequences (digital features) in path-dependent or path-independent ways. The analog feature based recognition system responses faster and can reject unregistered objects but is sensitive to the speeds and walking paths of walkers, and requires a longer training phase and wider data transmission bandwidth. The digital feature based recognition system is less sensitive to the object speeds and walking paths, advantages in a shorter training phase and lower data transmission bandwidth but lacks the ability to reject unregistered objects. The experimental results also showed that the recognition performance deteriorated with the increase of object number. The proposed sensor systems could work as a low-security biometric system for a small group of objects and offer one new dimension for the applications of human-machine interfaces and human biometrics.

Chapter 7 described the setup and implementation details of the computation
and communication platform, the combination of TRF6901 and MSP 430149. The computation load distribution among slave, master, and host were also described in the context of real-time implementation.

In summary, the explicit advantages of human tracking and identification with DSNs include better spatial coverage, robustness, survivability, and modularity. The flexibility and leverage of our geometric sensor paradigm lie in its modularity of multiplex visibilities, which allows improvement in sensing accuracy and feature extraction efficacy by spreading and overlapping of FOVs. We expect that, through our study, different visibility patterns modulated by Fresnel lens can be better understood in their function as spatial filters and be able to capture human motion features with a high sensing efficiency. Our research proposes various prospective applications of pyroelectric sensors for human-machine interfaces and human biometrics.

8.2 Future Work

In fact, what we have done with pyroelectric sensors is just one tiny step onto a vast landscape. The human tracking and identification in pyroelectric terms is far away from being a technology where all the possibilities have already been explored. During our experiments in human tracking, how to reduce the errors in optical geometry modeling and system alignment poses itself as an important issue for system performance improvement. To develop some simple and effective self-calibration algorithms to compensate for those errors turns out to be an arresting task in the future work. Besides, despite the fact that we have demonstrated the tremendous capability of distributed pyroelectric sensors in multiple human tracking and identification, with various visibility modulation schemes, a number of fundamental
questions are indeed left unanswered.

1) What is the optimal visibility design for pyroelectric human tracking?

2) What is the optimal visibility design for pyroelectric human identification?

3) What is the highest resolution for human motion feature capture in the pyro-electricity term?

4) How far can a wireless distributed pyroelectric sensor system go in tracking and identifying humans?

Here, some speculations upon the future work in those directions are listed.

1) Visibility Optimization

Different visibility layouts will create pyroelectric sensor systems with different capabilities: one good for motion detection, another for feature extraction. As illustrated in Appendix A, there are two main types of geometric sensor systems, those with an embedded object space and those with an external object space. In this thesis, the first type of sensor systems has been mainly investigated. This type of sensor systems can offer a high sensing efficiency but lacks the ability to implement complicated visibility modulation. By contrast, the geometric sensor systems with an external object space can provide more possible patterns for visibility modulation at the expense of lower sensor efficiency. For the tracking purposes, more visibility patterns of the second type of sensor systems should be investigated, in an effort to reduce the computation complexity brought on by tracking a larger number of humans.

To the human identification end, two types of visibility modulation can be further tested, namely the periodic and the random. In capturing motions of
the human body along the vertical direction, a periodic visibility modulation of a sensor array functions as spatial filters with a specific passing band. A (pseudo-)random visibility modulation of a sensor array yields a set of spatial filters with orthogonal passing bands. The study of these two types visibility modulation would improve human identification rates.

2) **Node-Centric Computation**

The sensor systems we proposed are still under centralized schemes. All the high level computation, including data fusion, tracking, and identification, are processed on the host module. The master module only synchronizes the communications and validates data, without any work on data fusion and abstraction. In a complicated sensor network, more computation and data fusion/abstraction have to be tackled by those master nodes. Simpler, distributive algorithms should be developed. Those proposed algorithms for distributed processing in clustering and tracking [102] can cast an extra light upon the development of the node-centric DSNs.

3) **High Density Sensor Space**

The prototype tracking system presented in this thesis only employs four sensor nodes to demonstrate the advantages of geometric sensors in tracking multiple objects. It of course can be extended to more nodes with similar two-column visibility patterns, to achieve higher tracking resolution and the capability of tracking more human objects. A more general case of sensor deployment is illustrated in Fig. 8.1, in which the two neighboring nodes form local visibility overlap regions. In a high density sensor space, it is expected that not only the human displacements but more nuanced human motions, such as postures and limb movements, can be captured as well, so the whole sensor system can sever
Figure 8.1: A general case of sensor deployment for multiple object tracking.

as a high resolution human-machine interface.

4) Advanced Integration of Tracking and Identification

A fully functional real-time multiple human tracking and identification system demands short testing event sequences for fast walker identification and multiple object recognition at the same time. The identification slave node we used in experiments contains only four pyroelectric sensors. The possible solutions include increasing the number of sensors in the identification node, optimization of the visibility coding, and using multiple identification slave nodes. Other methods such as separating the visibilities of different identification nodes and choosing the $K$ highest likelihood feature models for a testing event sequence generated by $K$ walkers should be studied.

Fig. 8.2 shows the block diagram of an integrated human tracking and identi-
Figure 8.2: Block diagram of an integrated human tracking and identification system.

The block diagram illustrates the key components of the system, which includes event detection, event acquisition, visibility coding scheme, sensor geometry, target dynamics, trajectory smoothing, motion inference, membership estimation, and target identification. The system also involves feature matching, feature models, decision rules, feature-event clusters, and calibration with visibility coding scheme, feature-event clusters, and feature models.

The key issue of the system is how to build up a motion-event codebook and feature-event cluster from the calibration experimental results.
Appendix A

Segmentation of the Plane by Geometric Sensors

A.1 Geometric Sensor System Models

A geometric sensor system consists a detection space, a reference structure and an object space, which are made up of sensors, obscuants and radiation object fields respectively. The detection space can be external to the object space shown in Fig. A.1, just like in Coded Aperture Imaging (CAI) systems, or embedded to the object space shown in Fig. A.2. Correspondingly, we will have the visible objects which can be detected by all the sensors or embedded objects which only could be observed by some of the sensors in the systems.

In this section, we mainly discuss these two classes of geometric sensor (GS) systems, noted as type I and type II GS systems for the simplicity, with 2D radiation object field.

A.1.1 System Description

The image formation in a GS system usually consists two steps:

(1) the modulation of the incident radiation by the proper configured reference structure with elements of absorbing (opaque) obscurants;
(2) the implementation of computer algorithm to reconstruct the radiation objects from the projection on the detection space.

During the imaging process, the position-sensitive sensors can detect the arrival of photons and produce impulse response signals if the number of arriving photons surpasses a threshold. Each obscurant is made of a piece of absorbing (opaque) material and thus the photons are only allowed to reach those point sensors through
the space without occlusion by any obscurant.

As a result, the projection on the detection space from a radiation object field will be expressed as

$$M(r) = \int_{\Omega} V(r) R(r, r') \, dr'$$

(A.1)

where $R(r, r')$ is the radiation field; $V(r)$ is the visibility function describing the distribution of the obscurants; $\Omega$ is the region with distribution of the radiation field; $M(r)$ is the measurement of the radiation field, i.e. its projection on the detection space.

Eq. (A.1) gives us a complicated nonlinear mapping between the radiation field and the detection space. To simplify the mapping, we can discretize all these functions. Correspondingly, the radiation field will be segmented into many cells by the intersections of the occlusions formed by different sensor and obscurant combinations. Each cell is featured with a binary string of “0” and “1”, namely called a signature, to describe its detectability to sensors.

The discretized GS system can be expressed as

$$M = VR,$$

(A.2)

where $R$ is the discretized radiation field; $V$ is the signature matrix; $M$ is a discrete measurement.

It can be seen that the number of those segments with different signatures is associated with the resolution of the imaging system. Therefore, how many sensors and obscurants we should employ in a GS system to achieve an expected imaging resolution, is the first as well as the central problem to solve for a GS system design.
A.1.2 Notation and Definitions

We formally introduce here the abstract set up for a general discretized RST system:

\[ S = \{ s_1, ..., s_m \} \]: Set of sensors

\[ O = \{ o_1, ..., o_n \} \]: Set of obscurants

\[ m = |S| \]: Number of sensors

\[ n = |O| \]: Number of obscurants

\[ c_{ij} \]: Occlusion cone formed by \( s_i \) and \( o_j \)

\[ C = \{ c_{11}, ..., c_{mn} \} \]: Set of the occlusion cones

\[ A(\mathcal{C}) \]: Arrangement of \( \mathcal{C} \)

\[ f \]: Face of \( A(\mathcal{C}) \)

\[ \chi(f) \in \{ 0, 1 \}^m \]: Signatures of \( f \)

\[ \Pi(S, O) \]: Set of signatures

\[ \pi(S, O) = |\Pi(S, O)| \]: Number of unique signatures

\[ \pi(m, n) = \max \pi(S, O) \]: Maximum number of unique signatures

Note that \( \pi(m, n) \) is also determined by the geometric configuration of \( S \) and \( O \).

A.1.3 Problem Statement

Now, the whole problem can be described as: If \( \exists S |S| = m \) and \( \exists O |O| = n \), then what are upper and lower bounds of \( \pi(m, n) \) in object space?

However, to obtain a solution of the problem for a GS system with an arbitrary geometric configuration is quite difficult, if not impossible. Therefore, as an initial effort to approach the problem, we can try to solve some sub-problems formulated with simple geometric configurations without losing the intrinsic characteristics of the general problem.
Problem 1: what is upper and lower bounds of \( \pi(m, n) \) on a line for a GS system with sensors and obscurants uniformly distributed along parallel lines?

Problem 2: what is upper and lower bounds of \( \pi(m, n) \) on a plane for a GS system with sensors and obscurants uniformly distributed on parallel planes?

Problem 3: what is upper and lower bounds of \( \pi(m, n) \) on a plane for the same GS system in problem 2 but with an employment of multi-layer reference structure?

Problem 4: what is upper and lower bounds of \( \pi(m, n) \) on a plane for a GS system with sensors and obscurants uniformly distributed on lines?

Problem 5: what is upper and lower bounds of \( \pi(m, n) \) for the same GS system in problem 4 but with an employment of multi-layer reference structure?

We can see that the problem 1, 2 and 3 describe some type I GS systems while problem 4 and 5 formulate some type II GS systems.

Next we will give mathematical analyses and solutions of those specific problems.

### A.2 Mathematical Analysis

#### A.2.1 Type I 2-D GS Systems

In this section, we study the 2-D type I GS system shown in Fig. A.1 via problem 1, 2 and 3. The upper and lower bounds of the maximum number of unique signatures of the radiation field in those specific type I GS systems are obtained by using a statistical method.

Problem 1

Problem 1 formulates an 1-D GS system shown in Fig. A.3. In this GS system, the \( m \) point sensors are uniformly placed along a line with an interval of \( d \), while the \( n \) obscurants in size of \( r \) are placed on some possible positions which are also
uniformly placed along a line with an interval of $R$.

The resultant segments by the occlusions on the object line will also be uniformly distributed in the size of $l$ with an interval of $L$. From some basic geometry, we have

$$\frac{r}{l} = \frac{R}{L} = \frac{D}{H}, \quad (A.3)$$

where $D$ and $H$ are the distrances between detection plane, reference structure and object plane respectively.

![Figure A.3: The equivalent 1-D GS system.](image)

**Proposition 1.1**: An upper bound of the $\pi(m, n)$ in problem 1 is $\Omega(mn)$, where $y = \Omega(x)$ represents $y \leq Cx$.

**Proof**: For $m$ sensors and $n$ obscurants, we have $mn$ occlusions and $2mn$ boundary lines for those occlusions. Since each boundary line will have one intersection with the object line, then we have $2mn$ intersected points and $2mn + 1$ segments on the object line. Since each of the $2mn + 1$ segments on the object line has a signature, we will obtain that $\pi(m, n) < \Omega(mn)$. ♠

**Proposition 1.2**: A lower bound of the $\pi(m, n)$ in problem 1 is $O(\frac{mn}{m})$, where $y = O(x)$ represents $y \geq Cx$. 

144
Figure A.4: The sensor-obscurant pattern to form a new signature.

Proof: Fig. A.4 illustrates a sensor-obscurant pattern to achieve a new signature. It can be seen that in each pattern to realize a new signature we can use less-than-$m$ obscurants which block a segment’s visibility to some sensors. Thus, we can obtain $N$ different signatures with less-than-$Nm$ obscurants. That is $\pi(m, n) > O\left(\frac{n}{m}\right)$. In order to obtain all the $2^m$ signatures, we need less-than-$m2^m$ obscurants.

However, this kind of realization of new signatures is unsatisfactory, since the obscurant efficiency is too poor. To improve the efficiency of obscurant use, we can employ the pattern which only has two different obscurants with its neighboring pattern, shown in Fig. A.5.

In such a GS system, the $n$ obscurants are selected from $N + m - 1$ possible positions which are uniformly placed along a line. The probability to place an obscurant on each of those positions is $p = \frac{n}{N+m-1}$. Then $N$ segments will be formed on the object line noted as $\{t_1, \ldots, t_N\}$. Obviously, any pair of the $i^{th}$ segment $t_i$ and $k^{th}$ sensor $s_k$ could be blocked by one possible obscurant.

Proposition 1.3: A lower bound of the $\pi(m, n)$ in problem 1 is $O\left(\frac{mn}{\log(mn)}\right)$.

Proof: For the GS system in Fig. A.5, the probability of two segments having a same bit of their signatures is
where the event $x_{ijk}$ is defined as that the $k^{th}$ bit of the signatures of $t_i$ and $t_j$ are the same; the event $x_{ijk}^1$ and $x_{ijk}^0$ are defined as that both $t_i$ and $t_j$ are blocked and neither of them are blocked from $s_k$ respectively; $p = \frac{n}{N+m-1}$ is the probability to choose a specific obscurant.

For two segments $t_i$ and $t_j$, one obscurant $o_l$ could block at most two source-sensor pairs. Therefore, for the $m$ sensors, at most half events of $x_{ijk}$, where $k = 1, ..., m$, are independent. Thus, the probability of two segments $t_i$ and $t_j$ having a same signature is

$$\Pr(x_{ij}) = \Pr(x_{ijk}^1) + \Pr(x_{ijk}^0)$$

$$= p^2 + (1 - p)^2 = 1 - 2p,$$  \hspace{1cm} (A.4)

$$\Pr(x_{ij}) = \Pr(\bigcup_{k=1}^{m} x_{ijk}) \leq (1 - 2p)^\frac{m}{2}. \hspace{1cm} (A.5)$$
As follows,

\[
\Pr\left(\bigcup_{i,j} x_{ij}\right) \leq \sum_{i,j} \Pr(x_{ij}) < N^2(1-2p)^{\frac{N}{2}}
\]
\[
\leq N^2e^{-pm} = N^2e^{-\frac{mn}{N+m-1}} < 1
\]

(A.6)

where the last inequality holds when \( N \leq C\frac{mn}{\log(mn)} \). Thus

\[
\Pr[\pi(m, n) \geq N] = 1 - \Pr(\bigcup_{i,j} x_{ij}) > 0,
\]  

(A.7)

for \( N = C\frac{mn}{\log(mn)} \). ♠

In summary, the solution of the problem 1 is

\[
\Omega(mn) > \pi(m, n) > O\left(\frac{mn}{\log(mn)}\right).
\]  

(A.8)

**Problem 2**

Problem 2 formulates a 2-D GS system of type I GS system. Both sensors and the possible positions of obscurants are uniformly placed on two parallel planes. In the similar way, we can extend the same results for problem 1 to the 2-D model.

**Proposition 2.1:** An upper bound of the \( \pi(m, n) \) in problem 2 is \( \Omega(mn) \).

**Proof:** For \( m \) sensors and \( n \) obscurants, if the sizes of obscurants are small enough such that the segments on the object plane formed by the similar patterns in Problem 1 do not overlap each other, the total number of the segments will be \( mn \). That is \( \pi(m, n) \leq O(mn) \). When the segments overlap each other and form new segments, since each segments only can overlap with less than a constant \( C \) other segments, the total number of the segments will be less than \((C+1)mn\). Thus, the conclusion of \( \pi(m, n) \leq O(mn) \) holds for any case. ♠
Figure A.6: The obscurant pattern to form a new signature in 2-D model.

Proposition 2.2: A lower bound of the $\pi(m,n)$ in problem 2 is $O\left(\frac{mn}{\log(mn)}\right)$.

Proof: The pattern to obtain a new signature in 2-D model is shown in Fig. A.6. When the size of all the obscurants is assumed to be small enough such that any two segments on the object plane with different signatures formed by different sensor-obscurant patterns will not overlap each other, a lower bound of $O\left(\frac{mn}{\log(mn)}\right)$ for $\pi(m,n)$ can be also achieved, where $m = m_x \times m_y$ is the number of sensors. In general cases, the segments overlap with neighboring segments and might form new segments but the lower bound still holds. ♠

That is, the solution of the problem 2 is as the same as that of problem 1

$$\Omega(mn) > \pi(m,n) > O\left(\frac{mn}{\log(mn)}\right).$$

Problem 3

In most application cases, the area of the fully coded view field is limited. Though by using one layer reference structure we can obtain all the possible sig-
Problem 3 presents a multi-layer reference structure GS system shown in Fig.A.7. The possible positions of obscurants are uniformly placed on several parallel planes other than one in Problem 2.

**Figure A.7:** The patterns to form new signatures with multi-layer obscurants.

**Proposition 3.1:** An upper bound of the $\pi(m, n)$ in problem 3 is $\Omega(mn)$.

**Proof:** The proof for Proposition 3.1 is the same as that for Proposition 2.1.

**Proposition 3.2:** A lower bound of the $\pi(m, n)$ in problem 3 is $O\left(\frac{mn}{\log(mn/\kappa)}\right)$.

**Proof:** If the number of reference structure layer is $\kappa$, assuming the signatures created by different layer reference structures are independent, for $m$ sensors and $n$ obscurants we will have

$$\pi(m, n) \geq O\left(\kappa \frac{mn/\kappa}{\log(mn/\kappa)}\right) = O\left(\frac{mn}{\log(mn/\kappa)}\right),$$

(A.10)

which means that by using the multi-layer reference structure, we can increase both the obscurant efficiency and the space efficiency.
As a summary, the solution of the problem 3 is

\[ \Omega(mn) > \pi(m, n) > O \left( \frac{mn}{\log(mn/k)} \right) . \]  

(A.11)

A.2.2 Type II 2-D GS Systems

In this section, based on the previous results, upper and lower bounds of the segments of the radiation field of the type II 2-D GS system shown in Fig. A.2 can be obtained in a similarly statistical way, with a little more effort.

**Problem 4** Problem 4 formulates a simplified type II GS system with embedded object plane in the detection plane. Both the sensors and obscurants are placed on parallel lines.

**Proposition 4.1**: An upper bound of the \( \pi(m, n) \) in problem 4 is \( \Omega(m^2n^2) \).

**Proof**: Given \( m \) sensors and \( n \) obscurants, there will be at most \( \binom{2mn}{2} \) intersected points of the occlusion cones on the object plane. Since each of these intersected points only can be a vertex of four faces in a plane, one upper bound of the number of those faces, which is also an upper bound of the unique signature number, can be obtained as \( \Omega(m^2n^2) \). ♠

In fact, the signatures on an object plane are formed by all the possible combination of the subsets of the \( m \) sensors and \( n \) obscurants, shown in Fig. A.8. The signature distribution on a plane for 10 sensors and 10 obscurants is illustrated in Fig. A.9.

**Proposition 4.2**: A lower bound of the \( \pi(m, n) \) in problem 4 is \( \frac{L \log(L)mn}{\log[L \log(L)mn]} \), where \( m/\log(mn) < L < m \).

**Proof**: As we obtained in Proposition 1.3, for \( m \) sensors and \( n \) obscurants placed on \( N + m - 1 \) possible positions, \( N \) segments with different signatures can
Figure A.8: The patterns to form new signatures with subsets of sensors and obscurants.

Figure A.9: The distribution of signatures for 10 sensor and 10 obscurants.

be obtained on a specific line. However, when \( N > 2^m \), we still just can get \( 2^m \) different signatures. Such a case is called obscurant saturation.

Fig. A.8 shows that different uniformly placed subsets of \( m \) sensors and \( n \) sensors placed on \( N + m - 1 \) possible positions form signatures on different lines on object plane. The number of signatures on those different lines without obscurant saturations will be \( N + 2N + 3N(1 + \frac{1}{2}) + \ldots + LN(1 + \frac{1}{2} + \ldots + \frac{1}{L-1}) \approx NL^2 \log(L) \), where
integer $L$ represents the number of subsets sensors having different sensor numbers without obscurant saturation, and $2^{m/L} \approx N$. The number of the rest signatures will be $2 + 2^2 + ... + 2^{m/L-1} = 2^{m/L} \approx N$ due to the obscurant saturation. Then we can see that the total number of signatures on the plane is nearly $L^2 \log(L) N$.

Meanwhile, since $m < N < mn$, we have

$$\frac{m}{\log(m)} > L = \Theta\left(\frac{m}{\log(N)}\right) > \frac{m}{\log(mn)}.$$  \hspace{1cm} (A.12)

Since the bit number of the signatures on those lines are $m$, $m/2$, ..., $m/L$, we can express the probability of that any two of those $NL^2 \log(L)$ signatures are identical as

$$\Pr(x_{ij}) = \Pr\left(\bigcup_{k=1}^{m} x_{ijk}\right) \leq (1 - 2p)^{\frac{m}{2}}$$  \hspace{1cm} (A.13)

As follows,

$$\Pr\left(\bigcup_{i,j} x_{ij}\right) \leq \sum_{i,j} \Pr(x_{ij}) < [L^2 \log(L) N]^2 (1 - 2p)^{\frac{m}{2}}$$

$$\leq [L^2 \log(L) N]^2 e^{-\frac{mn}{L}}$$

$$= [L^2 \log(L) N]^2 e^{-\frac{mn}{L(N + m - 1)}}$$

$$< 1,$$  \hspace{1cm} (A.14)

where the last inequality holds when $N \leq C \frac{mn}{L \log[L \log(L) mn]}$. That is

$$\Pr\{\pi(m, n) \geq L^2 \log(L) N = C \frac{L \log(L) nm}{\log[L \log(L) mn]}\}$$

$$= 1 - \Pr\left(\bigcup_{i,j} x_{ij}\right) > 0,$$  \hspace{1cm} (A.15)
where \(\frac{m}{\log(m)} > L > \frac{m}{\log(mn)}\). ♠

Therefore, the solution of the problem 4 is

\[
\Omega(m^2n^2) > \pi(m, n) > O \left( \frac{L\log(L)mn}{\log[L\log(L)mn]} \right).
\]  

(A.16)

**Problem 5**

Problem 5 presents a type II 2-D GS system with multi-layer obscurants, as illustrated in Fig. A.10,

![Diagram](image)

**Figure A.10:** The patterns to form new signatures with multi-layer obscurants.

**Proposition 5.1:** An upper bound of the \(\pi(m, n)\) in problem 5 is \(\Omega(m^2n^2)\).

**Proof:** The proof for for Proposition 5.1 is the same as that for Proposition 4.1.

♠

**Proposition 5.2:** A lower bound of the \(\pi(m, n)\) in problem 5 is \(O(\frac{mn}{\log(mn/\kappa)})\).

**Proof:** Similar to the Proposition 3.2, if the signatures obtained by different layer obscurants are independent, we can obtain a lower bound of \(\pi(m, n)\) as

\[
\pi(m, n) > O \left( \frac{L\log(L)mn/\kappa}{\log[L\log(L)mn/\kappa]} \right) = O \left( \frac{L\log(L)mn}{\log[L\log(L)mn/\kappa]} \right).
\]  

(A.17)
where \( m/\log(mn/\kappa) < L < m \). ♠

Therefore, the solution of the problem 5 is

\[ \Omega(m^2n^2) \geq \pi(m, n) > O \left( \frac{L \log(L) mn}{\log[\log(L)mn/\kappa]} \right). \]  \hspace{1cm} (A.18)
Bibliography


157


Biography

Personal Data

Name at Birth: Qi Hao

Date of Birth: December 18, 1973.

Place of Birth: Baoji, Shanxi, People's Republic of China.

Education

Doctor of Philosophy, Duke University, U.S.A., expected 2006

Master of Engineering, Shanghai Jiaotong University, P. R. China, 1997

Bachelor of Engineering, Shanghai Jiaotong University, P. R. China, 1994

Publications

Journal Papers


**Journal Papers Prior to PhD Research**


