INTELLIGENT FALL DETECTION SYSTEM FOR ELDERCARE

A Dissertation

presented to

the Faculty of the Graduate School

at the University of Missouri-Columbia

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

---

by

LIANG LIU

Mihail Popescu, Dissertation Supervisor

DECEMBER 2014

The undersigned, appointed by the dean of the Graduate School, have examined the dissertation entitled

INTELLIGENT FALL DETECTION SYSTEM FOR ELDERCARE

presented by Liang Liu,

a candidate for the degree of doctor of philosophy,

and hereby certify that, in their opinion, it is worthy of acceptance.

MIHAIL POPESCU

MARJORIE SKUBIC

MARILYN RANTZ

TONY X. HAN

DEDICATION

I would like to thank my current and former lab-mates, classmates and friends for making this journey a pleasant one, and for all their help, encouragement and support.

I also extend my deepest appreciation and love to my family. Without their forever love, encouragement and unswerving support, this journey could not have been possible.

ACKNOWLEDGEMENTS

This dissertation could not have been possible without support from my advisor, committee, colleagues, friends and family. First, I would like to extend my deepest gratitude to my advisor, Professor Mihail Popescu for all his patience, support and guidance during the past five years. Mihail brought me into the exciting healthcare environment with an engineer's insight to pursue cutting edge research. He touched me with his passion for research and pushed me forward to excellence.

I would like to thank Professor Majorie Skubic for her caring and help over the years, for her suggestions on my pattern recognition course study, for serving as a member of my committee for both of my master's degree and PhD study, and for letting her warm hearted husband, Dan, change my flat tire.

I am grateful to Professor Marilyn Rantz for her kind support and willingness to involve me in the research project, giving fast feedback to improve my work, and serving on my dissertation committee.

I am indebted to Professor Tony Han for his wise guidance throughout my master's degree study, serving as a committee member for my defense, and his kind suggestions on how to better fit into US study life as an international student.

I would also like to thank Professor James Keller for his knowledgeable guidance throughout my study of computational intelligence, supervised, unsupervised learning, *etc*. I would like to thank Professor Dominic Ho for instructing me on study signal processing courses and for his academic discussions with me.

I am grateful to our Center for Eldercare and Rehabilitation Technology.

Table of Contents

List of Figures

List of Tables

INTELLIGENT FALL DETECTION SYSTEM FOR ELDERCARE

Liang Liu

Dr. Mihail Popescu, Dissertation Supervisor

ABSTRACT

Fall among elders is a main reason to cause accidental death among the population over the age 65 in United States. The fall detection methods have been brought into scene by implemented on different fall monitoring devices.

For the advantages in privacy protection and non-invasive, independent of light, I design the fall detection system based on Doppler radar sensor. This dissertation explores different Doppler radar sensor configurations and positioning in both of the lab and real senior home environment, signal processing and machine learning algorithms.

Firstly, I design the system based on the data collected with three configurations: two floor radars, one ceiling and one wall radars, one ceiling and one floor radars in lab. The performance of the sensor positioning and features are evaluated with classifiers: support vector machine, nearest neighbor, naïve Bayes, hidden Markov model. In the real senior home, I investigate the system by evaluating the detection variances caused by training dataset due to the variable subjects and environment settings. Moreover, I adjust the automatic fall detection system for the actual retired community apartment. I examine different features: Mel-frequency cepstral coefficients (MFCCs), local binary patterns (LBP) and the combined version of features with RELIEF algorithm. I also improve the detection performance with both pre-screener and features selection. I fuse the radar fall detection system with motion sensors. I develop a standalone fall detection system and generate a result to display on a designed webpage.

Chapter 1 Introduction

1.1 Motivation

Falling among the elderly is a dominant reason for accidental death in the population above age 65 [1]. The death rate caused by such falls has risen quickly over the past ten years [2]. Studies have showed that the outcome would be improved if medical intervention could be initiated faster after a fall [3, 4, 55, 69, 70]. So medical outcome improvement is significant if the intervention time can be reduced when the fall is detected automatically and reported to the related nursing staff.

1.2 Research Background

Reports show that 90 percent of adults over the age 65 prefer to stay in their residence as they age [5]. Recently, lots of research has been devoted to different fall detection methods. Generally, fall monitoring devices can be divided into two categories: wearable and non-wearable devices. The wearable device requires to be attached on the subject to report a fall, such as “Push-button” [6], Accelerometers [46, 53-59, 63, 82, 83, 98, 120], Mobil phone [60-62, 79, 91], wearable wireless device [77, 78], wearable camera [93] and others [66, 67, 72, 73, 80, 84-90, 92, 96, 97, 124]. However, wearable devices can malfunction under certain situations, as when the elder loses consciousness and cannot activate the sensor or the elder falls during shower and is not wearing the sensor [7]. So non-wearable devices are introduced to overcome these physical limitations [43]. They cover floor vibration sensors [8], video cameras [9, 49, 51, 52, 64, 65, 107, 109, 115-117], infrared cameras [10, 50, 108], Microsoft Kinect [99-106, 110-

114, 123], smart carpets [11], microphone arrays [12-14, 118, 121], smart bed [95], and sensor networks [26-28, 48, 68, 75, 81, 94, 119]. Non-wearable radar sensors have many advantages over the other monitoring devices in that they are not dependent on light, can sense through structures, are non-invasive, inexpensive, and are easy to deploy in any home environment, and are often used in bathrooms. The ideal design object of a fall detection system is to detect all the falls with the fewest false alarms.

As a supplement to conventional surveillance applications, radar sensors have been widely applied for activities recognition, such as target presence detection, gender classification, individual identification during human walking, hand gesture recognition, face recognition, transport mode classification, gait characterization, and medical applications [15, 18, 19, 24, 25, 31, 32, 38, 39, 41, 47, 71, 122, 125, 126]. For the Doppler effect, the moving velocity is proportional to the frequency shift of a wave radiated by the moving object. Assuming an object is composed of several small parts, the micro-Doppler signature is produced by the movements of those small parts for the moving object. The micro-Doppler signature is an additional modulation of the base Doppler frequency shift. The movement of a human subject includes movements of legs, arm, torso, etc. The different human body parts generate specific radar signatures [15].

There are mainly two feature extraction categories. The methods in the first category are based on the pattern of the spectrogram. As an example, Kim and Lin [16] used six features extracted from a de-noised radar spectrogram to recognize seven human activities: running, walking, walking while holding a stick, crawling, boxing while moving forward, boxing while standing in place, and sitting still. Dura-Bernal et al. [17] segmented the spectrogram into temporal events, established prototypes and categorized the human actions. They collected two datasets: 5-category human transport modes and 7-category human actions. In the first dataset, the subject was approaching the radar and performed walking, inline skating, slow cycling, and fast cycling. In the second dataset, the subject was on the treadmill and performed slow walk, fast walk, slow run, fast run, clapping hands, moving their arms up while calling “help me,” and beckoned with their right arm while calling “come here.” In another category, the methods are derived from sound processing techniques. This is supported by the assumption that human operators have the ability to listen to the Doppler audio output of the surveillance radar and are able to differentiate certain targets [18]. The Mel-frequency cepstral coefficients (MFCCs) feature extraction is a popular method in speech recognition. It has been successfully applied to the radar sensor-based fall detection research conducted by the author as part of an interdisciplinary team devoted to elderly healthcare research which involved GE Global Research, the Sinclair School of Nursing, a local retirement home, The Tiger Place, and the MU Department of Electrical and Computer Engineering [20-23]. The time-frequency features have been extracted for sequential modeling on radar signals for fall detection in [47]. The local binary patterns are a popular method in computer vision to recognize objects [44]. The one-dimensional local binary patterns have also been implemented for signal processing [45].

1.3 Contribution

We first used floor radars in our fall risk assessment project to estimate the fall risk and corresponding interventions from a daily living environment for nine different walk types using gait velocity and stride duration [19, 31, 41]. Inspired by the Doppler signature detection hardware, we proposed the idea of an automatic fall detection system using Doppler signatures with the same kind of radar sensors used in our 2011 studies [20] whereby those preliminary results could be confirmed and expanded upon.

The performance of the proposed detection system was set up for improvement in two ways. One improvement was from the algorithm; and the other improvement was from the experimental settings. For the algorithm side, we proposed a fuzzy fusion method to fuse multiple information sources considering the sensitivity and preference of each classifier to a specific kind of feature. Those multiple information sources included the outputs of three different classifiers for each of the two sensors on the floor in [21]. For the experimental settings side, each local observation from a sensor was biased and insufficient to detect a fall since the radial velocity component of the moving subject plays such a dominant role in radar sensor detection. Those defects came from several sources including the tilt angle, sensor distance to the ground, antenna shield region and the moving direction of the subject with respect to each sensor. For the two-floor sensor systems in [21-22], the detection of a fall across a sensor did not produce results as good as a fall towards or away from a sensor with 10% detection variability. In order to reduce the effects of falling direction to the sensor, we installed a ceiling radar sensor, although this radar sensor was originally designed to be installed on wall for surveillance purposes. We adjusted a wall mounted sensor with a height of 1.27 meters, which is about a normal person's upper body height. In the system [22] located in our motion lab , the detection of a fall observed from the ceiling sensor outperformed by capturing 9% more area under the receiver operating characteristic curve than that captured by the wall mounted sensor with a less than 30% detection variability for the falling direction. When a fall occurs closer to the floor sensor, the ceiling sensor had a slightly 3-4% lower performance than the floor sensor. We observed that the system comprised of one ceiling sensor and one floor sensor achieved the best results. So we deployed this system in our motion lab for testing purposes first and then we deployed the same system at the Tiger Place senior apartments in our local community that features “aging in place” with assisted living services available. Tiger Place, developed by Americare in affiliation with the MU Sinclair School of Nursing, features apartments for independent living with a 24-hour response through a state-of-the-art call system [37, 40].

Besides the above experimental-based test for data collection, I implemented the automatic fall detection system for the real senior home. I investigated different features: MFCC, LBP and the combined version of features with RELIEF algorithm [128]. I improved the fall detection performance from both the pre-screener and features selection on real senior home data. I designed the classifier level fusion method for this research’s fall detection system. I also fused the radar fall detection system with other sensors, such as the motion sensor network in the senior homes [23, 74, 127]. I have developed the standalone fall detection system and generated the corresponding result, displaying it on the project’s designed webpage. Those results are quite promising. This system could help nursing staff to better understand information on the health status for the senior residents. The long-term goal is to provide seniors with a healthier and safer environment for their independent living.

Five journal and conference publications have resulted from this dissertation work with at least one journal paper in preparation. The publications are listed as follows:

- Liu L, Popescu M, Skubic M, Rantz M, Yardibi T & Cuddihy P, "Automatic Fall Detection Based on Doppler Radar Motion," Proceedings, 5th International

Conference on Pervasive Computing Technologies for Healthcare, Dublin, Ireland, May 23-26, 2011.

- Liu L, Popescu M, Rantz M, Skubic M, "Fall Detection using Doppler Radar and Classifier Fusion," Proceedings of EMBS on BHI, Shenzhen, Jan 2-7, 2012.
- Liu L, Popescu M, Rantz M, Skubic M, “Doppler Radar Sensor Positioning in a Fall Detection System”, Proceedings of 34th EMBS, San Diego, August 28-September 1, 2012.
- Liu L, Popescu M, Skubic M, Rantz M, “An Automatic Fall Detection Framework Using Data Fusion of Doppler Radar and Motion Sensor Network”, Proceedings of 36th EMBS, Chicago, August 26- 30, 2014.
- Liu L, Popescu M, Skubic M, Rantz M, Cuddihy P, “Automatic In-room Human Fall Detection Using Doppler Radar Signatures”, The Journal of Ambient Intelligence and Smart Environments (JAISE), (under revision).
- “An Automatic In-home Fall Detection System with Ceiling Radar”, IEEE Transaction on Biomedical Engineering (ITBE), (under preparation)

1.4 Organization

The rest of this dissertation is organized as follows. Chapter 1 describes the Doppler radar sensor system and validates its application in motion monitoring. Chapter 2 describes the fall detection system configurations and data collections in both lab and real senior apartments. Chapter 4 presents the fall detection algorithm and the evaluation criteria for the detection performance. Chapter 5 explains the data fusion for the multiple information sources at a classifier level. Chapter 7 presents the effective detection range around the radar sensor. Chapter 8 shows the improved system as it relates to the real-world data in senior homes with encouraging results.

Chapter 2 Doppler Radar Sensor

2.1 Principle

2.1.1 Range Control Radar

A radar sensor can transmit electromagnetic energy to a target and receives the reflected signal from the target. We can measure the accurate target-related information, such as location and velocity. Unlike other optical-related or infrared sensors, it works for long-range applications under all kinds of light changing and complex conditions. Unlike outside long-range air-to-air or surface-to-air applications, our in-room application of the pulse Doppler radar uses monostatic radar, in which the transmitter and receiver are collocated and share the same antenna [32].

For a pulse Doppler radar, the pulse train of the waveform can be produced by gating a continuous stable frequency source. The pulse repetition interval is T, pulse length is ι, and the carrier frequency is $f_{\circ}$. Then the waveform can be written as

$$u(t)=\sum_{n=-\infty}^{\infty} p(rect(t/\tau)-nT)\cos 2\pi f_0 t\,, \tag{2.1}$$

where $p(\bullet)$ function denotes a pulse, and $rect(\bullet)$ describes a zero centered gating function with a unit width and height.

By applying the Fourier transform to (1), we have

$$\begin{aligned} &U(f)=(\tau/2T)comb_{1/T}(\sin c f\tau) \\ &\otimes(\delta(f-f_0)+\delta(f+f_0)) \end{aligned} \quad . \tag{2.2}$$

Assuming the waveform is narrowband enough, the positive frequency of the spectrum is

$$U_{+}(f) = (\tau / 2T) comb_{1/T}(\sin c f \tau) \otimes \delta(f - f_0) \qquad (2.3)$$

where *comb*(•) function is centered at zero with a line interval at n/T. The convolution moves the center of the spectrum to $f_{\circ}$.

The target echo is modulated twice on transmission and reception. We simplified the modulation by imposing a rectangular function on the beam with width of ρ, the time on target. Thus, the received waveform of the pulse Doppler radar is

$$x(t) = rect(t / \rho) u(t). \qquad (2.4)$$

The transform is

$$X(f) = \rho \sin c f \rho \otimes U(f) \qquad (2.5)$$

where *u* and *U* are defined as (2.1) and (2.2), and we also have $\rho >> T$.

2.1.2 Doppler Effect

For the radial velocity *v*, the received waveform is scaled by $(c+v)/(c-v)$ and *c* is the light speed. We assume $v << c$ and spectrum is narrowband, and the significant spectral energy is around $\pm f_{\circ}$. Then there is an overall spectral shift of $+2vf_{\circ}/c$. In the view of radar, stationary object echoes produce clutter lines at $f_{\circ}$ with intervals *n/T* on spectrum. The moving targets' echoes produce offset lines from the clutter lines. In this way, the consequential relative movement of the targets can be observed. By appropriate filtering of the spectrum, we can see the targets and estimate their Doppler shift and radial velocity.

The radial velocity for a target moving to radar can be calculated with Doppler frequency shift f_D from the transmitted frequency $f_{\circ}$ via

$$v = f_D c / (2 f_0). \qquad (2.6)$$

2.2 Doppler Radar Signature in Activities Monitoring

Assuming that different body parts generate an independent velocity v_i, the Doppler signature for a moving subject is represented by

$$f_D = f_0 \sum_i (2v_i / c) \ , \tag{2.7}$$

where v_i is from each body part, such as limb, leg, torso, etc.

The short time Fourier transform (STFT) is applied on the returned Doppler signal to show the pattern of human activities via

$$STFT(t, f) = \int_{-\infty}^{\infty} x(t+\tau) w(\tau) \exp(-j2\pi f \tau) d\tau \, , \tag{2.8}$$

where $x(t)$ is the received signal and $w(t)$ is the window function.

Fourier transform is a one-to-one mapping between the signal in time domain $x(t)$ and its spectrum in the frequency domain $X(f)$. The spectrum represents the frequency contents in the time domain. For a continuous time signal, the spectrum is transformed by

$$X(f) = \int_{-\infty}^{\infty} x(t) \exp(-j2\pi ft) dt \tag{2.9}$$

In a real system, the continuous time signal is usually digitized as a discrete time signal $x(n)$. We assume that $x(n)$ is a periodic signal with a period N. The discrete Fourier transform (DFT) is defined as

$$X(k) = \sum_{n=0}^{N-1} x(n) \exp(-j2\pi kn / N), k = 0,1,..., N-1 \, . \tag{2.10}$$

To improve the computational efficiency, the fast Fourier transform (FFT) is designed to reduce the complexity of DFT from $O(N^2)$ to $O(NlogN)$.

Although the spectrum demonstrates the frequency contents in the time domain, it lost the time information. By sliding a window over the time and computing the FFT to the signal in the window, the short time Fourier transform can show the frequency content changes over time.

The discrete STFT is represented by

$$STFT(x(n)) = X(n,k) = \sum_{m=0}^{N-1} x(n+m)w(m)\exp(-j2\pi km/N), k = 0,1,...,N-1, \quad (2.11)$$

where $w(m)$ is the window function, n is the time index and k is the frequency index.

In the latter part, the pattern of the returned Doppler signal is shown with STFT. The time is along the x-axis, and frequency contents are distributed along the y-axis. The magnitude of STFT is shown by the intensity of the color map. This is called a spectrogram. On the spectrogram, we can clearly see how the frequency contents change over time.

2.3 System Validation

In order to validate the system, we first illustrated the micro-Doppler motion with the diagram of a moving pendulum in front of the radar sensor. Then two groups of experiments were designed. The first group targeted on swing motion, including cylinder swing and body part swing when the rest of the body part is still. The second group focused on falling motion, such as a mattress falling.

Each group of experiments was followed by an extension to a more complex situation. The signature for human walking was presented as a more complex case for swing experiments. The micro-Doppler signature of human falling was introduced as a complex extension for falling object experiments.

2.3.1 Micro-Doppler Motion for a Pendulum

The pendulum is swinging towards the radar sensor in a vertical plane, shown in Figure 2.1. The length of the pendulum is L. One side of the pendulum is fixed as center O and the other side can move around the O with an angular velocity w. In a polar coordinate system, the distance between a point E on the pendulum to O is x. The linear velocity of E is

$$v_E = wx , \qquad (2.12)$$

At time t, the pendulum is at an angle $\theta = wt$, with respect to the balance position B. The projected velocity at E to the x-axis is

$$v_{E\_x} = v_E \cos\theta = v_E \cos wt . \qquad (2.13)$$

The projected linear velocity on the x-axis is

$$v_E x = wx\cos\theta = wx\cos wt . \qquad (2.14)$$

At point E, the observed Doppler frequency shift by receiver is

$$f_E = f_0 \frac{2v_E x}{c} = f_0 \frac{2wx}{c}\cos\theta = f_0 \frac{2wx}{c}\cos wt . \qquad (2.15)$$

This is a sinusoidal signal. Then the whole micro-Doppler signature of the pendulum is

$$f_D = \sum_{x\in[0,L]} f_0 \frac{2wx}{c}\cos wt . \qquad (2.16)$$

If the pendulum is moving in a circle track in front of radar sensor, then the received micro-Doppler signature will become

$$f_D \sin\alpha = f_D \sin w't = (\sum_{x\in[0,L]} f_0 \frac{2wx}{c}\cos wt)\sin w't . \qquad (2.17)$$

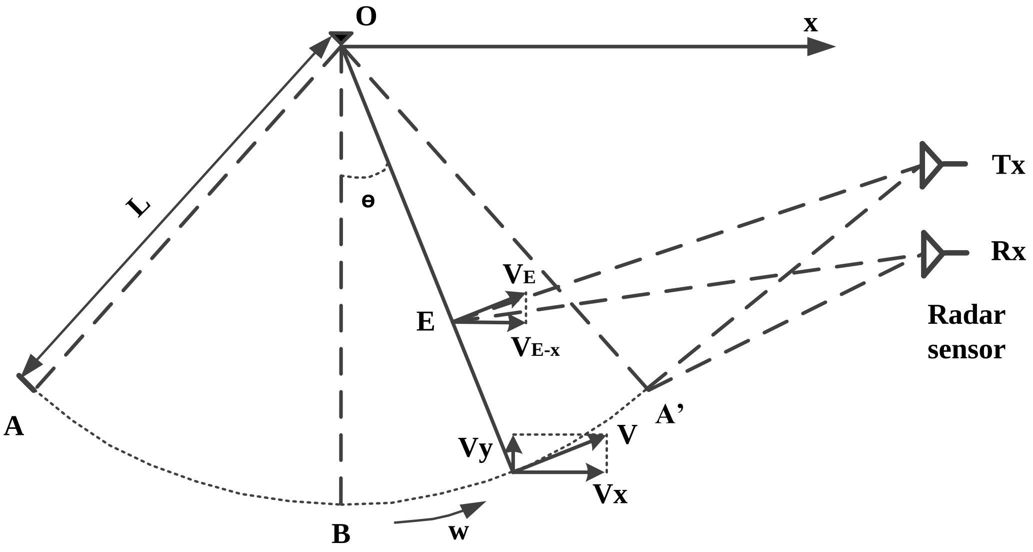


Figure 2.1 The diagram of a swing pendulum in front of radar sensor from a vertical view.

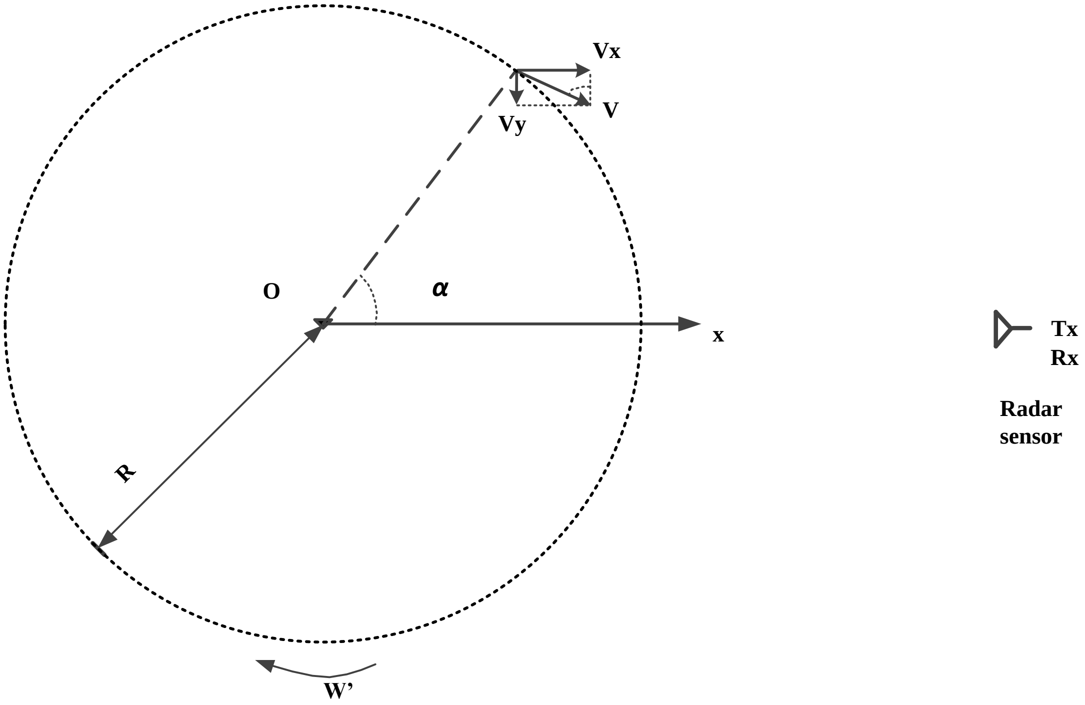


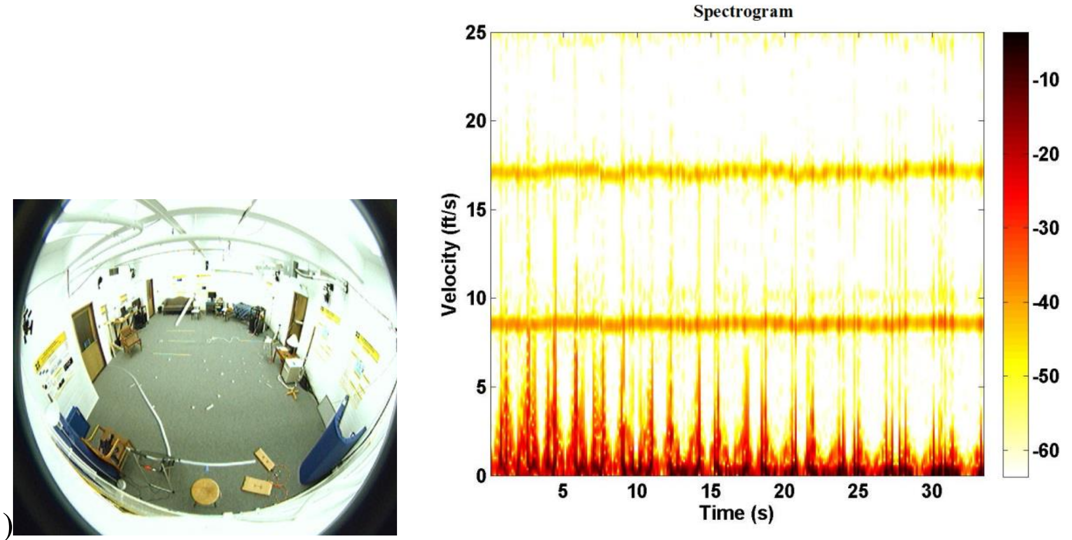
Figure 2.2 The diagram of a moving subject in front of radar sensor from a horizontal view.

2.3.2 Swing Experiments

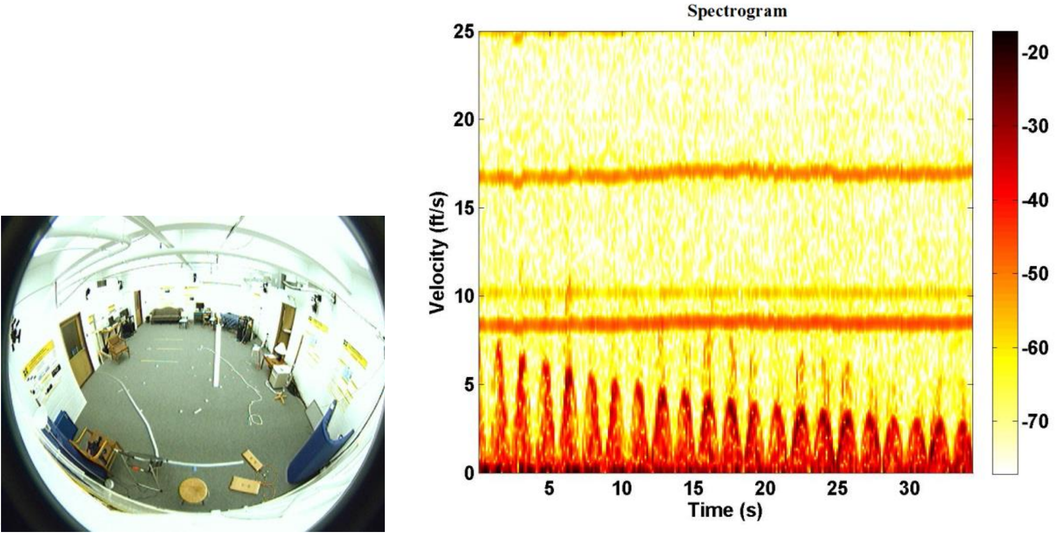
2.3.2.1 Paper Cylinder Swing Experiment

A paper cylinder serves as a pendulum in this experiment. The one end is hung from a rope connected to the ceiling at the center of the room. The other end is dragged to a higher position A in Figure 2.1 and then the cylinder is set free. Thus the cylinder will fall from A to A’. This paper cylinder swings in three ways: across the radar; towards or away from the radar; and in a circular or Ellipse track in front of the radar.

(1)



(2)



(3)

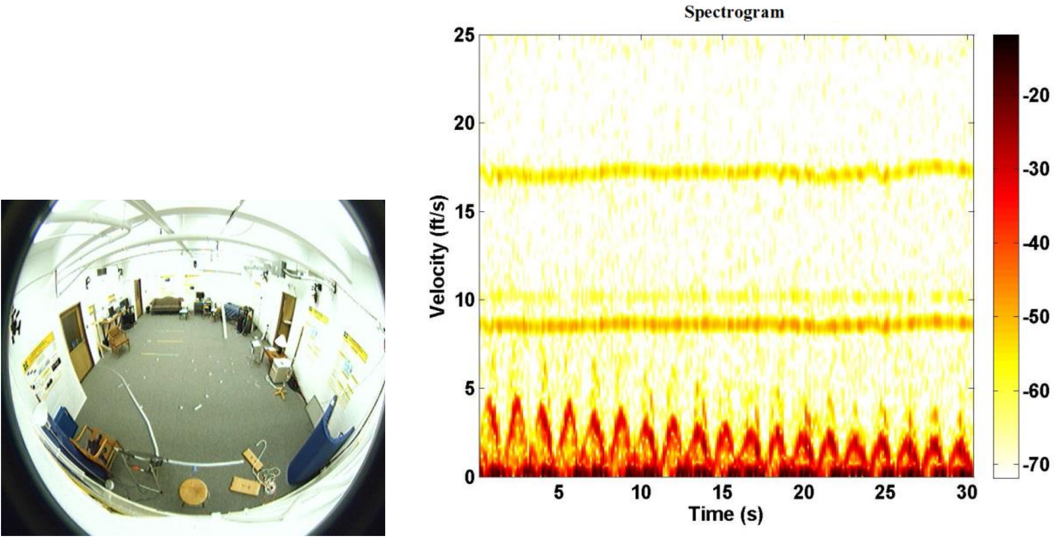


Figure 2.3 Swing experiment with a cylinder which swings (1) across the radar; (2) towards the radar, and (3) in a circular track.

In this experiment, we should also notice that the amplitude of the velocity is attenuating with time and the cylinder stops at the balance position. There is an attenuation coefficient μ. So Equation (2.15) is modified as

$$f_{E\_att} = f_E e^{-\mu x} = f_0 \frac{2wx}{c} e^{-\mu x} \cos wt \; . \tag{2.18}$$

The whole micro-Doppler signature of the pendulum in Equation (2.16) is rewritten as

$$f_{D\_att} = f_D e^{-\mu x} = \sum_{x \in [0,R]} f_0 \frac{2wx}{c} e^{-\mu x} \cos wt \; . \tag{2.19}$$

The receiver is only sensitive to the radial velocity to radar sensor. When the cylinder moves towards or away from radar, the sinusoidal shape with the attenuation can be clearly observed on the spectrogram in Figure 2.3(2). The velocity peak appears when the cylinder end is at the balance point B. At positions A and A’, the velocity is zero.

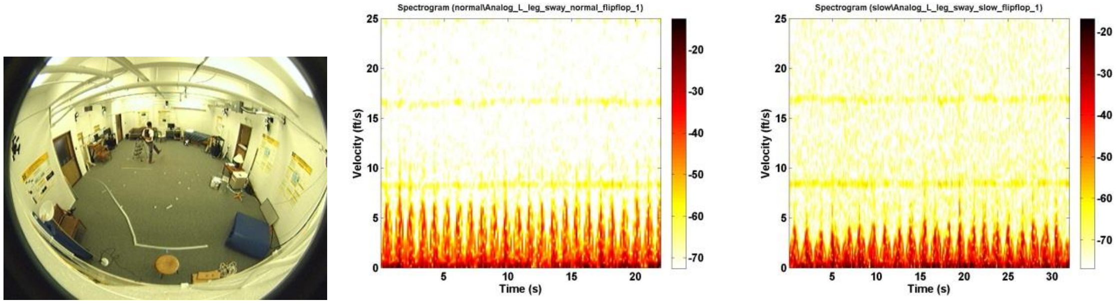
When the cylinder is moving in a circular track, the radial velocity is computed by adjusting Equation (2.17) with

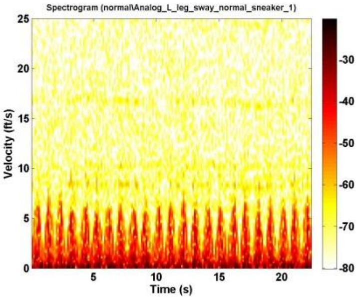
$$f_{D\_att}\sin\alpha = f_D e^{-\mu x}\sin w't = \Big(\sum_{x\in[0,R]} f_0 \frac{2wx}{c} e^{-\mu x}\cos wt\Big)\sin w't \quad . \qquad (2.20)$$

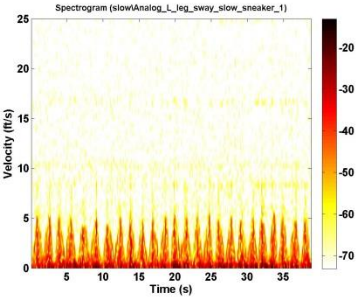
As a result, the sinusoidal shape can still be observed in Figure 2.3(3). But the intensity of each peak is not that evenly distributed as shown in Figure 2.3(2) due to the $\sin\alpha$. The micro-Doppler signature can be clearly observed as a spike when the cylinder is swinging across the radar, as shown in Figure 2.3 (1). The lines at 8 and 16 Hz on the spectrogram are caused by noise. The noise may come from the radar wires or some other electronic equipment in the lab room.

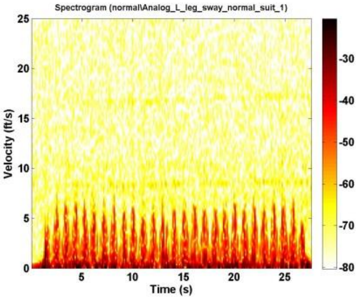
2.3.2.2 Leg/arm Swing Experiment

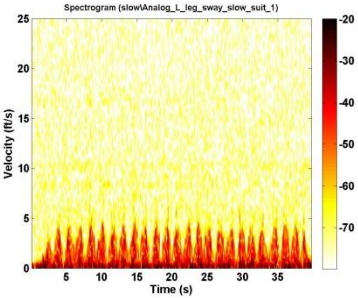
In leg/arm swing experiment, a person is facing toward the radar and moves his leg or arm backward or forward toward the radar. Each swing is repeated twice with normal speed and slow speed. For the leg swing, the subject swings each leg by wearing three different types of shoes: flip-flops, sneakers, and suit shoes. For arm swing, the subject first swings each arm. Then the subject swings both arms in different directions and in the same direction.

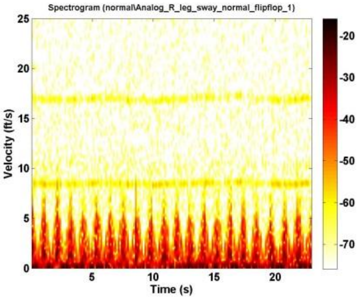


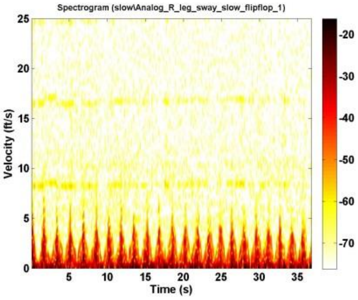

Spectrogram (normal\Analog_L_leg_sway_normal_sneaker_1)
Velocity (ft/s)
0 5 10 15 20 25
Time (s)
5 10 15 20
-30 -40 -50 -60 -70 -80

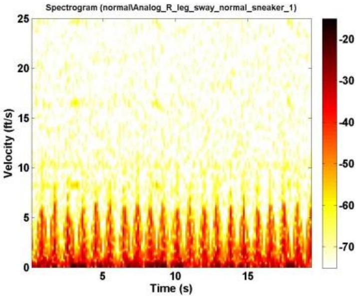

Spectrogram (slow\Analog_L_leg_sway_slow_sneaker_1)
Velocity (ft/s)
0 5 10 15 20 25
Time (s)
5 10 15 20 25 30 35
-20 -30 -40 -50 -60 -70

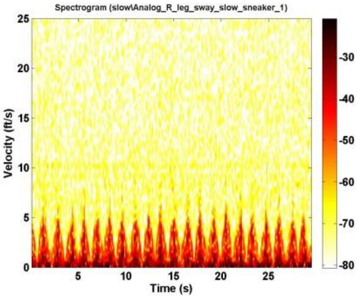

Spectrogram (normal\Analog_L_leg_sway_normal_suit_1)
Velocity (ft/s)
0 5 10 15 20 25
Time (s)
5 10 15 20 25
-30 -40 -50 -60 -70 -80

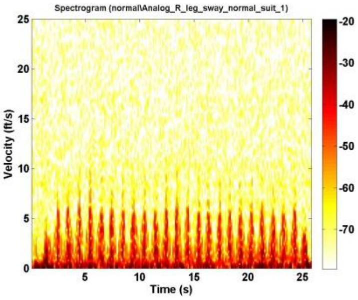

Spectrogram (slow\Analog_L_leg_sway_slow_suit_1)
Velocity (ft/s)
0 5 10 15 20 25
Time (s)
5 10 15 20 25 30 35
-20 -30 -40 -50 -60 -70

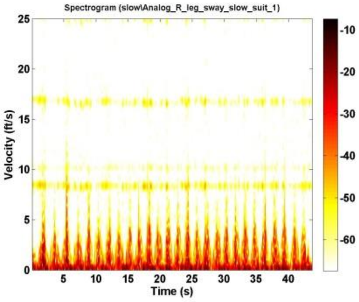

Spectrogram (normal\Analog_R_leg_sway_normal_flipflop_1)
Velocity (ft/s)
0 5 10 15 20 25
Time (s)
5 10 15 20
-20 -30 -40 -50 -60 -70


Spectrogram (slow\Analog_R_leg_sway_slow_flipflop_1)
Velocity (ft/s)
0 5 10 15 20 25
Time (s)
5 10 15 20 25 30 35
-20 -30 -40 -50 -60 -70


Spectrogram (normal\Analog_R_leg_sway_normal_sneaker_1)
Velocity (ft/s)
0 5 10 15 20 25
Time (s)
5 10 15
-20 -30 -40 -50 -60 -70


Spectrogram (slow\Analog_R_leg_sway_slow_sneaker_1)
Velocity (ft/s)
0 5 10 15 20 25
Time (s)
5 10 15 20 25
-30 -40 -50 -60 -70 -80


Spectrogram (normal\Analog_R_leg_sway_normal_suit_1)
Velocity (ft/s)
0 5 10 15 20 25
Time (s)
5 10 15 20 25
-20 -30 -40 -50 -60 -70


Spectrogram (slow\Analog_R_leg_sway_slow_suit_1)
Velocity (ft/s)
0 5 10 15 20 25
Time (s)
5 10 15 20 25 30 35 40
-10 -20 -30 -40 -50 -60

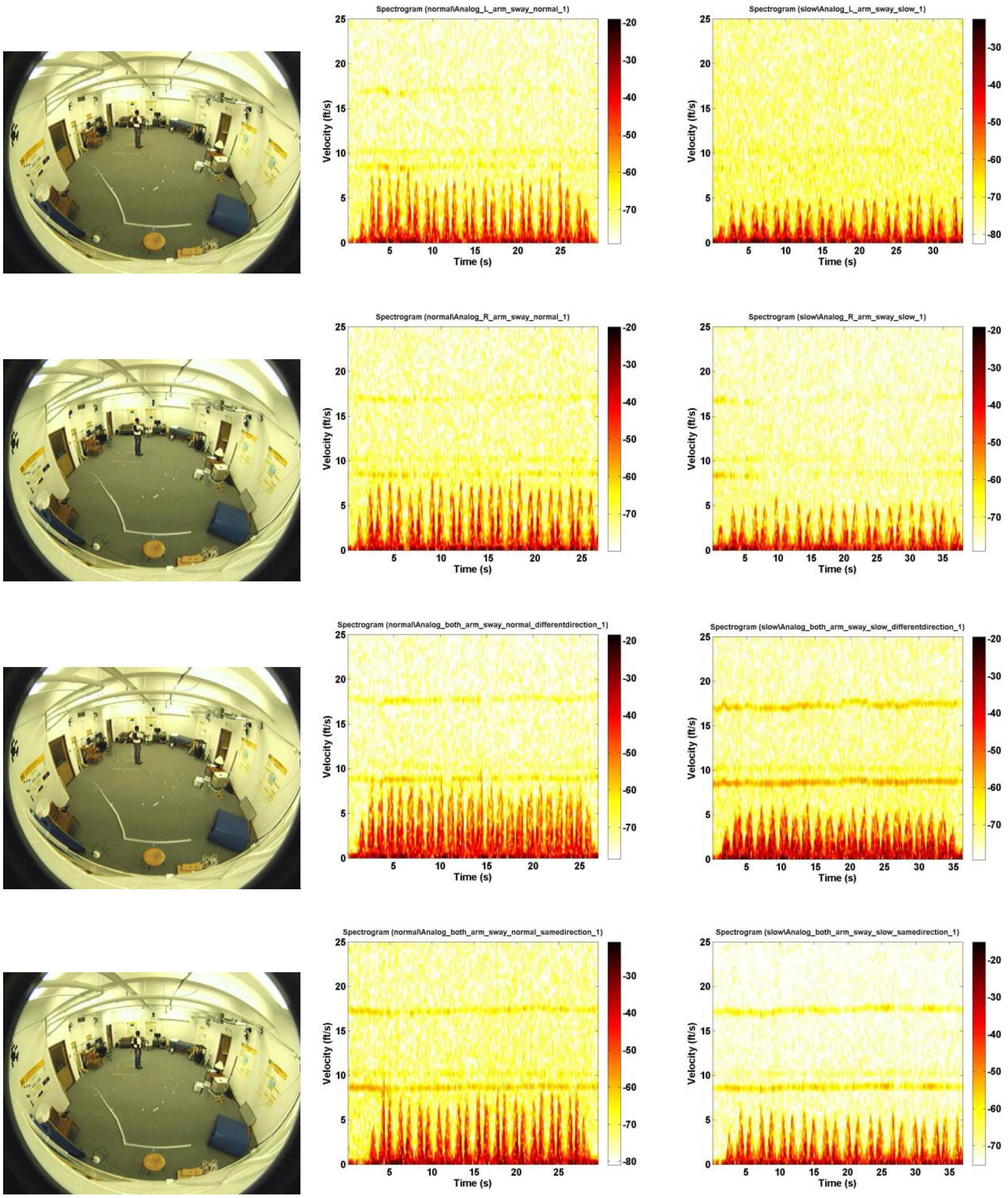


Figure 2.4 Swing experiment with the leg/ arm: The first column shows the subject motion in the experiment setup. The second column shows the normal speed of swing. The third column shows the slow speed of the swing. The spectrogram number and explanation are go from the top down and from left to right. (1) Normal speed, left leg with flip-flop swing; (2) Slow speed, left leg with flip-flop swing; (3) Normal speed, left leg with sneaker swing; (4) Slow speed, left leg with sneaker swing; (5) Normal speed, left leg with suit shoes swing; (6) Slow speed, left leg with suit shoes swing; (7) Normal speed, right leg with flip-flop swing; (8) Slow speed, right leg with flip-flop swing; (9)

Normal speed, right leg with sneaker swing; (10) Slow speed, right leg with sneaker swing; (11) Normal speed, right leg with suit shoes swing; (12) Slow speed, right leg with suit shoes swing; (13) Normal speed left arm swing; (14) Slow speed, left arm swing; (15) Normal speed, right arm swing; (16) Slow speed, right arm swing; (17) Normal speed both arms swinging in different directions; (18) Slow speed, both arm swinging in different direction; (19) Normal speed, both arms swinging in same direction; (20) Slow speed, both arms swinging in same direction.

There are some observations for this experiment. For the same type of swing with different speeds, we can clearly see the velocity difference with the same pattern on the spectrogram in Figure 2.4 (1, 3, 5, 7, 9, 11, 13, 15, 17, 19) **vs** Figure 2.4 (2, 4, 6, 8, 10, 12, 14, 16, 18, 20), respectively. There is no big difference among the signals from leg swing with different types of shoes in Figure 2.4 (1)-(12). For the same types of swing, the left leg/ arm are similar to the right one in Figure 2.4 (13, 14) **vs** Figure 2.4 (15, 16). When a single arm swings, the two consecutive peak distances are different. This is true since the person can move his single arm forward, moreso than in backward direction. When the person moves his two arms in different directions at same time in Figure 2.4 (17-18), summation of both arms contribute to the micro-Doppler signatures. So the distances of any two consecutive peaks are identical. When two arms swing in the same direction in Figure 2.4 (19-20), the distance difference still existed as the swing of one arm, but the signal strength was enhanced.

2.3.3 Micro-Doppler Signature for Human Walk

In Figure 2.5, the subject is walking towards the radar first and then walking away from the radar. At the time of 13th second, the subject is turning around in front of the radar. At the turn-around position, the velocity of the subject is almost zero in the radial direction of the radar. The green dash-dotted line represents the walking start time and black dashed line is the walking end time. The blue circles on the spectrogram represent the steps taken in walking [19].

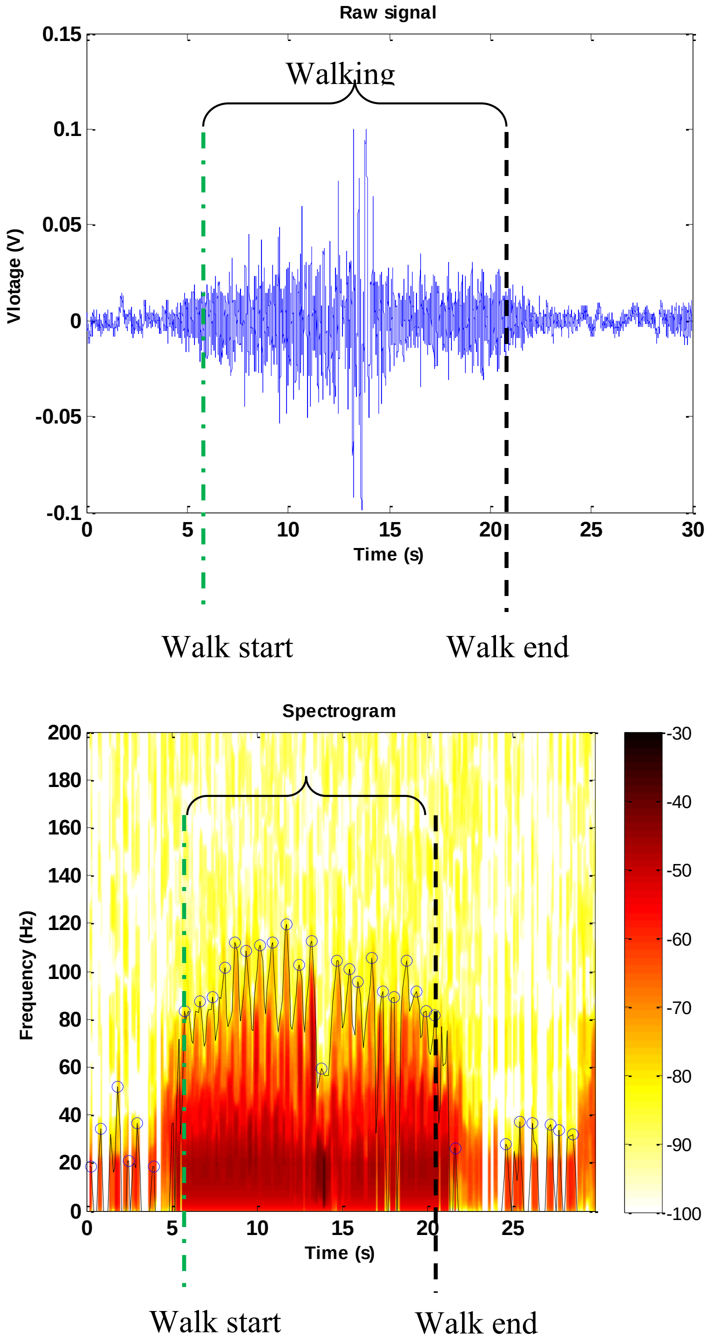


Figure 2.5 Human walk signal. (a) Raw signal; (b) Spectrogram.

2.3.4 Falling Object Experiments

This experiments show that the falling micro-Doppler signatures can be identified with radar at different positions in the lab. One sensor is installed on the longer wall with a height of 50 ft. The other one is placed on the ceiling at the center of the room and pointing down toward the floor. Both of the sensors are facing toward the room's center.

During the experiment, a mattress was set up in a vertical position. When participant pushed the mattress with neglected motion using hand from a standing still position, the mattress fell from a vertical to horizontal direction parallel to the floor. The mattress was pushed and fell three times in Figure 2.6 (1)-(3) as recorded by two working radar sensors. Although the signals from each sensor were different for the same fall, the signature could still be observed on the spectrogram.

Aside from the above system with two working sensors, we repeated the experiments under the system with a single working sensor in Figure 2.7 to check the possible interference of the two working sensors in Figure 2.6. Based on the observation, there is no noticeable interference between the two sensors when they work at same time. We also notice that the signal amplitude recorded by the ceiling radar sensor is larger than the wall recorded one.

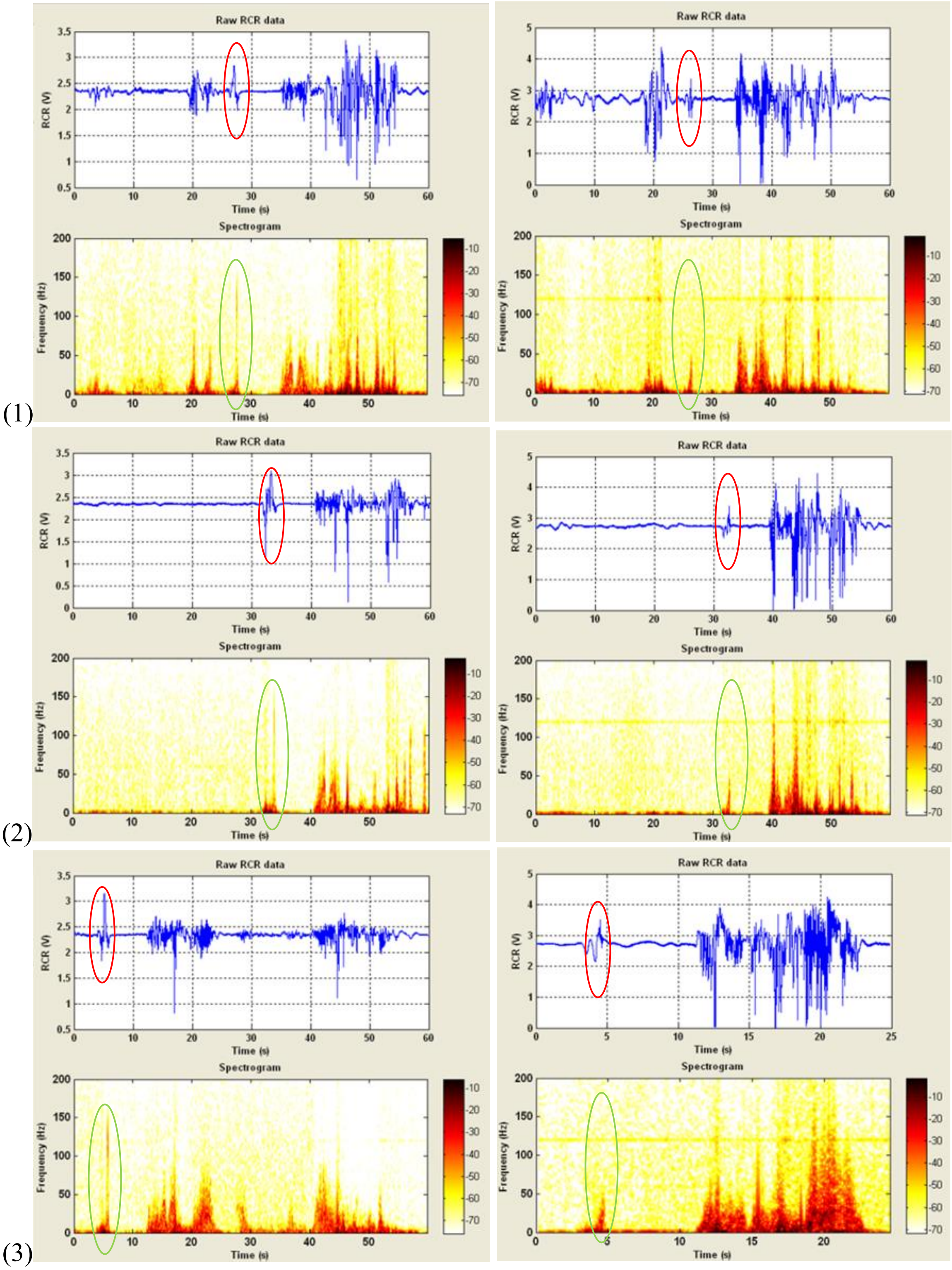


Figure 2.6 Motions captured by two sensors at same time. (a) Signal segments recorded by ceiling sensor are in the left column. (b) Signal segments recorded by the wall sensor are in the right column. Three mattress drops (1)-(3) are shown. In each row, signal segments were collected from both sensors to record a single mattress drop. The start

time of the signal segment was different since an individual data acquisition device was used for each sensor.

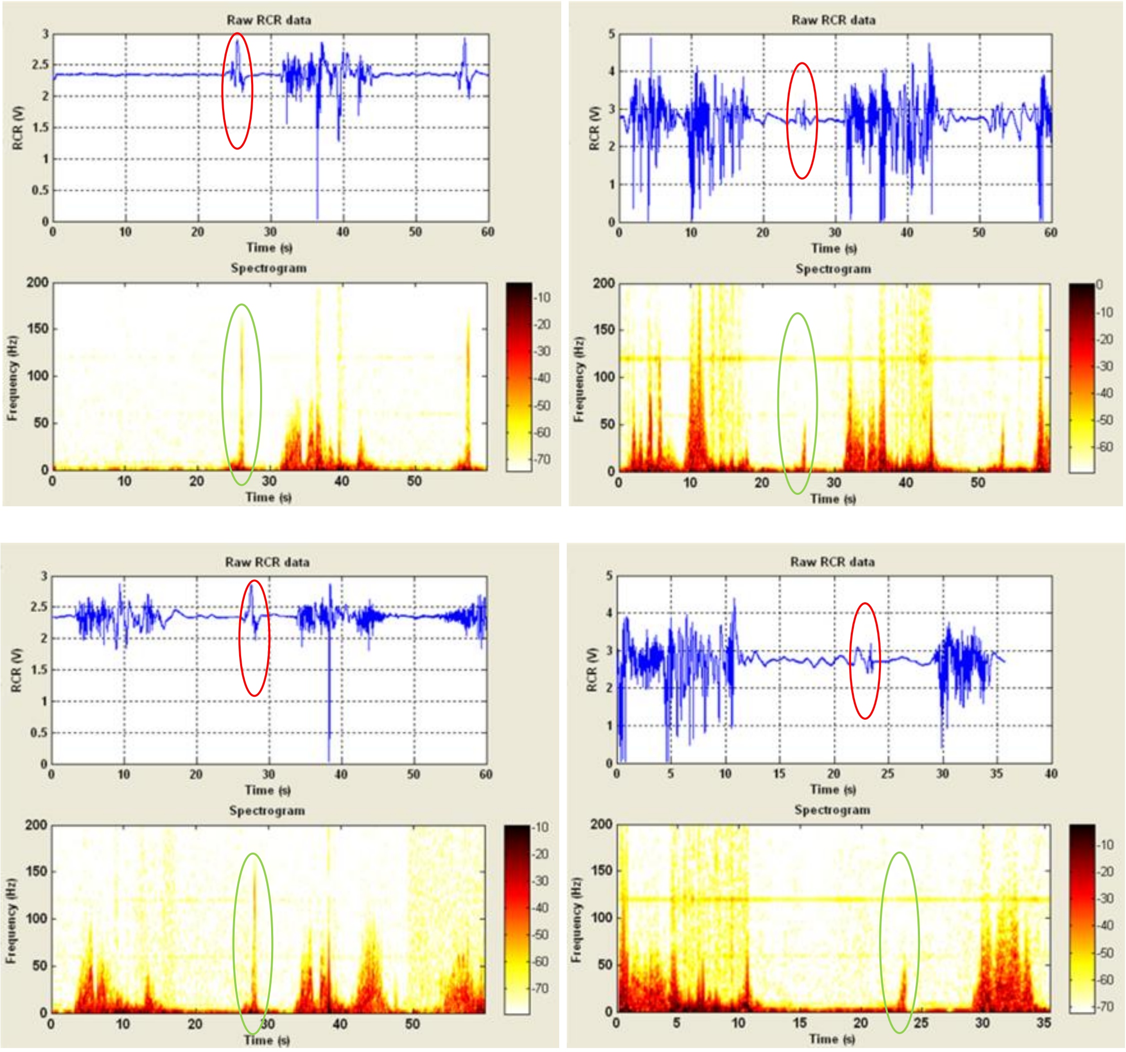


Figure 2.7 Motion captured by two sensors individually: (a) Signal segments recorded by ceiling sensor are in the left column. (b) Signal segments recorded by the wall sensor are in the right column. Four mattress drops are shown with each drop showing the signal segment and its spectrogram.

2.3.5 Micro-Doppler Signature for Human Fall

A fall is recorded in the signal segment shown in Figure 2.8(a). The red solid rectangle denotes the occurrence of the fall. The corresponding spectrogram is shown by Figure 2.8(b).

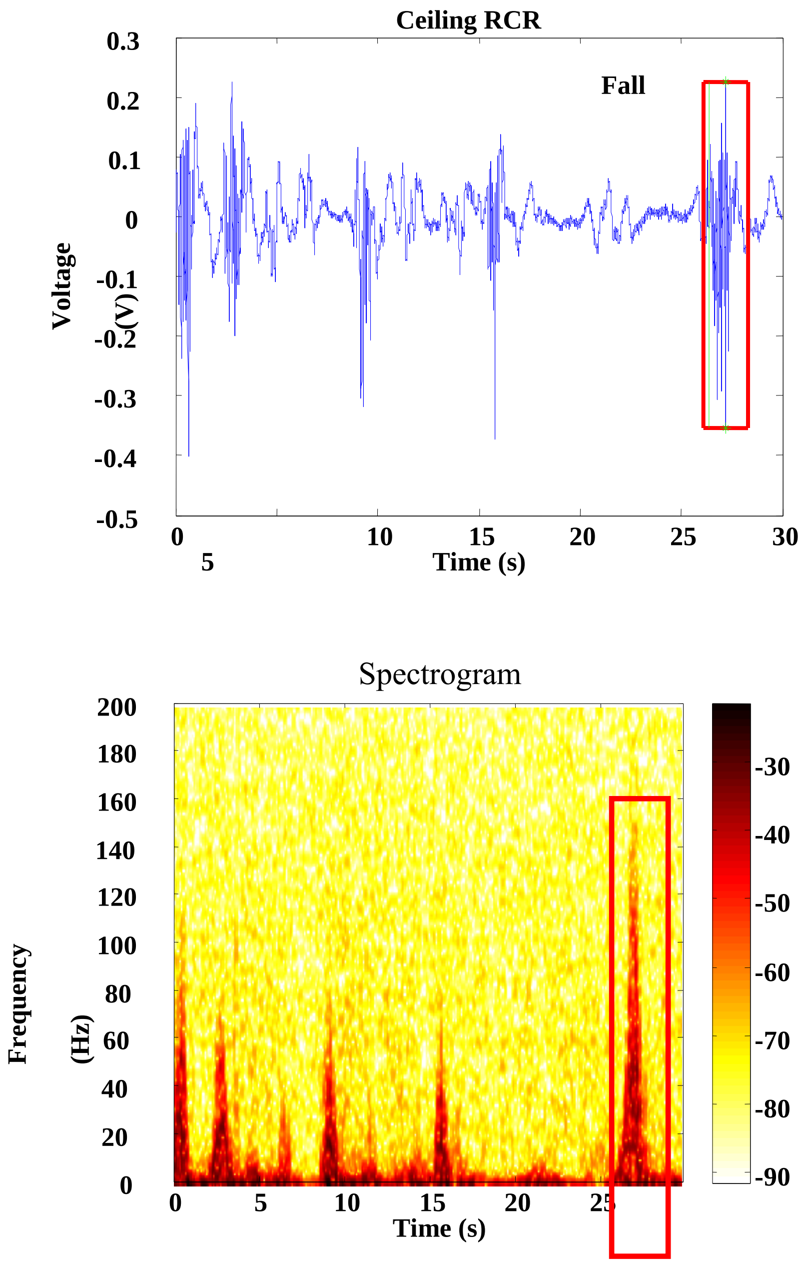


Figure 2.8 Human fall signal. (a) Raw signal; (b) Spectrogram.

On the raw signal, we cannot identify the fall with the amplitude or shape information. On the spectrogram, the fall peak covers a wider frequency region (0 to 160 Hz)which is wider than other activities shown in the same signal segment. This is because a fall includes multiple body movements, which generate more frequency components.

2.4 Summary

I have demonstrated that a certain motion can be recognized by the micro-Doppler signatures. With the swing experiment, we can understand that walking is a procedure by adding together different body part swings. Unlike the significant repeated walking pattern on the spectrogram, falling is a one-time activity. The falling object experiments showed us that it is possible to identify the fall pattern on the sensor signal. In the following sections, more methods are presented to detect the fall.

Chapter 3 Fall Detection System and Data Collection

This chapter will focus on a fall detection system with different configurations and the data collection in each system. In the lab room, there are three different configurations: two floor radar configurations [20]; one ceiling and one wall radar configuration [22]; one ceiling and one floor radar configuration. We collected fall data from each of the configurations. We also applied the floor radar and ceiling radar in actual senior apartments.

3.1 System Overview

The fall detection system is composed of a radar sensor system part and an algorithm processing part. We used the pulse Doppler range-controlled radar (RCR) developed by General Electric Co. The center frequency or carrier frequency is 5.8 GHz, which represents the frequency for the carrier wave. The carrier wave can transmit information through space as an electromagnetic wave. The carrier wave has a much higher frequency than the input signal. The measured range for the target velocity is 15.24 and 152.40 cm/s. For our in-room application, we set the range up to 6.1 meters for optimal coverage as shown in Figure 3.1.

The schema of the system for data collection and storage is shown in Figure 3.2. We used a wireless router to transmit the sensor data from the data acquisition board and to store the data on a nearby computer. The data was transferred to the server through the internet. The data acquisition board connected to the RCR as the data logger.

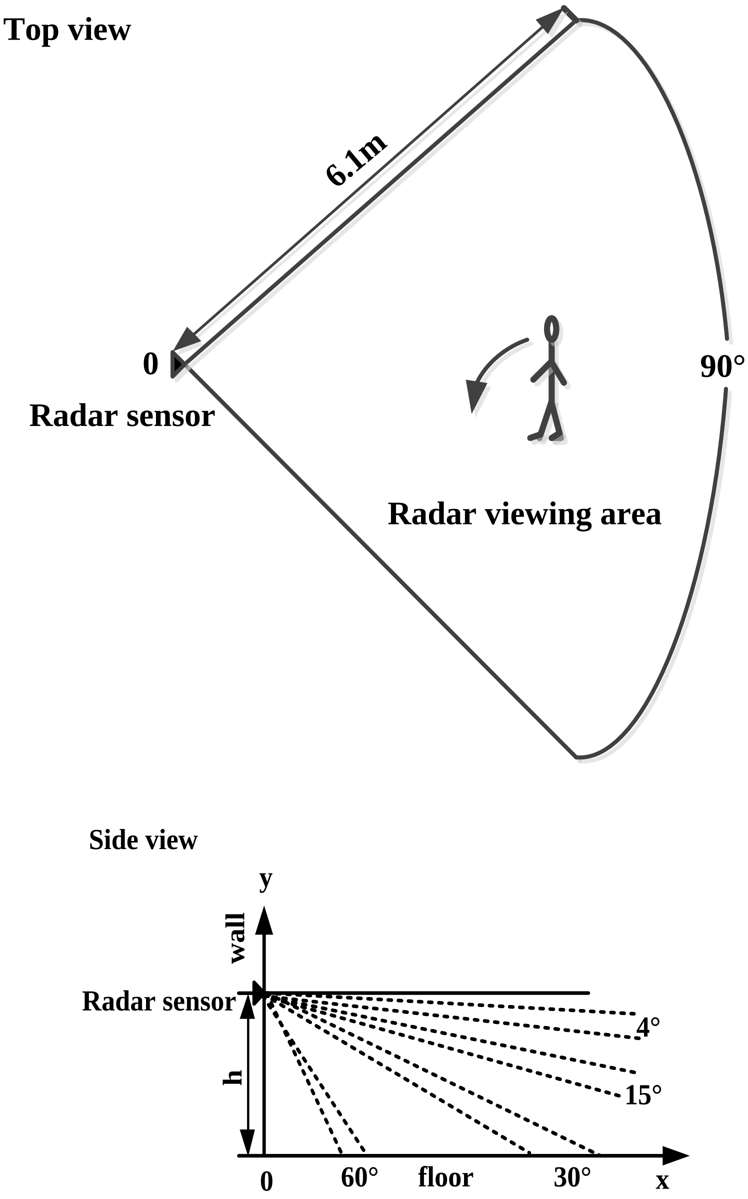


Figure 3.1 Coverage region of a range-controlled radar sensor: (a) Top view; (b) Side view. The inclination is tilted within 5˚ for each setup to get the clearest signal.

For the floor RCR, we put RCR and its auxiliary equipment, such as power suppliers, the data acquisition board, and the wireless router in a small portable box with a dimension 0.42 m x 0.32 m x 0.23 m as shown in Figure 3.3. In this box, the RCR was mounted on a bracket in the box with a height of 10 cm from top to bottom. The data acquisition board was mounted on the top of the RCR and the wireless router was attached beside the board. The power suppliers and other wires were put outside of the coverage view of the radar. For the best view of the inner structure of the box, we did not include the wireless router and power suppliers in the picture.

The surface of each floor RCR was wrapped by foil to exclude two kinds of potential false alarm sources: 1) the possible electronic fields formed by electric motors or high voltage equipment such as the heater; 2) the moving or vibrating objects such as fans, pulleys for a roll-down curtain, and conveyor belts on the treadmill machine.

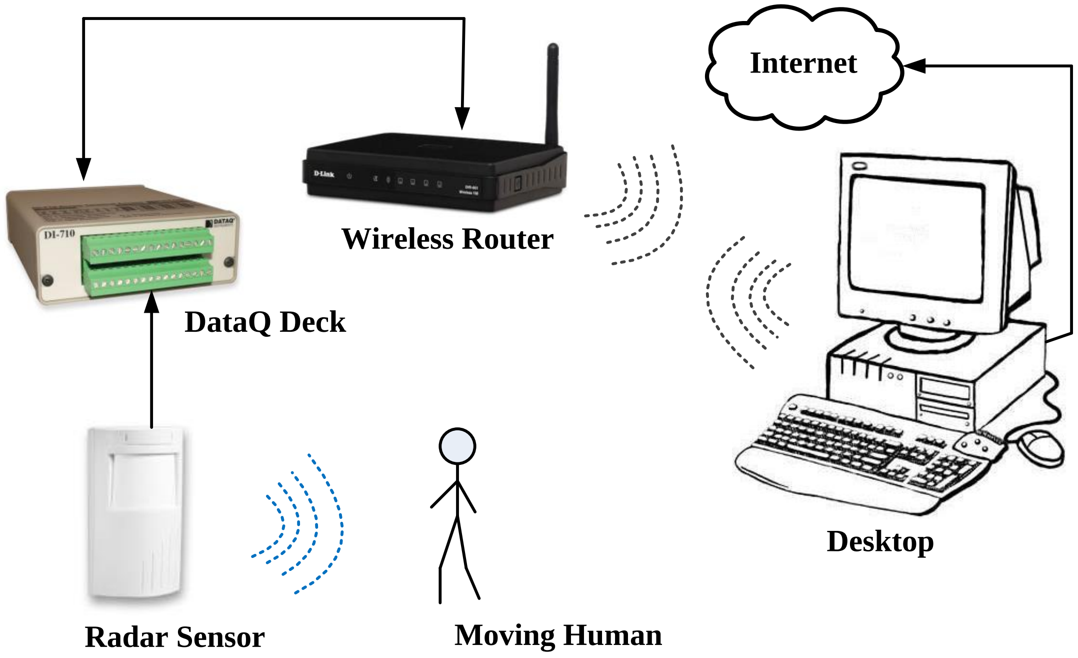


Figure 3.2 The schema of the system for data collection and storage.

Compared to in-room environment, there is less potential noise source on the ceiling since there is little equipment around. In order to record more information to find a fall, we left the ceiling RCR without foil cover. In lab environment, we attached the RCR on the beam in the ceiling in the center of the room. In the actual senior apartment, we went to the attic and fixed the radar in a wooden bracket above the ceiling in the center of the apartment living room and bathroom. The ceiling RCR is facing downward looking toward the room floor. Before each installation, all the RCRs are carefully checked and calibrated to obtain the clearest signal. The sampling frequency is set at 960 Hz to record the in-room human activities.

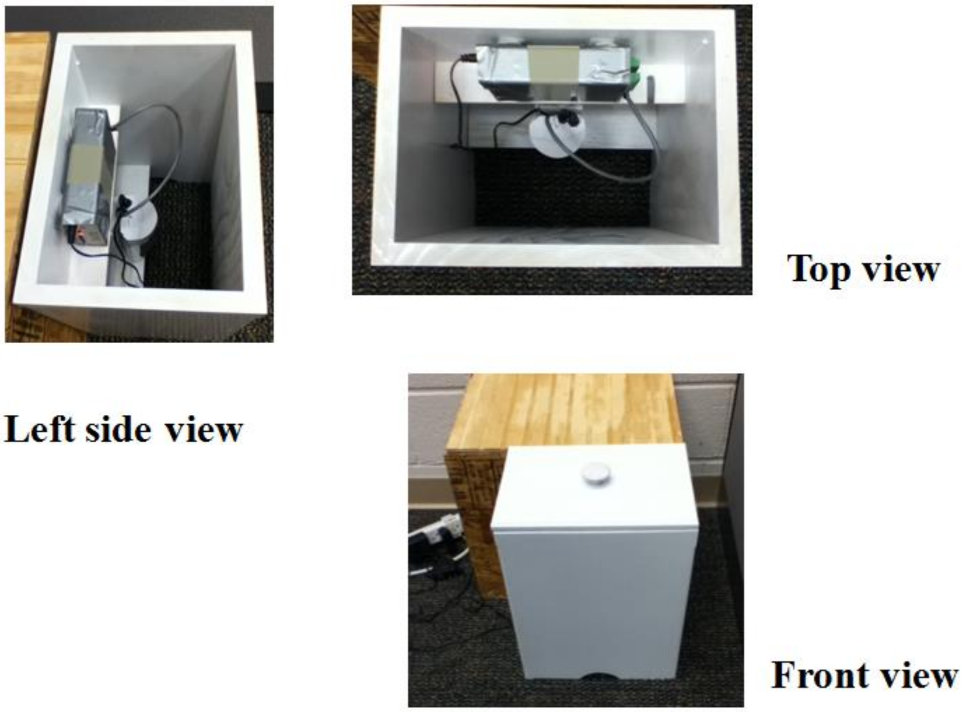


Figure 3.3 Floor radar sensor deployment. In the senior apartment, one sensor is set beside the front door pointing to the main aisle. The box dimension is 0.42 m x 0.32 m x 0.23 m (height x width x length).

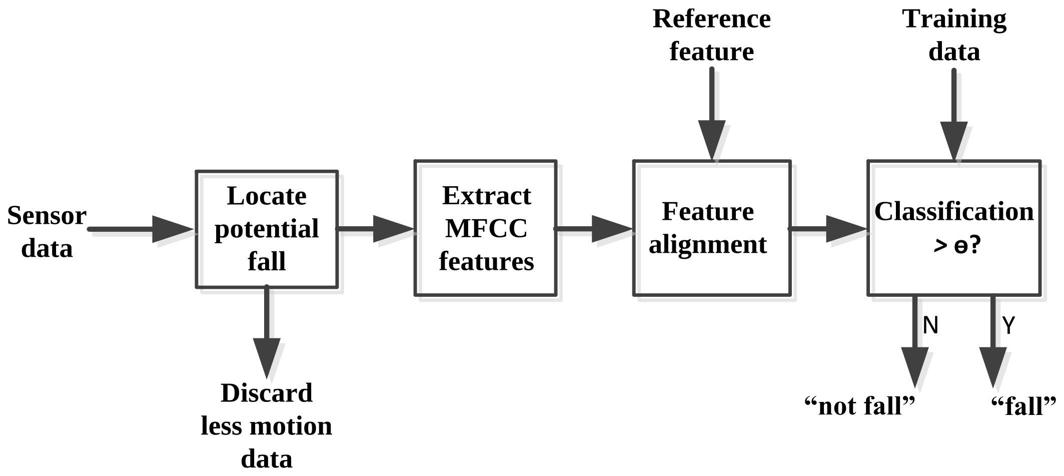


Figure 3.4 Block diagram for the processing procedure to detect a fall.

For the algorithm processing part, the whole procedure is displayed as the block diagram in Figure 3.4. The potential fall is filtered by passing the sensor data into a filter, which is based on the pattern analysis in the energy spectrogram. If the energy summation in the interested energy region increases fast in its nearby neighborhood, a potential fall candidate is found and ready for further recognition. This fast energy increase is described by finding a peak pattern on the spectrogram. If there is no peak pattern, we consider that there is no significant motion in the signal and filter it away from further processing. If there is a likely fall, MFCC features are extracted and will be aligned with a pre-selected fall reference feature [42]. A trained classifier is applied to those input features. The output confidence is used to decide whether it is a fall or not after comparing the confidence with the threshold. A detailed description for each diagram block is described in the following sections.

3.2 Lab Configurations

Three configurations will be presented in this section to provide some insights about the radar positions and their effect on detection performance. Each configuration has two radars. Configuration I, in Figure 3.5, has two RCRs placed on the floor. In Figure 3.7, Configuration II has a RCR on the wall and the other one on the ceiling. Configuration III involves one floor RCR and one ceiling RCR. The datasets collected with each configuration are also presented.

3.2.1 Lab Configuration 1 – Two Floor Sensors

The first configuration is presented by Figure 3.5. As shown in Figure 3.5 (b), the two floor sensors are oriented toward the center of the room. The distance from RCR1 to the room center is about 3.7 m, denoted by L1. The distance from RCR2 to room center is 1.85 m, denoted by L2. In Figure 3.5 (a), the subject is performing a forward fall towards RCR1 and cross RCR2. We note that RCR2 is closer to the fall than RCR1.

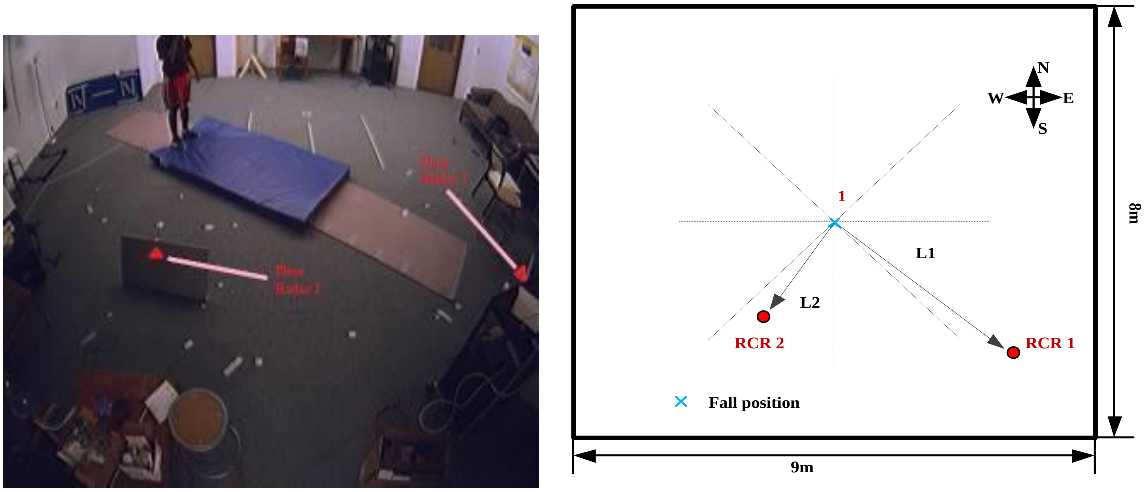


Figure 3.5 Sensor deployment setup in motion lab with Configuration I: two floor radars system. (a) Real view; (b) Position diagram.

Figure 3.6 Four types of fall for Configuration I. (a) Forward fall toward RCR 1; (b) Forward fall toward RCR 2; (c) Forward fall between RCR 1 & RCR 2.

In this Configuration, 90 falls were collected from two subjects. As seen in Figure 3.6, the subjects were falling in the center of the room, performing 20 forward falls towards RCR1, 25 falls towards RCR2, 25 falls between RCR1 and RCR2, and 20 falls away from RCR1. 341 non-fall activities were selected from walking, body sway, cylinder sway, squat, pick up a book from the floor, *etc*. More details information about the datasets is presented in Table 3.1.

Table 3.1 Dataset for Configuration I.

Fall	# files	Non-fall	# files
Balance–		Walk	
Forward to RCR1	20	Turn	
Cross RCR1	25	Sit down	
Forward to RCR1&2	25	Stand up	
Away from RCR1	20	Body sway	
		Bend over	
		Stretch arms	
		Step over	
Total falls	90	Total Non-falls	341

3.2.2 Lab Configuration 2– One Ceiling and One Wall Sensor

In the second experimental configuration, one RCR is placed in the room ceiling center and the other one is attached on the north wall at a height of 1.27 meters as shown in Figure 3.7. The ceiling RCR is vertically pointing down to the floor center of the room. The wall RCR is pointing horizontally to the chest of the standing subject in the room center, as shown in Figure 3.7 (a). In Figure 3.7 (b), falls were performed in a radial pattern. The radius R1 is 1 m, R2 is 2 m and R3 is 3 m. In Figure 3.7(a), the subject is standing in the center of the room and facing the north wall.

In Configuration II, we collected 126 falls from two subjects at 21 positions shown in Figure 3.7(b). At each position, each subject performs three types of falls (see Figure 3.8): forward fall, left side fall, right side fall. 817 non-fall activities such as walking and bending down are selected from the recorded radar signal at locations that show high energy density in the related spectrogram. The dataset information is shown in Table 3.2.

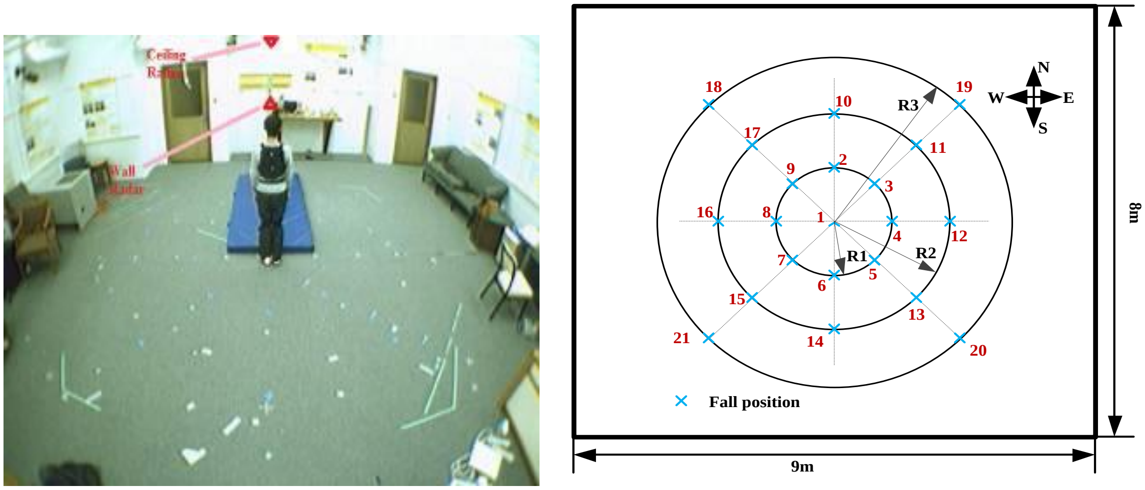


Figure 3.7 Sensor deployment setups in motion lab with Configuration II: wall and ceiling radars system. (a) Real view; (b) Position diagram.

Table 3.2 Dataset for Configuration II.

Fall	# files	Non-fall	# files
Balance–		Walk	
Forward	42	Turn	
Left	42	Sit down	
Right	42	Stand up	
		Body sway	
		Bend over	
		Stretch arms	
		Step over	
Total falls	126	Total non-falls	817

Figure 3.8 Three types of falls for Configuration II. (a) forward fall; (b) left side fall; (c) right side fall.

3.2.3 Lab Configuration 3 – One Ceiling and One Floor Sensors

In this configuration, one RCR is mounted on the room ceiling center and another RCR is on the floor, which is held by a tripod. In Figure 3.9(a), the subject is standing in the room center and falling towards the floor RCR. The position diagram for RCRs is shown by Figure 3.9(b).

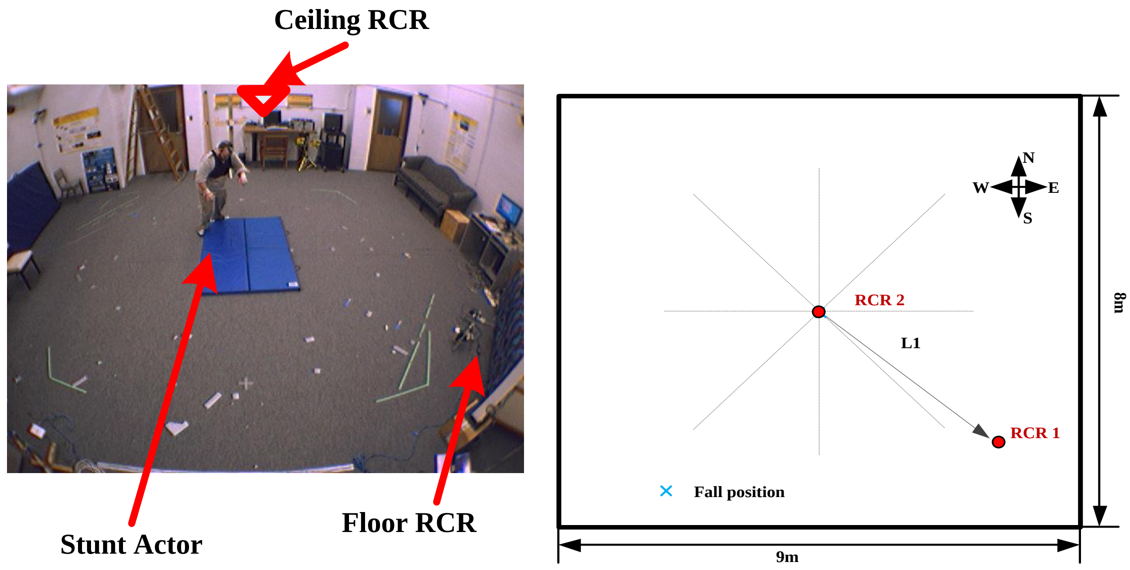


Figure 3.9 Sensor deployment setups in motion lab with Configuration III: floor and ceiling radars system. (a) Real view; (b) Position diagram.During the experiment, the mattress was in the center of the room.

In Table 3.3, stunt actors performed 21 types of falls in 105 files. The 383 non-fall files contained eight non-fall types in five daily scenarios. We located 349 potential falls as the false positives for the ceiling RCR and 361 potential falls as the false positives for the floor RCR.

Table 3.3 Dataset for Configuration III

Fall (21types)	# files	Non-fall (8 types)	# files
Balance–		Walk	175
Forward	5	Turn	14
Backwards	5	Sit down	35
Left	5	Stand up	35
Right	5	Body sway	18
Lose Consciousness–		Bend over	75
Forward	5	Stretch arms	15
Backwards	5	Step over	16
Left	5		
Right	5		
Crumple	5		
Trip & fall–			
Forwards	5		
Sideways	5		
Forwards	5		
Sideways	5		
Backwards	5		
Reach-fall (chair)–			
Forwards	5		
Left	5		
Right	5		
Forwards	5		
Backwards	5		
Couch Fall–			
Upper body first	5		
Hips first	5		
Total falls	105	Total non-falls	383

3.2.4 Lab Dataset

More details about the subject information in above three lab configurations are listed in Table 3.4. The subject information includes the gender, height, and weight information. The last three subjects are stunt actors trained by nursing school staff [36]. After training, these actors can fall simulating what an actual fall of elders looks like according to the patterns most evident in elderly fall analysis.The first four subjects are students. They try to act in the same way as the stunt actor to imitate the elders. The notations for the dataset under each configuration are FF_SYS, CW_SYS, and CF_SYS.

Table 3.4 Datasets and subject information for lab configurations

Configuration #	Subject #	Gender	Height (cm)	Weight (kg)	Fall #	Fall direction
Configuration I	1	male	188	88	20 20 5	Forward to RCR1 Forward to RCR2 Forward to RCR1&2
	2	female	162	53	20 20 5	Forward to RCR1&2 Away from RCR1 Forward to RCR2
Configuration II	3	male	183	83	21 21 21	Forward Left side Right side
	4	male	171	61	21 21 21	Forward Left side Right side
Configuration III	5	female	160	61	21	21 types of fall
	6	female	163	53	42	21 types of fall
	7	male	173	77	42	21 types of fall

3.3 Apartment Configuration

As part of the MU sensor guided fall study research group, I helped install the fall detection system in the real senior apartment, Tiger Place. All the elders live independently in this apartment. The University of Missouri-Columbia (MU) Institutional Review Board (IRB) approved this project, and we also obtained written consents from those residents who agreed to participate in this project. The stunt actor visited each apartment of those participants and performed fall and other non-fall activities. Besides the radar sensor, each participating apartment was also equipped with Microsoft Kinect, which measures and monitors subtle changes in the gait and movement of older people, and a webcam. The webcam was switched on during the stunt actor falls. This provided a convincing ground truth to validate the detection algorithms. To assure privacy protection to the residence, the webcam was not active during the day. The ground truth for falls generated by the residents can be observed through the depth image recorded by Kinect in the living room.

We placed two radar configurations in the Tiger Place senior apartments: a floor RCR and a ceiling RCR. The floor RCR box was set beside the apartment front door and pointed to the main path of the living room. In Figure 3.10, the stunt actor is falling away from the floor RCR. The mattress is aligned with the main path. The ceiling RCR was placed in the attic affixed to a wooden bracket above the center point of the living room's ceiling. One ceiling RCR is on the top of the apartment living room in Figure 3.10. Considering the room size influence on the detection and occurrence place for real falls, the other three ceiling RCRs were installed in the ceiling area of three bathrooms.

The data collection in each apartment still followed the 21 fall prototypes, as listed in Table 3.5.

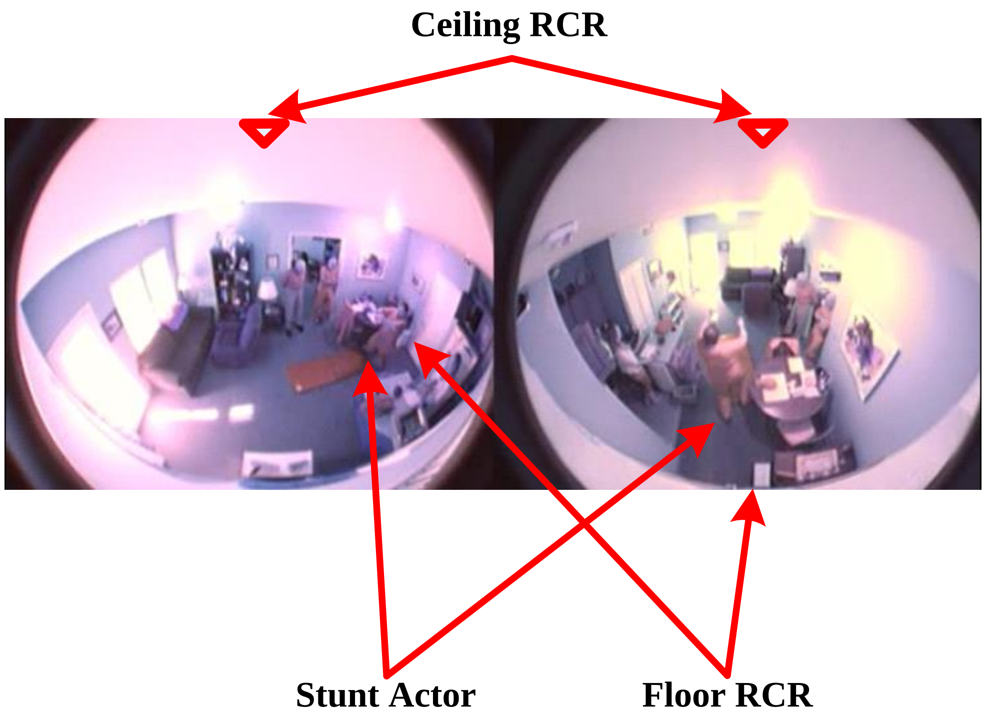


Figure 3.10 Senior apartment in Tiger Place. RCR 1 is on the floor and RCR 2 is on the ceiling. (a) Left side camera view. (b) Front camera view.

Table 3.5 DATA_TP has 20 Falls from Stunt Actors and 16 Nonfalls from senior Apartment.

Fall (21 types)	# files	Non-fall (8 types)	# files
Balance–		Bend at kneel and stoop	2
Forward	2	Bend, kneel, lie	2
Backwards	1	Sit with leg extended	1
Left	1	Sit-ups, stretches	2
Right	2	Ly to half kneel to stand	1
Losing Consciousness–		Trip during walk	1
Forward	1	Sudden stop during walk	2
Backwards	2	Walk stop turn around	1
Right	1	Walk and sit on chair	1
Crumple	1	Bend over	3
Trip & fall–			
Forwards	2		
Sideways	1		
Sideways	2		
Backwards	1		
Reach-fall (chair)–			
Backwards	2		
Couch Fall– upper body first	1		
Total fall	20	Total nonfall	16

3.4 Summary

In this chapter, we describe the different radar sensor configurations in lab and real senior apartments. In the lab room, three configurations were considered. We collected data with two floor radar configuration; one ceiling and one wall radar configuration; one ceiling and one floor radar configuration. We also applied the floor radar and ceiling radar in the Tiger Place senior apartments and collected the data once a month. In the next chapter, we are going to show the features for fall and non-fall detection.

Chapter 4 Fall Detection Algorithm

4.1 Signal Preprocessing

On the sensor data, we use energy burst peak finder to locate the motion segment which includes a potential fall activities [22].

As we mentioned in chapter 2, the short time Fourier transform (STFT) can divide a signal into several sub-windows, perform the fast Fourier transform (FFT) for each window and store the complex amplitudes in a table. The columns of this table represent time and the rows of this table represent frequency. Thus we can determine the sinusoidal frequency and phase content of local section for the signal. For the incoming raw radar signal $x(n)$, we rewrite the short time Fourier transform of equation (2.11) as

$$STFT(m, f) = \sum_{n=-\infty}^{\infty} x(n)w(n-m)e^{-jwn}\ , \qquad (4.1)$$

where $w(n)$ is the Hamming window and is defined by

$$w(n) = 0.54 - 0.46\cos(\frac{2\pi n}{N-1})\,. \qquad (4.2)$$

The spectrogram is defined as an intensity plot of STFT magnitude by the magnitude squared of STFT

$$spectrogram\{r(n)\} \equiv |STFT(m,w)|^2\,. \qquad (4.3)$$

The raw sensor signals and their spectrograms of a fall and a non fall signal are shown in Figure 2.7.

The interested frequency range is [25, 50] Hz for fall detection problem. So we sum up the energy in the interested frequency range after STFT via

$$EB(m) = \sum_{w=25/(2\pi)}^{50/(2\pi)} STFT(m,w) . \tag{4.4}$$

Then we smooth the curve with moving average to obtain the energy burst curve with K times moves

$$\hat{EB}(m) = \sum_{i=0}^{K-1} EB(m-i) \cdot \tag{4.5}$$

On the energy burst curve, the abruptly change is represented by a peak, which means the occurrence of an extreme motion, such as fall. Peak finder with threshold is used to locate this potential fall location. By adjusting the threshold, the number of potential fall activities can be varied.

4.2 Feature Extraction Method

Mel-frequency Cepstral Coefficients (MFCCs) are usually used as the acoustic features [33, 34]. Since MFCCs features consider human perception sensitivity with respect to frequencies, MFCCs features have the best performance for sound recognition. MFCCs are used as the core features for recognition in most audio or acoustic signal retrieving or detection techniques.

We apply this feature extraction method to radar signal in fall detection problem. We extract MFCC features from the selected window on radar signal. We do the classification between fall and non fall activities by using the extracted feature. First, we explain this method step by step. Secondly, we tune different parameters and evaluate its performance to find the best parameters [22].

There are six steps to compute the MFCCs features.

(1) Pre-emphasis. The aim is to compensate the high-frequency part which was suppressed during sound production. This can be performed by sending the input signal *x(n)* through a high-pass filter

$$y(n) = x(n) - ax(n-1) \ , \qquad (4.6)$$

where we use 0.97 as the factor a.

(2) Frame blocking and Hamming windowing. The input signal frame is segmented into sub-frames. Each sub-frame has a size of 256 samples. If the signal sampling frequency is 960Hz, the sub-frame sampling frequency is 3kHz and the sub-frame frame rate is 100 frame/second, the number of sub-frame in a input signal frame is computed with *L*=fix((960*2-256)/(3000/100))=55. The notation “fix” means the nearest integer in the direction of zero. Each sub-frame has to be multiplied with a Hamming window to keep the continuity of the first and the last points in the sub-frame.

(3) Zero-padding and fast Fourier transform. Zero-padding is needed if the sub-frame size is smaller than the point of DFT. We used *N*=512 points DFT. In spectral analysis, different timbres in the sound signal correspond to different energy distribution over frequencies. So FFT is used to each sub-frame which has been windowed by Hamming window as

$$X(k) = \sum_{n=0}^{N-1} w(n)x(n)\exp(-j2\pi kn/N), \qquad (4.7)$$

where *k* corresponds to the frequency $f(k) = kF_s/N$, Fs is the sampling frequency *k*=0, 1, …, *N*-1.

(4) Filter bank constructed by triangular Band-pass Filters. The relation of mapping the power of the spectrum onto the mel scale can be described by

$$Mel(f) = 1127 \bullet \ln(1 + f/700). \qquad (4.8)$$

Mel-frequency reflects the human's subjective aural perception. It is proportional to the logarithm of the linear frequency.

An array of triangular bandpass filters is assigned over the Mel-frequency with equal interval spaces but with logarithm based spaces over linear frequency to multiply the frequency response to get the energy of each bandpass filter. Based on the experience and several trials with the number of the band pass filter, we selected 30 triangular bandpass filters. The first 10 filters are with equal linear space as 20 and another 20 filters using logarithmic space as 1.0712. The bank frequency range covers from 5Hz to 839.5745Hz.

The filter bank is constructed with N_f equal area triangular filers. The *i*-th filter is defined by

$$H_i(k)=\begin{cases}\dfrac{2(kF_s/N-f_{bd_{i-1}})}{(f_{bd_{i+1}}-f_{bd_{i-1}})(f_{bd_i}-f_{bd_{i-1}})}, if \quad f_{bd_{i-1}}\le kF_s/N<f_{bd_i}\\ \dfrac{2(f_{bd_{i+1}}-kF_s/N)}{(f_{bd_{i+1}}-f_{bd_{i-1}})(f_{bd_{i+1}}-f_{bd_i})}, if \quad f_{bd_i}\le kF_s/N<f_{bd_{i+1}}\\ 0,\ if \quad kF_s/N<f_{bd_{i-1}}, kF_s/N>f_{bd_{i+1}}\end{cases} \tag{4.9}$$

where $i=1,\ldots,N_f$, $f_{bd_{i-1}}$ is the lower boundary of the i-th filter, f_{bd_i} is the center of the *i*-th filter, and $f_{bd_{i+1}}$ is the upper boundary of the *i*-th filter. *k* corresponds to the *k*-th coefficient of the *N* point DFT.

The boundary points of the N_f filters are shown as

$$f_{bd_i}=\hat{f}_{mel}^{-1}\left(\hat{f}_{mel}(f_{low})+i\frac{\hat{f}_{mel}(f_{high})-\hat{f}_{mel}(f_{low})}{N_f+1}\right), \tag{4.10}$$

where f_{high} and f_{low} are the boundaries of the filter bank, and

$$\hat{f}_{mel}^{-1} = 700 \bullet \left(\exp\left(\frac{\hat{f}_{mel}}{1127} \right) - 1 \right) . \qquad (4.11)$$

The filter bank is normalized and the sum of coefficients for each filter equals one. Thus, the *i*-th filter meets the constraint

$$\sum_{k=1}^{N} H_i(k) = 1, i = 1, ..., N_f . \qquad (4.12)$$

(5) Discrete Cosine Transform (DCT) & coefficient matrix.

The log-energy output of the *i*-th filter is computed by

$$E_i = \log_{10}\left(\sum_{k=0}^{N-1} | X(k) | H_i(k)\right), 0 \le i < N_f , \qquad (4.13)$$

where N_f is the number of triangular filters in the filter bank and we use N_f=30.

We apply DCT on 30 log energy E_i obtained from the triangular bandpass filters to have Mel-frequency cepstral coefficients. Usually, the number of cepstral coefficient M is smaller than the number of filters N_f in the filter bank. For each sub-frame, we obtain M cepstral coefficients with

$$C_m = \sum_{i=1}^{N_f} \cos(m(i - 0.5)\pi / N_f) E_i \text{, m=1, …, M.} \qquad (4.14)$$

Each sub-frame generates Mel-frequency cepstral coefficients. For a selected 2-second signal window, the total number of sub-frames *T* is 55. The MFCC features are saved as a matrix, which has a dimension of *M*x*T*. For the *i*-th processing 2-second signal window, the feature matrix is defined by

$$[C]_i = \begin{bmatrix} C_{1,1} & \cdots & C_{1,T} \\ \vdots & \ddots & \vdots \\ C_{M,1} & \cdots & C_{M,T} \end{bmatrix}_i , \qquad (4.15)$$

where T equals to 55 and M equals to 7. In the following classification, we threw away the coefficients of the first line of feature matrix in Equation (4.15) since they are the dominant components and do not play a positive role in classification.

4.3 Classification

For the performance consistent consideration, the fall detection system uses four different classifiers: nearest neighbor, naïve Bayes, support vector machine (SVM) and hidden Markov model (HMM). The performance evaluation method is also proposed.

4.3.1 Nearest Neighbor

The Nearest Neighbor (NN) is the most popular method in recognition. In metric space, an unknown sample is assigned to the class of the nearest point. Usually, we use Euclidean distance to measure the distance between the unknown sample and other samples.

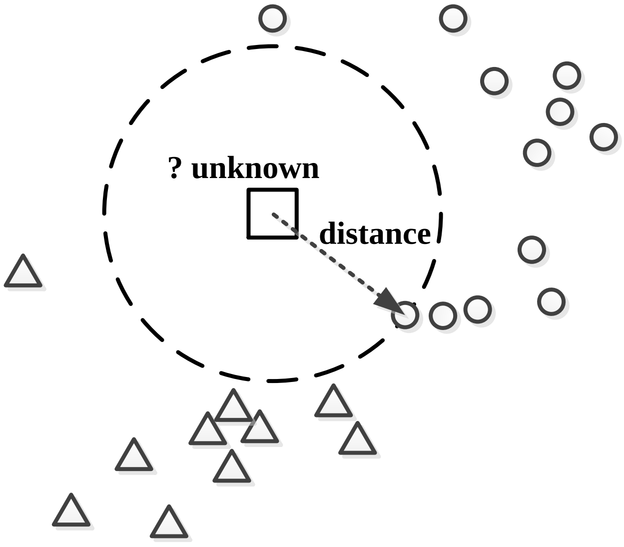


Figure 4.1 A nearest neighbor. The unknown square point is assigned to the circle class since the nearest point to the square is a circle.

For the MFCC features, we need to extend the Euclidean distance to a *(M-1)×T* dimensional coefficient matrix $[C]_i$ where M is the number of feature dimension for each sub-frame. T is the number of sub-frame in a processing frame. The distance metric between unknown sample $\vec{s}$ to another sample $\vec{t_k}$ is presented by

$$Dist_{s,t_k} = |\vec{s} - \vec{t_k}| = \left(\sum_{i=1}^{T} \sum_{j=1}^{M-1} (s_{ij} - t_{k,ij})^2 \right)^{1/2} . \tag{4.16}$$

The aim is to find the minimum distance

$$\mathrm{Dist}_{\mathrm{s,t_i}} = \min\{\mathrm{Dist}_{\mathrm{s,t_1}},...,\mathrm{Dist}_{\mathrm{s,t_k}},...\}, \tag{4.17}$$

where $i,k = 1,...,K-1$ and K is the total sample number. Then assign $\vec{s}$ to the class of the sample $\vec{t_i}$.

4.3.2 Naïve Bayes

The naïve Bayes classifier is based on the independent assumption that a particular feature of a class is independent to other features' presence or absence in this class. This classifier only requires a small amount of training data to predict the necessary parameters for classification. We only need to determine the variances of the variables for each class rather than the entire covariance matrix.

We reshaped the matrix into a feature vector *x* for each signal frame. The feature vector *x* is independent, and we have

$$p(x) = \prod_{i=1}^{(M-1)T} p(x_i), \tag{4.18}$$

where $x = [x_1,...,x_{(M-1)T}]$ is the feature vector and *(M-1)×T* is the total number of elements in this vector. For fall detection, the class number N_c is 2 and we have c_{fall}, class of fall activities, and $c_{nonfall}$, class of non-fall activities. So the prior probabilities for the two classes are computed by

$$p(c_{fall}) = \frac{\# fall}{\# fall + \# nonfall} \ , \tag{4.19}$$

and

$$p(c_{nonfall}) = \frac{\# nonfall}{\# fall + \# nonfall} \, . \tag{4.20}$$

Based on a given feature vector *x* and prior probability of class observation, the probability of a class is

$$p(c_j \mid x) \propto p(c_j) \prod_{i=1}^{(M-1)T} p(x_i \mid c_j) \, , \ j=1, 2. \tag{4.21}$$

The classification rule was developed by [38].

$$c = \arg\max_{c_j \in C} p(c_j \mid x) \propto \arg\max_{c_j \in C} p(c_j) \prod_{i=1}^{(M-1)T} p(x_i \mid c_j) \, . \tag{4.22}$$

We assume that $p(x_i \mid c_j) \sim N(\mu_{ij}, \sigma_{ij}^2)$. This normal distribution has the mean and variance for the *i*-th element of the class c_j. We computed the log likelihood for $p(c_j \mid x)$ with

$$Ln(p(x_i \mid c_j)) = -\frac{(x_i - \mu_{ij})^2}{2\sigma_{ij}^2} - \frac{1}{2}\ln(2\pi\sigma_{ij}^2) \, . \tag{4.23}$$

Taking the log to Equation (4.21), we have

$$L(c_j \mid x) \propto \ln(p(c_j)) + \sum_{i=1}^{(M-1)T} \ln(p(x_i \mid c_j)) \tag{4.24}$$

As a result, the classification rule in Equation (4.22) is rewritten as

$$c = \arg\max_{c_j \in C} L(p(c_j \mid x)) \propto \arg\max_{c_j \in C} \sum_{i=1}^{(M-1)T} \ln(p(x_i \mid c_j)) - \ln(p(c_j)) \, . \tag{4.25}$$

4.3.3 Support Vector Machine

Support vector machine (SVM) is another commonly used classification method. An SVM model is a representative of sample points in a mapped higher dimensional space. In this space, the samples of different classes are separated by a clear gap that is as wide as possible. The two sides of this gap are constructed by two parallel hyperplanes to separate the data. The maximum distance between those two hyperplanes is called a margin. The purpose of SVM in training is to maximize the margin and create the widest gap. We assume that the training dataset is represented with sample-label pairs $\{(x_i, y_i) \mid x_i \in \Re^p, y_i \in \{-1,1\}\}$, *i=1,...,K*. Within *K* training samples, the *i*-th data sample x_i is labeled as y_i. For linearly separable data in Figure 4.2, *w* is the orientation vector of the decision plane and *b* is the bias. The offset of the hyperplane to the origin along the normal vector *w* is $\frac{b}{\|w\|}$. The set of points *x* on any hyperplane satisfies

$$w \bullet x - b = 0. \qquad (4.26)$$

The “margin” refers to the two hyperplanes which can separate the data without any data between them with the maximized distance. These two hyperplanes are presented by

$$w \bullet x - b = 1, \qquad (4.27)$$

and

$$w \bullet x - b = -1. \qquad (4.28)$$

The distance between these two hyperplanes is $\frac{2}{\|w\|}$.

To avoid the samples falling between the margins, all the samples should satisfy the constraint. Our aim is to solve the problem

$$\min_{w,b}\left\{\| w^T w \|\right\}, \qquad (4.29)$$

subject to

$$y_i(w \bullet x_i - b) \geq 1,\ i=1,...,K. \qquad (4.30)$$

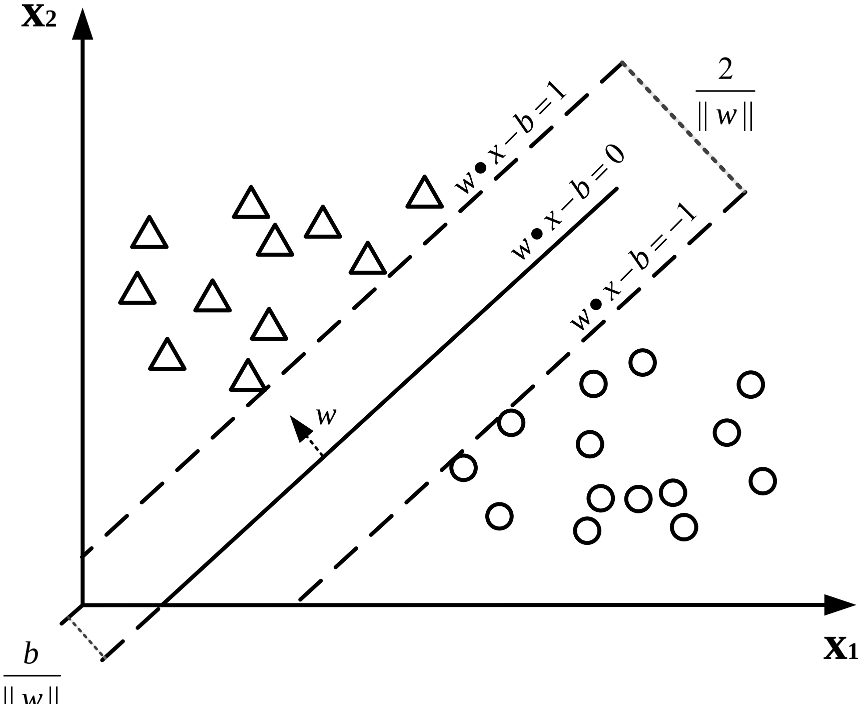


Figure 4.2 An SVM trained with samples from two classes. Maximum-margin hyper plane is represented by the solid line between the two margins.

For the case where no hyperplane exists to split the samples, a soft margin method is used by considering a margin slack variable ξ_i. It is used to measure the degree of misclassification of the data sample x_i. The SVM are constructed by optimizing

$$\min_{w,b,\xi}\left\{\frac{1}{2}(w^T w) + C\sum_{i=1}^{K}\xi_i\right\} \qquad (4.31)$$

Subject to

$$y_i(w^T\phi(x_i) + b) \geq 1 - \xi_i,\ \xi_i \geq 0,\ i=1,...,K, \qquad (4.32)$$

where $\phi(\bullet)$ is the mapping function and C is the penalty parameter of the error term.

Using the method of Lagrange multipliers, the problem becomes

$$\min_{w,b,\xi} \max_{\alpha,\beta} \left\{ \frac{1}{2}(w^T w) + C\sum_{i=1}^{K}\xi_i - \sum_{i=1}^{K}\alpha_i[y_i(w^T\phi(x_i)+b)-1+\xi_i] - \sum_{i=1}^{K}\beta_i\xi_i \right\} \text{ with } \alpha_i, \beta_i \geq 0\,. \quad (4.33)$$

I used LibSVM [22, 35] here. It has a SVM training step and SVM prediction step. The SVM training algorithm builds a model which assigns new examples into one category or other categories. In SVM prediction step, the unknown examples are mapped to the same space and their classes are decided by seeing which side of the gap they fall on. Because the features' dimension is as high as 55*6=330 for each sample, we used linear kernel in SVM for computational efficiency.

4.3.4 Hidden Markov Model

For the hidden Markov model (HMM), the state space is the set of $S = \{s_1, ..., s_N\}$ and the observation space is the set of $O = \{o_1, ..., o_M\}$. For an observed sequence $Y = \{y_1, ..., y_T\}$ with a length of T, $y_t = i$ represents the observation o_i at time t. The path $X = \{x_1, ..., x_T\}$ is a sequence of hidden states that generate the sequence of observations Y. In HMM, we cannot observe the N states directly. Markov property says that the state occupied at time t+1 only depends on the state occupied at time t. The *state transition matrix* is denoted as A={ $a_{i,j}$ } with *i*, *j*∊[1,…,N]. The element is defined as $a_{i,j} = \mathrm{P}(x_{t+1} = s_j \mid x_t = s_i)$ with $\sum_j a_{i,j} = 1$. It describes the probability of occupying state s_j at time t+1 from the state s_i at time t. Ergodic requests $a_{i,j}$ is non-negative. For the left-right model, A is upper triangular and $j \geq i$. The model can emit finite M symbols from each state. The *observation emission probability matrix* is represented by B={ $b_{j,k}$ } with

$j\in[1,\ldots,N]$ and $k\in[1,\ldots,M]$. The element $b_{j,k} = P(y_t = o_k \mid x_t = s_j)$ describes the emitted k-th o_k from the state s_j. For an observation y_t, the observation emission probability is computed by

$$b_j(y_t) = \sum_{d=1}^{D} c_{jd} N(y_t, \mu_{jd}, \Sigma_{jd}), d = 1,\ldots,D; j = 1,\ldots,N. \tag{4.34}$$

where the M multivariate normal distribution represented by

$$N(y_t, \mu_{jk}, \Sigma_{jk}) = (2\pi)^{-M/2} \mid \Sigma_{jk} \mid^{-1/2} \exp(-\frac{1}{2}(y_t - \mu_{jk})\Sigma_{jk}^{-1}(y_t - \mu_{jk})) . \tag{4.35}$$

The *initial state probability vector* is defined as $\pi_i = P(x_1 = s_i)$. It represents the probability of occupying state s_i at t=1. An HMM is defined with a set of parameters as λ={$\pi_i, a_{i,j}, b_{j,k}$} or {π, A, B}.

In the training part, we trained λ_f as the fall model and λ_{nf} as the non-fall model with the training dataset $[Y^1]_{K\times T}$, K training sequences with a length of T. Using the Baum-Welch algorithm, we obtained the learned A, B, π parameters for each model.

In testing part for an unknown observation o_t, we use 1 as M. Given the observation sequence, HMM can generate the observation sequence. For each model, we find the maximum probability of all possible state transition paths that produce o_t. o_t belonging to the fall class if we have the relation of

$$\Pr ob(o_t \mid \lambda_f) > \Pr ob(o_t \mid \lambda_{nf}) . \tag{4.36}$$

The probabilities expressed above with each model was computed using the Viterbi algorithm.

Details of the Baum-Welch algorithm and Viterbi Algorithm are described in [129]. We use h2m in this paper [130].

4.4 Evaluation Criteria and ROC

4.4.1 Confidence Calculation

A receiver operating characteristic (ROC) curve is used to evaluate the performance of the classifiers and features. By changing the discrimination threshold of a binary classifier system, the ROC curve describes the relation between true positive rate (TPR) and false positive rate (FPR). For our case, we generate a series of confidence for each classifier. Then a ROC curve can be produced by thresholding those confidence values.

Considering the difference among classifiers, fall confidence for NN and naïve Bayes can be computed by

$$Conf_{NN/NB} = normalized(\frac{Dist_{fall} - Dist_{nonfall}}{Dist_{fall}}), \tag{4.37}$$

where $Dist_{fall}$ is the distance to the nearest fall from current testing sample, and $Dist_{nonfall}$ is the distance to the nearest non-fall from current sample.

For SVM, the probability for the current classification result is taken as the fall confidence. For HMM, the likelihood difference between fall model and non-fall model is considered the fall confidence.

4.4.2 ROC Curve and AUC

A receiver operating characteristic (ROC) curve is used to evaluate the performance of the fall detection result. By changing the discrimination threshold of the binary classifier system, the ROC curve describes the relation between true positive rate (TPR) and false positive rate (FPR). For SVM, the probability for the current classification result is taken as the fall confidence. The ROC curve is produced by thresholding those confidence values.

On a ROC curve, area under the curve (AUC) can be calculated by using trapezoidal approximations with adjacent two points (x_{k-1}, y_{k-1}) and (x_k, y_k),

$$AUC_{ROC} = \sum_{k=1}^{n} (y_k - y_{k-1})(x_k - x_{k-1}) \cdot \tag{4.38}$$

In practice, we select the threshold at an operating point on the ROC. At the operating point, the classifier gives the best trade-off between the costs of failing to detect positives against the costs of raising false positives. With the selected threshold, we get the values for true positive (TP), true negative (TN), false positive (FP), and false negative (FN) in a confusion matrix. We calculate the following statistical measures to compare the detection result.

Accuracy (ACC) represents the ratio of correct detected fall and non-fall to the total number of samples. It can be calculated with

$$ACC_{OP} = \frac{TP + TN}{TP + FN + FP + TN} \cdot \tag{4.39}$$

Specificity (SPC) means no false positive. It is defined as

$$SPC_{OP} = \frac{TN}{TN + FP} \cdot \tag{4.40}$$

Sensitivity (SEN) denotes no false negative. It is computed with

$$SEN_{OP} = \frac{TP}{FN + TP} \cdot \tag{4.41}$$

Matthews correlation coefficient (MCC) is a measure used if the two classes have very different sizes. It is in [-1, 1]. MCC equals to 1 if the detected result has the perfect match with the ground truth. It equals to -1 if the detected result totally disagrees with the ground truth. It has been defined as

$$MCC_{OP} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} . \qquad (4.42)$$

From the above sections 4.1- 4.4, we have introduced the fall detection algorithm and the performance evaluation criteria, which is also generalized in Algorithm 4.1. More details about the MFCC parameter selection are discussed in Section 4.5.

Algorithm 4.1 Fall detection algorithm.

Assumptions:

The original signal is represented with *sigOrig*;

The signal window size is winSize=2 s;

The signal sampling frequency is *fs*;

The dataset has M fall and N nonfall events;

The function **MFCC_featExtract** is described in section 4.2;

The function **leaveOneOut** represents leave one out cross validation (in section 4.4) with a classifier (in section 4.3);

The function **thresholdConfidence** is used to compute the false positive, true positive and the area under the ROC curve.

```
falsePositive=[]; truePositive=[]; AUC_value=0;
For numEvent=1: (M+N)
    Preprocessing:
```

```
        Load the original signal sigOrig;
        Locate the fall or non-fall event position $Pos_{event}$;
        Compute the window length: winLen= winSize*fs;
        Compute the boundaries:
            $lo\_b = \lfloor Pos_{fall} - winLen / 2 \rfloor$;
            $hi\_b = winLen + lo\_b - 1$;
        Take the signal segment out: $sigSeg = sigOrig(lo\_b : hi\_b)$;
    Extract features: featEvent(numEvent)=MFCC_featExtract(sigSeg);
End
Leave one out cross validation with the dataset featEvent,
    confidenceValue=leaveOneOut(featEvent, classifierType);
Threshold the confidence to produce a ROC curve and compute AUC,
    [falsePositive, truePositive, AUC_value]
                    =thresholdConfidence(confidenceValue) .
```

4.5 Parameter Selection

We are looking for the best parameters for features with minimum dimension. Those parameters include the signal window length for extracting features from 0.5s-6s, the number of cepstral coefficients from 3-43, the sub-frame number adjusted by sampling frequency from 1 kHz-10 kHz. The feature alignment is also discussed.

4.5.1 Signal Window Length for Extracting Features

We use NN, naïve Bayes and SVM classifiers to find the optimal signal window length to generate the best performance efficiently. The signal window size is examined from 0.5s to 6s with an interval of 0.5s. Algorithm 4.2 presents the method to compute the performance and time cost with different signal window length.

Algorithm 4.2 finds the optimal signal window length.

Assumptions:

The window size searching range is in 0.5, 6 s with an interval of 0.5s;

The original signal is represented with *sigOrig*;

The signal sampling frequency is *fs*;

The iteration counter is set as itr=0;

The dataset has M fall and N non-fall events;

The function **MFCC_featExtract** is described in Section 4.2;

The function **leaveOneOut** represents leave one out cross validation (in Section 4.4) with a classifier (in Section 4.3);

The function **thresholdConfidence** is used to compute the false positive, true positive and the area under the ROC curve.

```
For  winSize=0.5:0.5:6
    itr=itr+1;
    falsePositive=[]; truePositive=[]; AUC_value=0;
    timeStart;
    For numEvent=1: (M+N)
        Preprocessing:
```

```
            Load the original signal sigOrig;
            Locate the fall or nonfall event position $Pos_{event}$;
            Compute the window length: $winLen = winSize*fs$;
            Compute the boundaries:
            $lo\_b = \lfloor Pos_{fall} - winLen / 2 \rfloor$;
            $hi\_b = winLen + lo\_b - 1$;
            Take the signal segment out: $sigSeg = sigOrig(lo\_b : hi\_b)$;
        Extract features: featEvent(numEvent)=MFCC_featExtract(sigSeg);
    End
    Leave one out cross validation with the dataset featEvent,
        confidenceValue=leaveOneOut(featEvent, classifierType);
    Threshold the confidence to produce a ROC curve and compute AUC,
        [falsePositive, truePositive, AUC_value]
                    =thresholdConfidence(confidenceValue) ;
    timeEnd;
    AUC_array(itr)=AUC_value;
    timeCost(itr)=timeEnd-timeStart;
End
```

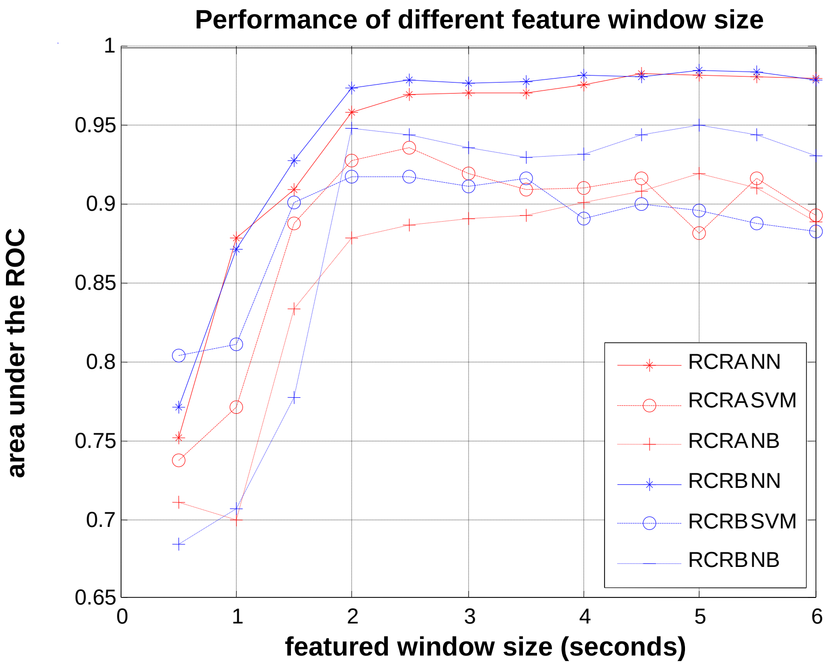


Figure 4.3 Classification performance with different window size.

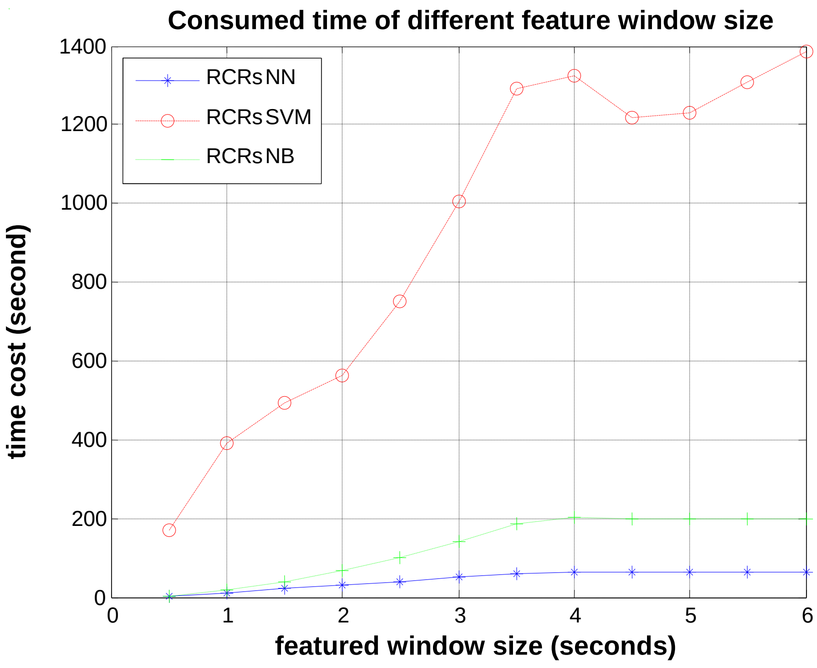


Figure 4.4 Time cost with different window size.

In Fig 4.3, the area under the ROC curve (AUC) is stable increasing before reaching the 2-second window size. The AUC dynamic range changes are insignificant if the window size keeps increasing. In Fig 4.4, the classification time at 2-seconds is much less than a larger window. Thus, the 2-second window has the best tradeoff between performance and processing time. MFCC features are extracted with a 2- second window from the signal in our algorithm.

4.5.2 Feature Window Position and Alignment

The feature window position has important influence in the detection performance. The floor RCR1 from lab Configuration I is used as the test dataset to illustrate this case. The time stamp of the ground truth is recorded by the webcam in the lab room.

We consider the window position cases as follows: 1) Use the time stamp window center as the center of the 2-second window; 2) use the signal peak as the center of the 2-second window; 3) use the signal peak as the start time for the 2-second window, and 4) use the signal peak as the end time for the 2-second window. More details are given in the Algorithm 4.3.

Algorithm 4.3 Finds the optimal signal window position.

Assumptions:

The effective fall signatures are related to the signal window locations for feature extraction.

Four locations are considered:

1) take the time stamp center as the center of the 2-s signal window;

---

2) take the peak location as the center of the 2-s signal window;

3) take the peak location as the start point of the 2-s window;

4) take the peak location as the end point of the 2-s window.

The original signal is represented with *sigOrig*;

The signal sampling frequency is *fs*;

The dataset has M fall and N non-fall events;

The function **MFCC_featExtract** is described in Section 4.2;

The function **leaveOneOut** represents leave one out cross validation (in Section 4.4) with a classifier (in Section 4.3);

The function **thresholdConfidence** is used to compute the false positive, true positive and the area under the ROC curve.

```
For  locNum=1:4
    falsePositive=[]; truePositive=[]; AUC_value=0;
    For numEvent=1: (M+N)
        Preprocessing:
        Load the original signal sigOrig;
        Compute the 2-s window length: winLen= 2*fs;
        Read the timestamp time for the current fall event: timestamp_start, timestamp_end;
        Compute the boundaries for the 2-s window:
        If locNum==1
            $winCenter = \left\lfloor \frac{(timeStamp\_start + timeStamp\_end)}{2} \right\rfloor$;
```

```
        $lo\_b = \lfloor winCenter - winLen / 2 \rfloor$;
        $hi\_b = winLen + lo\_b - 1$;
    End
    If locNum==2
        $Loc_{peak} = findpeakLoc(sigOrig)$
        $lo\_b = \lfloor Loc_{peak} - winLen / 2 \rfloor$;
        $hi\_b = winLen + lo\_b - 1$;
    End
    If locNum==1
        $Loc_{peak} = findpeakLoc(sigOrig)$
        $lo\_b = Loc_{peak}$;
        $hi\_b = winLen + lo\_b - 1$;
    End
    If locNum==4
        $Loc_{peak} = findpeakLoc(sigOrig)$
        $hi\_b = Loc_{peak}$;
        $lo\_b = hi\_b - winLen + 1$;
    End
    Take the signal segment out: $sigSeg = sigOrig(lo\_b : hi\_b)$;
    Extract features: featEvent(numEvent)=MFCC_featExtract(sigSeg);
End
```

```
    Leave one out cross validation with the dataset featEvent,
        confidenceValue=leaveOneOut(featEvent, classifierType);
    Threshold the confidence to produce a ROC curve and compute AUC,
        [falsePositive, truePositive, AUC_value]
                =thresholdConfidence(confidenceValue) ;
End
```

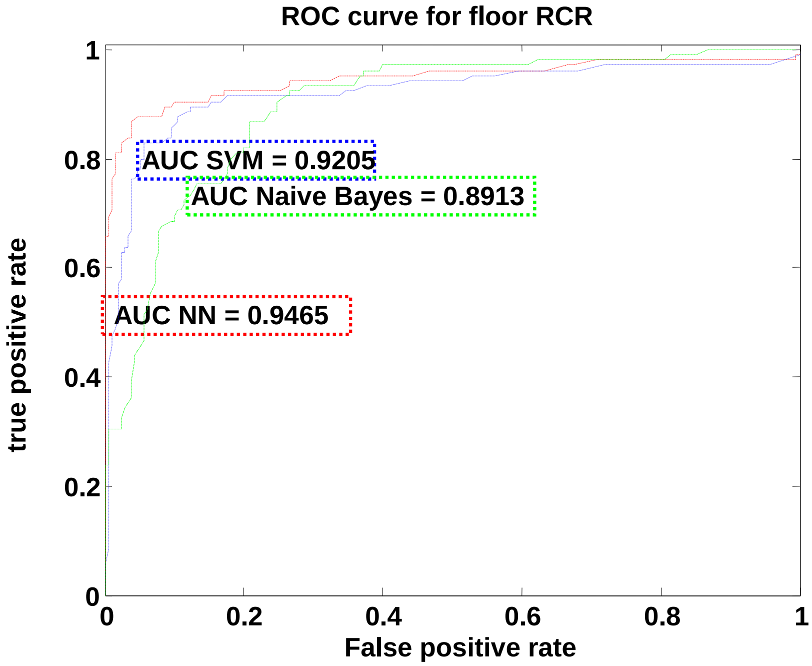


Figure 4.5 Case-1: Use the time stamp window center as the center of the 2-second window.

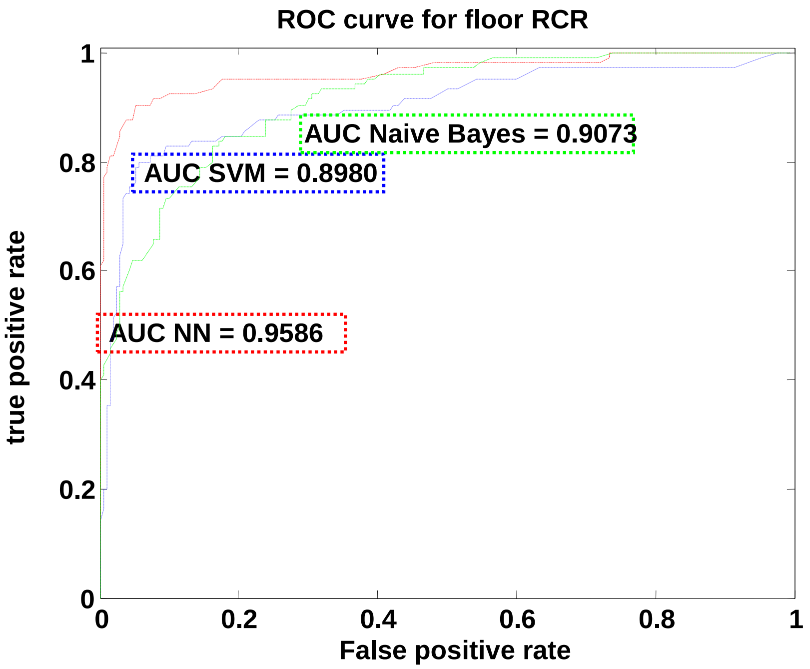


Figure 4.6 Case-2: Use the signal peak as the center of the 2-second window.

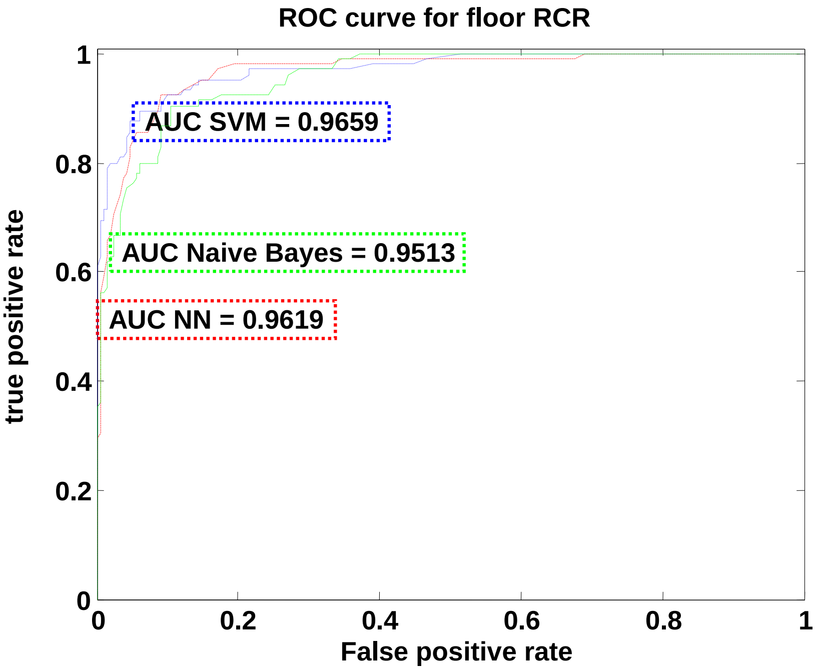


Figure 4.7 Case-3: Use the signal peak as the start time for the 2-second window.

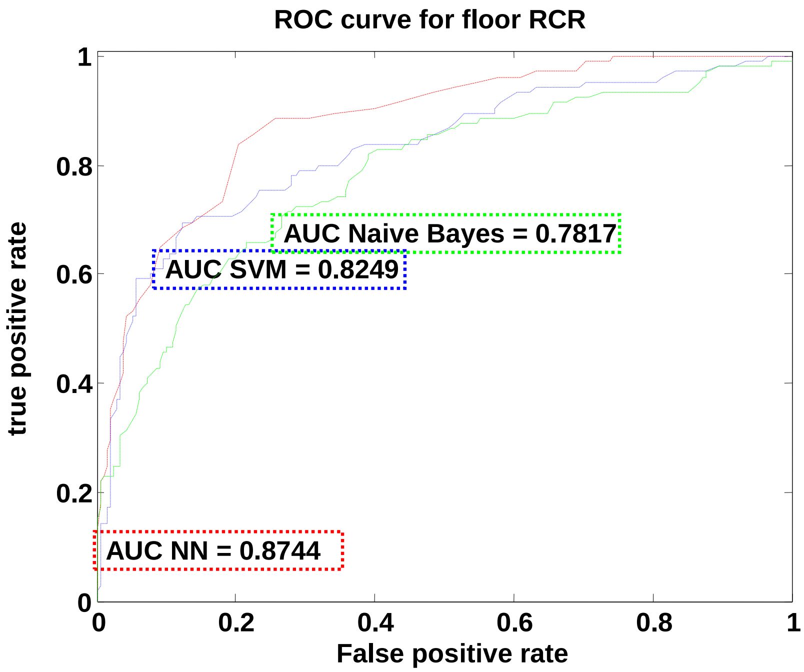


Figure 4.8 Case-4: Use the signal peak as the end time for the 2-second window.

Table 4.1 AUC value for each feature window position

Case #	NN	Naïve Bayes	SVM
1	0.848	0.881	0.821
2	0.959	0.907	0.898
3	0.962	0.951	0.966
4	0.874	0.782	0.825

Comparing the results of these four cases, we find that Case 2 and 3 have the best performance, as shown in Table 4.1. Based on this experiment, we learn that the feature window position is important in detection. We also noticed that the ground truth may have some bias since it is observed by eyes from the video. Therefore, feature alignment is necessary.

For each located potential fall, the located point from the energy burst curve is taken as the center of the feature window. For feature alignment purpose, features are extracted with a 2-second window and a 6-second window. The reference features are extracted from a lab fall, in which the subject was performing a forward fall just under the ceiling radar. We select the reference features with the length of 2-s window features, shown in Figure 4.9.

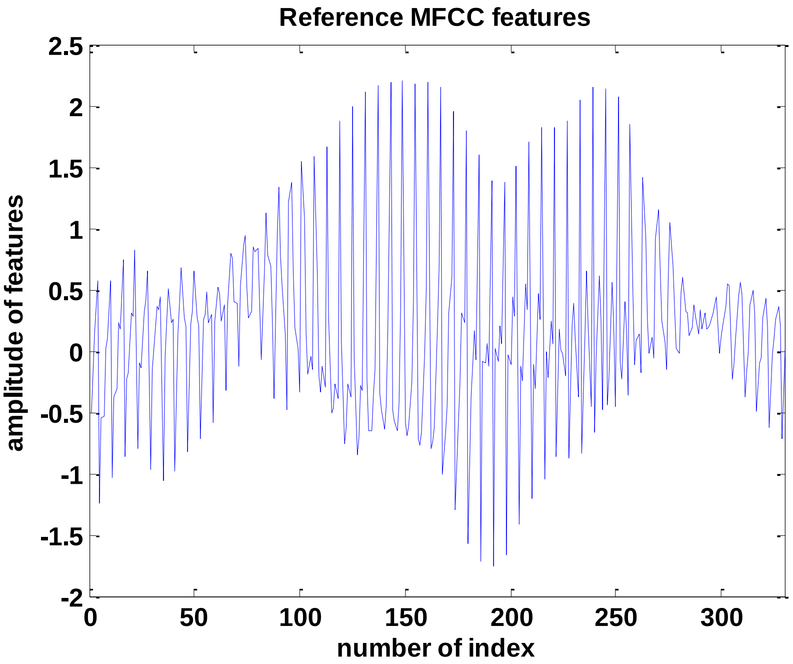


Figure 4.9 Block diagram for the processing procedure to detect a fall.

The length of the feature is 330. The feature shift length is calculated by locating the peak on the correlation between the current extracted 2-s window features with the reference features. By moving the 2-s window in the 6-s window with the shift length, we obtained the aligned features which have the highest similarity with the reference in the neighborhood of the potential fall. More details are presented in Algorithm 4.4.

Algorithm 4.4 Feature alignment.

Assumptions:

The effective fall signatures are related to the signal window locations for feature extraction.

The reference feature is denoted by *featRef*;

The point that needs to be shift for alignment is *alignDist*;

The original signal is represented with *sigOrig*;

The signal sampling frequency is *fs*;

The dataset has M fall and N nonfall events;

The function **MFCC_featExtract** is described in Section 4.2;

The function **featAlign** is described in Section 4.5.2;

The function **correlatePos** is used to compute the distance of the correlation peak;

The function **shiftPos** is used to find the aligned feature in *winLen_6s*;

The function **leaveOneOut** represents leave one out cross validation (in Section 4.4) with a classifier (in Section 4.3);

The function **thresholdConfidence** is used to compute the false positive, true positive and the area under the ROC curve.

```
falsePositive=[]; truePositive=[]; AUC_value=0;
For numEvent=1: (M+N)
        Preprocessing:
                Load the original signal sigOrig;
                Compute the 2-s window length: winLen_2s= 2*fs;
                Compute the 6-s window length: winLen_6s= 6*fs;
```

```
        Find the peak location on the original signal;
                $Loc_{peak} = findpeakLoc(sigOrig)$;
        Compute the boundaries for the 2-s window:
                $lo\_b_{2-s} = \lfloor Loc_{peak} - winLen\_2s / 2 \rfloor$;
                $hi\_b_{2-s} = winLen\_2s + lo\_b_{2-s} - 1$.
        Compute the boundaries for the 6-s window:
                $lo\_b_{6-s} = \lfloor Loc_{peak} - winLen\_6s / 2 \rfloor$;
                $hi\_b_{6-s} = winLen\_6s + lo\_b_{6-s} - 1$.
        Take the signal segment out:
                $sigSeg_{2-s} = sigOrig(lo\_b_{2-s} : hi\_b_{2-s})$;
                $sigSeg_{6-s} = sigOrig(lo\_b_{6-s} : hi\_b_{6-s})$.
    Extract features:
            featEvent_2s(numEvent)=MFCC_featExtract( $sigSeg_{2-s}$ );
            featEvent_6s(numEvent)=MFCC_featExtract( $sigSeg_{6-s}$ ).
    Feature alignment: correlation with the reference
        alignDist=correlatePos(featEvent_2s(numEvent), featRef);
        featEvent_aligned=shiftPos(featEvent_6s(numEvent), alignDist);
End
Leave one out cross validation with the dataset featEvent,
    confidenceValue=leaveOneOut(featEvent_aligned, classifierType);
Threshold the confidence to produce a ROC curve and compute AUC,
```

[falsePositive, truePositive, AUC_value]

=**thresholdConfidence**(confidenceValue) .

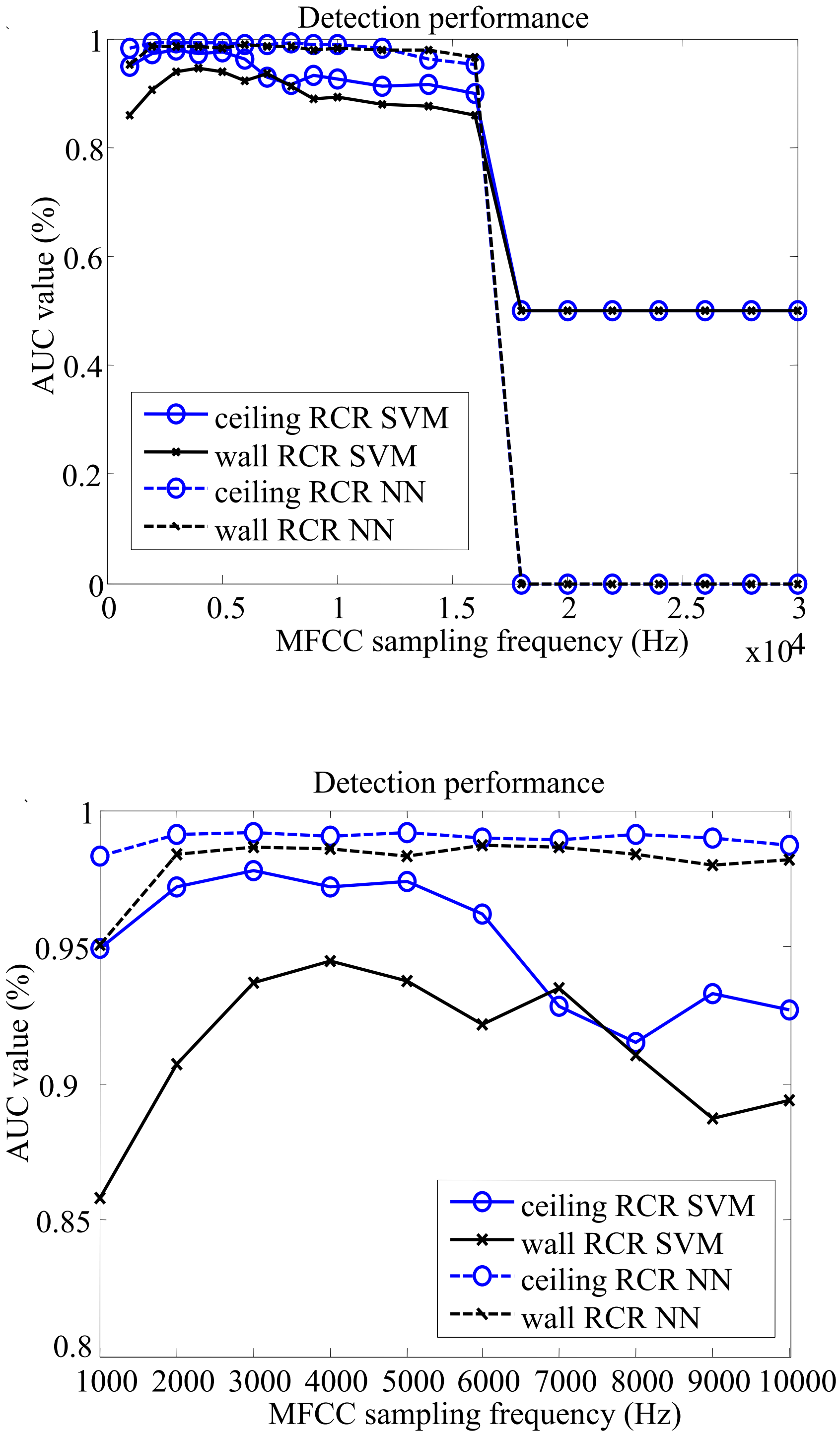
4.5.3 MFCC Feature Parameter Selection

To seek the best parameters in MFCC feature extraction, we designed three groups of experiments: the first group shows the best parameters set with and without feature alignment; the second group compares the best parameters set between the unaligned and aligned features with different alignment methods; the third group presents the best parameters set for unaligned and aligned features with a different classifier.

The first two groups used the CW_SYS dataset. Two classifiers, NN and SVM, are applied for the first two groups since those two classifiers generate the best performance in Case 3 of Table 4.1. The third group uses the floor radar data in CF_SYS with HMM.

4.5.3.1 Fall Detection using NN/SVM for Unaligned Features

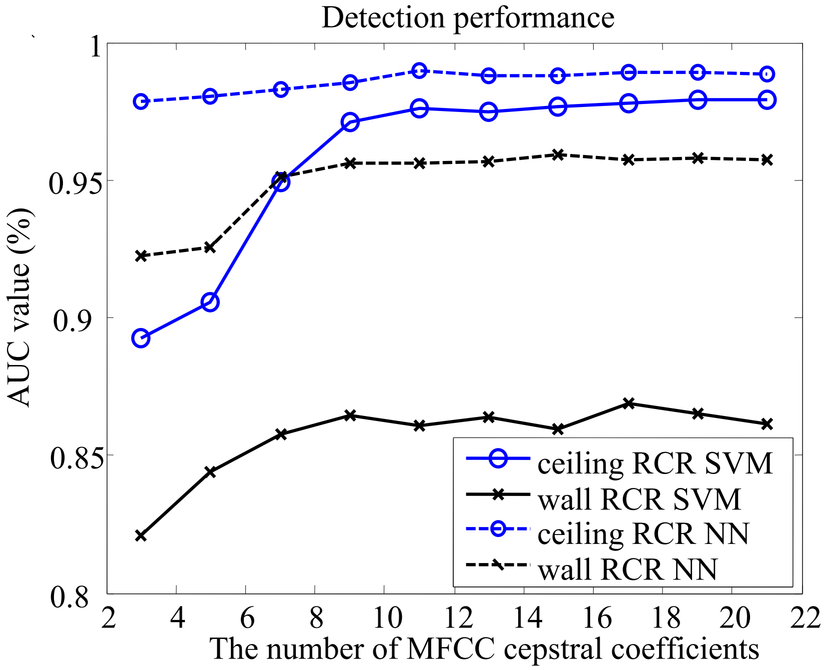
For the first group of experiments, the CW_SYS dataset is used for three experiments: 1) use different MFCC frequencies; 2) use different numbers of MFCC coefficients; 3) use different numbers of MFCC frame rates.



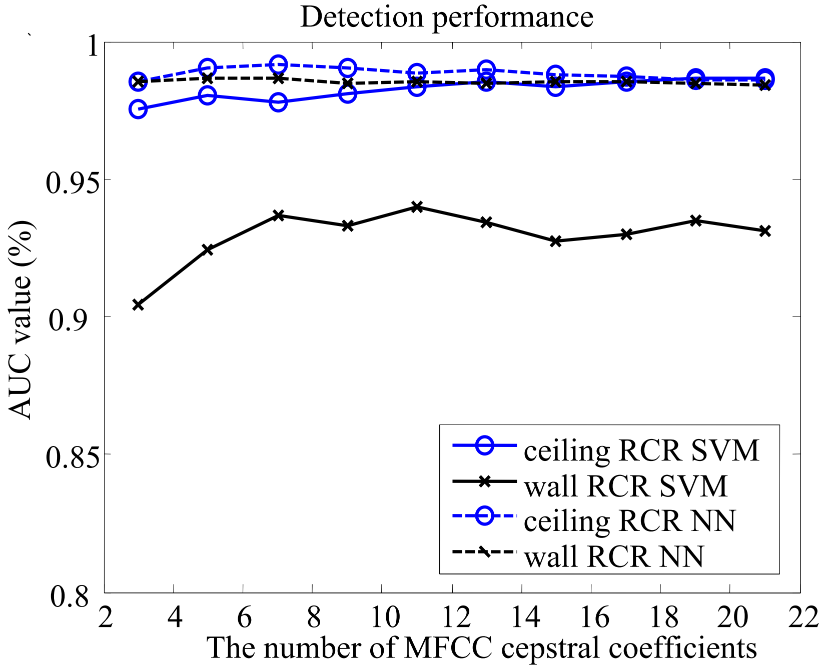
(a)

(b)

Figure 4.10 Case-1: Use different MFCC frequency. (a) Overview; (b) Detailed look.



(a)



(b)

Figure 4.11 Case-2: Use different numbers of MFCC coefficients. (a) The MFCC sampling frequency is 1 kHz; (b) The MFCC sampling frequency is 3 kHz.

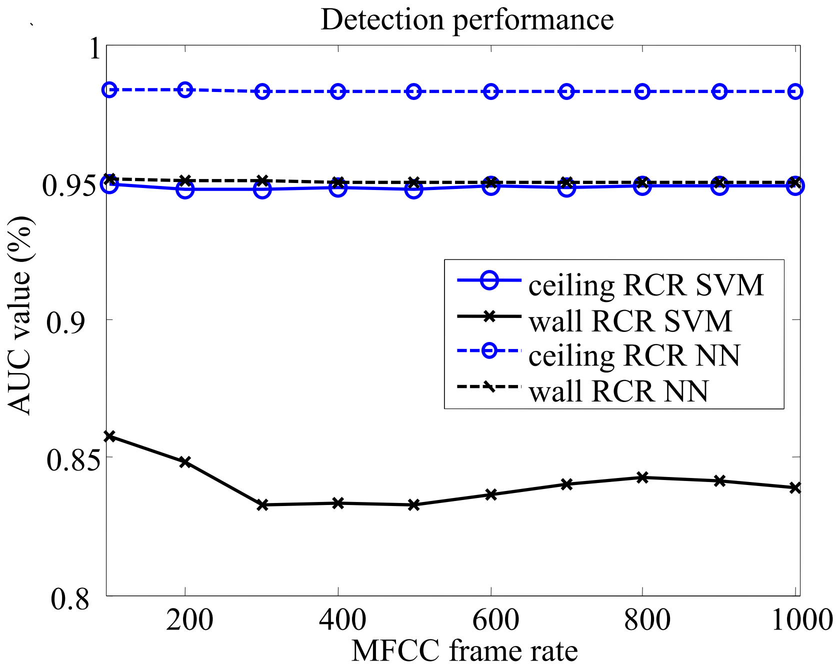


Figure 4.12 Case-3: Use different numbers of MFCC frame rates.

We have the following observations. The best MFCC sampling frequency is 3 kHz when the coefficient number is 7 and frame rate is 100 in Figure 4.10. In Figure 4.11, the best MFCC coefficient number is 17 with 1 kHz MFCC sampling frequency. The best MFCC coefficient number is 7 with 3 kHz MFCC sampling frequency. The results with 3 kHz are better than the results with 1 kHz for both of the radar sensors and classifier combinations. Also from the computational view, 7 coefficients are more efficient than 17 coefficients. By setting the 7 coefficients and 3 kHz sampling frequency, the best performance for MFCC frame rate is 100 in Figure 4.12. So for the unaligned features, the best parameters are 3 kHz sampling frequency; 7 MFCC coefficients, and 100 MFCC frame rates.

4.5.3.2 Fall Detection using NN/SVM for Aligned Features with Different Methods

In the second group of experiments, the unaligned and aligned features with different methods from CW_SYS dataset are tested with SVM classifier in Figure 4.13 and NN classifier in Figure 4.14, respectively.

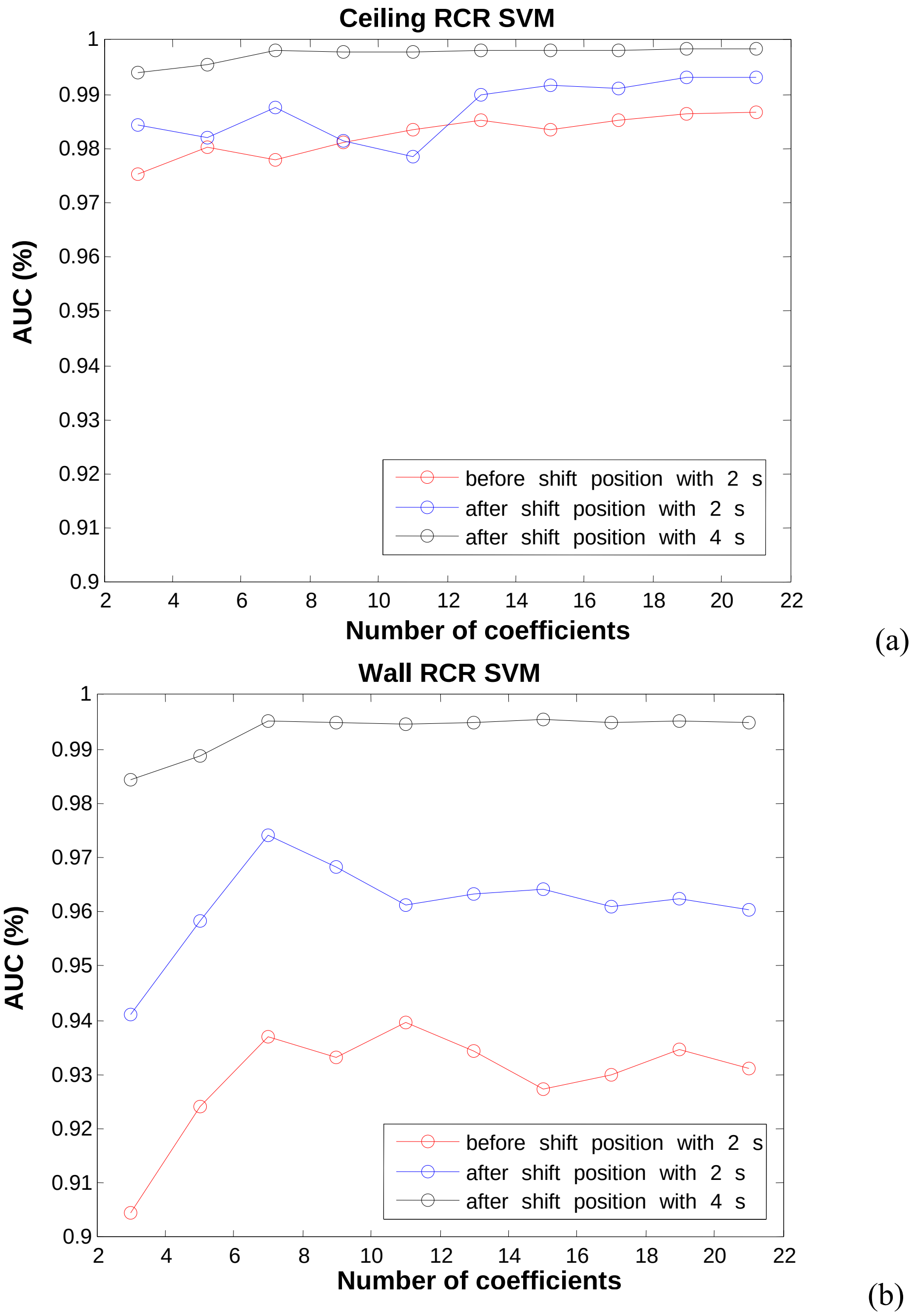


Figure 4.13 Case-1: Performance with SVM using different MFCC coefficients at mfccfs=3 kHz. (a) Ceiling RCR; (b) Wall RCR.

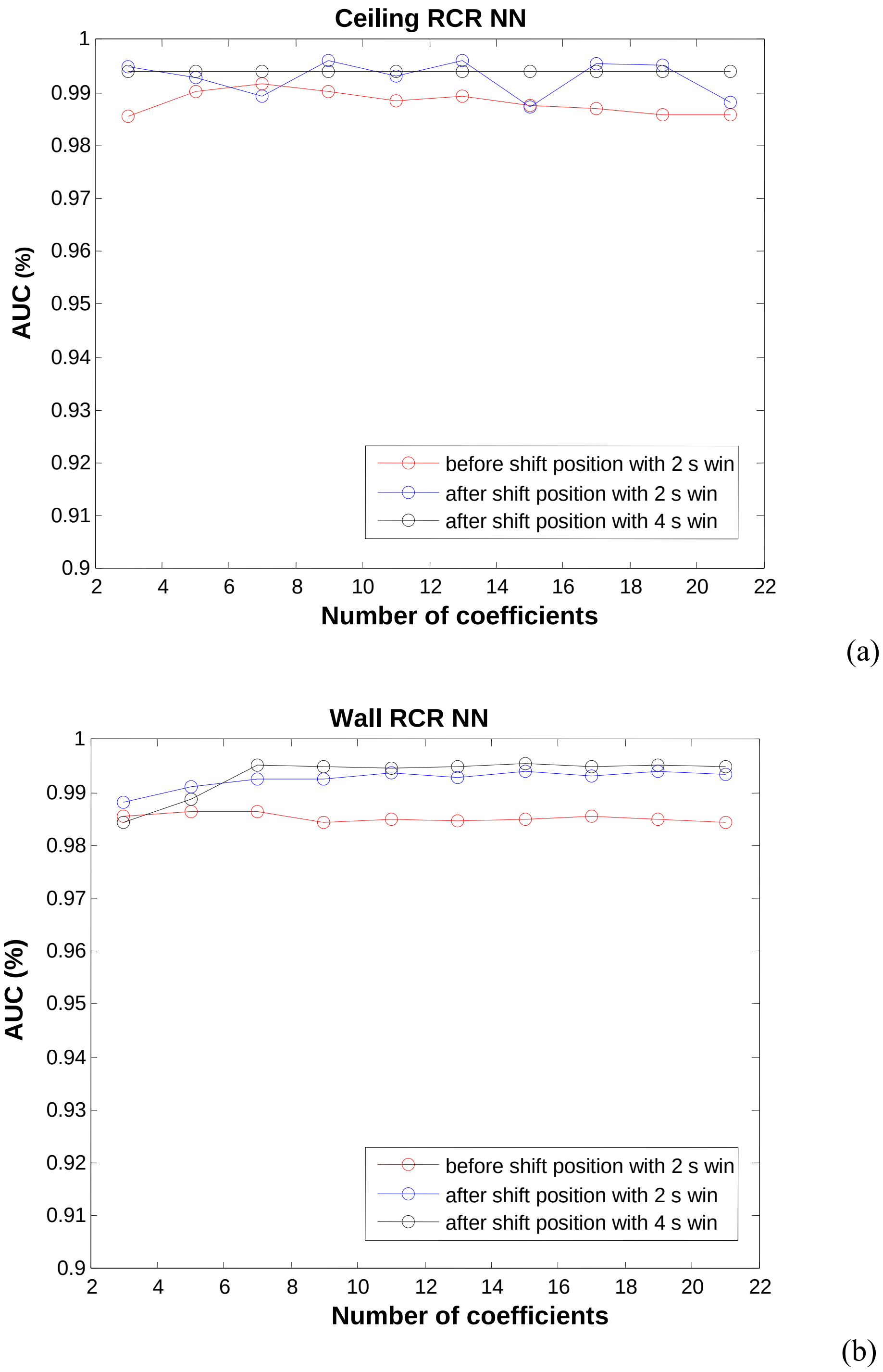


Figure 4.14 Case-2: Performance with NN using different coefficients at mfccfs=3kHz. (a) Ceiling RCR; (b) Wall RCR.

In this experiment, two alignment methods are considered. The first one is to shift the feature position by adding zeroes. This is denoted as “shift position with 2-s win.”

The second one is to shift the features in a larger window. This is called a “shift position with 4-s win.” The shift position is located by the correlation peak.

We find that the performance with the aligned features is better than the unaligned features for both classifiers with 7 MFCC coefficients. 7 MFCC coefficients are good for the performance with the aligned features. The alignment of “shift position with 4-s win” performs better than the alignment of “shift position with 2-s win.” This is true since more signal information is conserved in the 4-s window. For more comprehensive situations, the 2-s window is shifted within a 6-s window for the alignment.

The above two groups of experiments focus on the parameter selection for feature extraction. In the next section, we will investigate the parameters with HMM on the floor radar from Configuration III in lab.

4.5.3.3 Fall Detection using HMM for Unaligned and Aligned Features

In the experiments of the third group, HMM was applied on the dataset of floor radar. We used leave one out cross validation with H2M to evaluate the detection performance for the floor radar in lab. We selected left-right model with N=3 states for fall model. The initial parameters for both models are in Table 4.2.

To explain the problem with a 2-second signal segment, we can extract 6 dimensional MFCC features for each of the 55 sequential windows to represent the motion signatures. Using the lab data as the example, we have K=466 sequences. The sequence length is T=55.

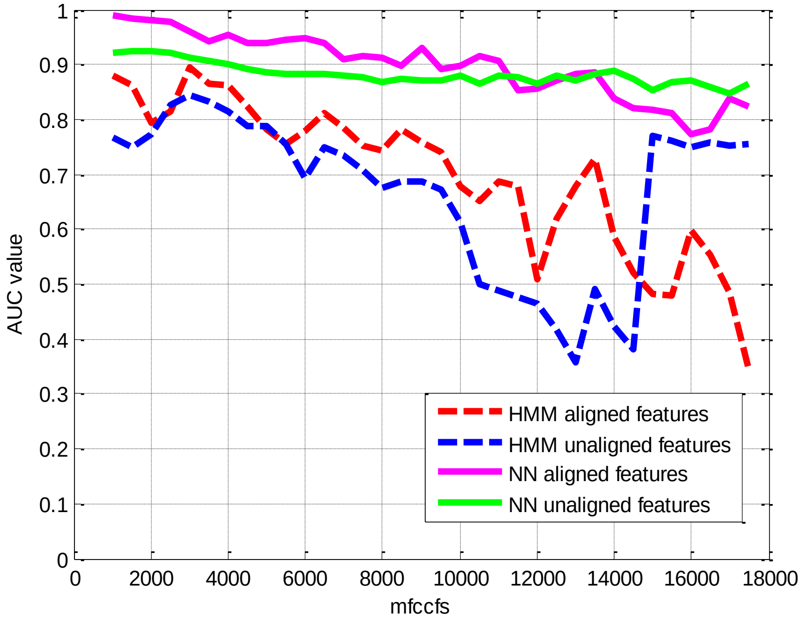
For an example, we take one fall as an unknown sample. In the training progress, we obtain the updated A, and μ,Σ with 20 iterations of EM steps using the remaining 465 labeled data. Thus, we get a learned fall model λ_f and a non-fall model λ_{nf}, which is represented by a set of parameters {π, A, μ,Σ}.

In order to find the best parameters with the signal, we did two experiments: 1) changed the sequence length by varying the MFCC sampling frequency; 2) changed the feature dimension by varying the number of MFCC coefficients.

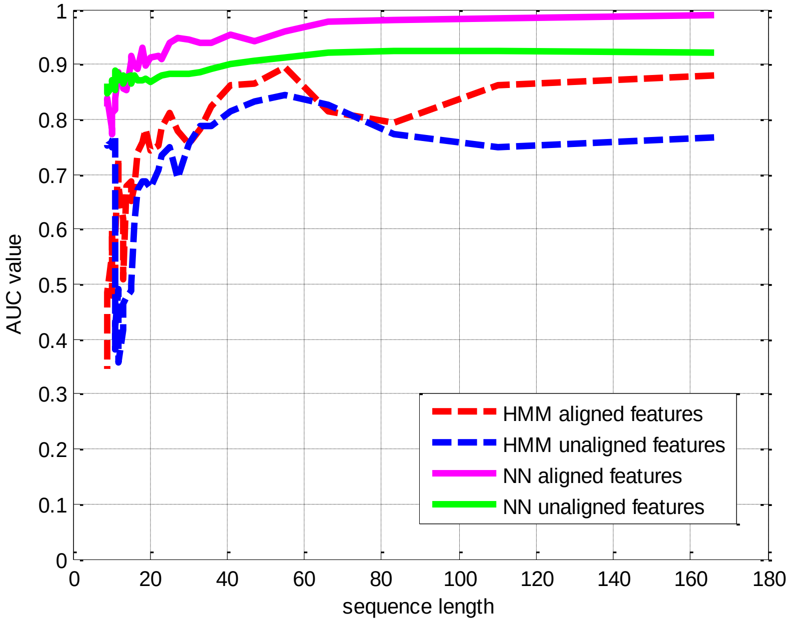
Table 4.2 Initial parameters and one example of trained models for HMM

	Fall model	Non-fall model
Initial	A0= 0.5 0.3 0.2 ; 0 0.9 0.1; 0 0 1 pi0= 1 0 0 Mu0= 0.2ones(3,6) Sigma0= 0.8ones(3,6)	A0= 0.6 0.3 0.1 0.2 0.55 0.25 0.05 0.1 0.85 pi0= 0.65 0.15 0.2 Mu0= 0.2ones(3,6) Sigma0= 0.8ones(3,6)
Coefs (7) mfccfs (3kHz)	A_f = 0.9200 0.0800 0.0000 0 0.9714 0.0286 0 0 1.0000 pi0_f = 1 0 0 mu_f = -0.0392 0.0609 0.1933 0.1923 0.7690 0.0802 -0.1486 -0.3614 -0.0155 0.7063 0.7215 0.0230 -0.0325 -0.0442 -0.0184 0.3497 0.1982 -0.0720 Sigma_f = 0.0400 0.0297 0.0496 0.0457 0.0845 0.1465 0.1665 0.0777 0.0553 0.1769 0.1174 0.2080 0.1820 0.1543 0.2033 0.4263 0.4969 0.2903	A_n = 0.7807 0.1991 0.0202 0.2910 0.5559 0.1531 0.0123 0.0391 0.9486 pi0_n = 0.9269 0.0717 0.0014 mu_n = 0.0370 0.0742 0.1891 0.2335 -0.6497 0.0057 0.0453 0.0816 0.2108 0.3448 -0.2925 0.0480 -0.0169 -0.0074 0.2021 0.8244 0.3997 0.1095 Sigma_n = 0.0965 0.0761 0.1621 0.1008 0.2830 0.1588 0.1076 0.0853 0.2346 0.1519 0.2770 0.1566 0.1304 0.1206 0.2985 0.1838 0.1543 0.1449

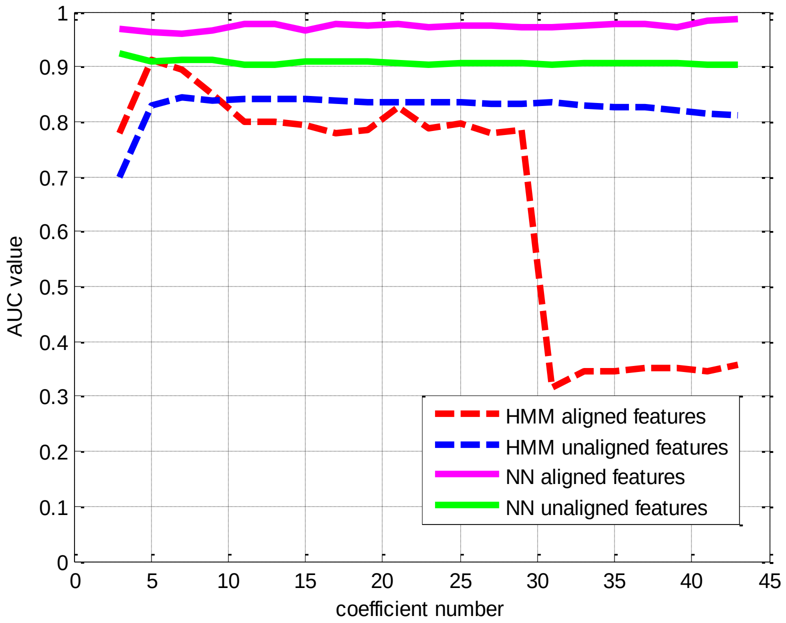
Using the feature dimension as 6, the sequence length was calculated by changing MFCC sampling frequency mfccfs. It is from 1 kHz to 17.5 kHz with an interval of 500 Hz from left to the right in Figure 4.15(a). For HMM the best performance happens when the sequence length is 55 (in Figure 4.15(b)), which is with 3 kHz MFCC sampling frequency. Then if we keep the sequence length at 55 and change the coefficient number, we find the best performance is with 7 coefficients in Figure 4.15(b) for HMM. So the best parameters are still 3 kHz mfccfs with7 coefficients. Performance with NN is also presented here as a comparison. For NN, the longer sequence number helps to improve the performance in Figure 4.15(b). The best parameters for NN are not as clear as the parameters for HMM.



(a)



(b)



(c)

Figure 4.15 Floor RCR performance with HMM and NN. (a) The results with different mfcc sampling frequency. (b) The results with different sequence length. (c) The results with different coefficient number.

4.6 Summary

In this chapter, the algorithm for the fall detection system was designed. The preprocessing method was proposed at first. The feature extraction method was described. Several classification methods were introduced and applied to fall detection. The evaluation criteria were presented as designed to evaluate the performance. The parameters for the feature extraction are presented as investigated for ceiling, wall, and floor radar sensors with classifiers.

Chapter 5 Fall Detection Results

In this chapter, we will present the detection results for both lab data and real senior apartment data described in Chapter 3 using the fall detection algorithm presented in Chapter 4.

5.1 Fall Detection with Lab Dataset

5.1.1 Results with Configuration I

The FF_SYS dataset includes 90 falls and 341 non-falls. In order to see the performance of this configuration, we also added another 19 falls. Thus we have 109 falls and 341 nonfalls. For each sensor, the performance is shown with ROCs obtained with the two classifiers SVM and NN. Figure 5.1 and Figure 5.2 show the results for RCR1 and RCR 2, respectively.

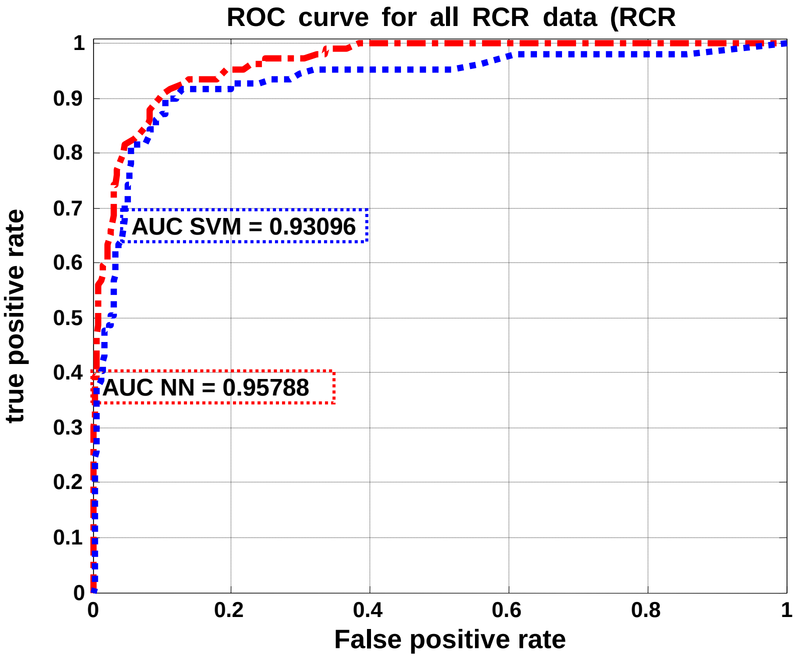


Figure 5.1 Performance results for RCR1.

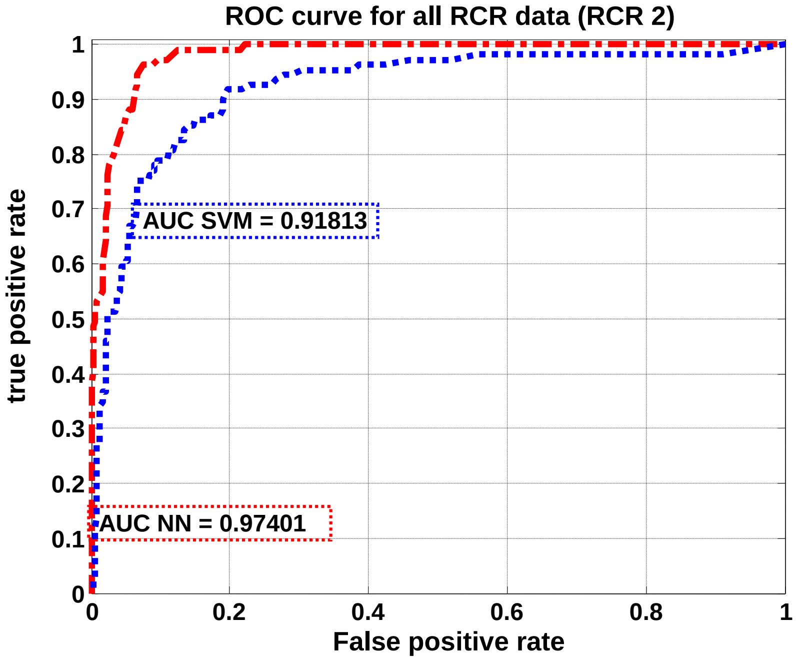


Figure 5.2 Performance results for RCA 2.

From the above results, we see that NN produced the best results for both sensors, with an area under the curve (AUC) of about 0.97. SVM performance was close, with an AUC of about 0.94.

By analyzing the false alarms of both classification algorithms, we found some examples of squatting and kneeling. A possible reason for this situation is that while the actor and the students were instructed how to walk and fall as an older adult, they were not told to bend and kneel in the same fashion. As a result, the speed of these actions was possibly close to that of a fall. These results were also recorded in a lab instead of the Tiger Place where the stunt actor eventually performed only the falls but not the false alarms.

5.1.2 Results with Configuration II

With the CW_SYS dataset, we ran the fall detection for the ceiling and wall RCRs, respectively. The ceiling RCR had 126 falls and 335 non-falls. The wall RCR has 126 falls and 353 non-falls. We show the detection results with SVM and NN classifiers in Figure 5.3. We noticed that the best detection happens for the ceiling RCR with the AUC 0.9990 using NN classifier. The best performance for the wall RCR was AUC 0.9750 with SVM classifier.

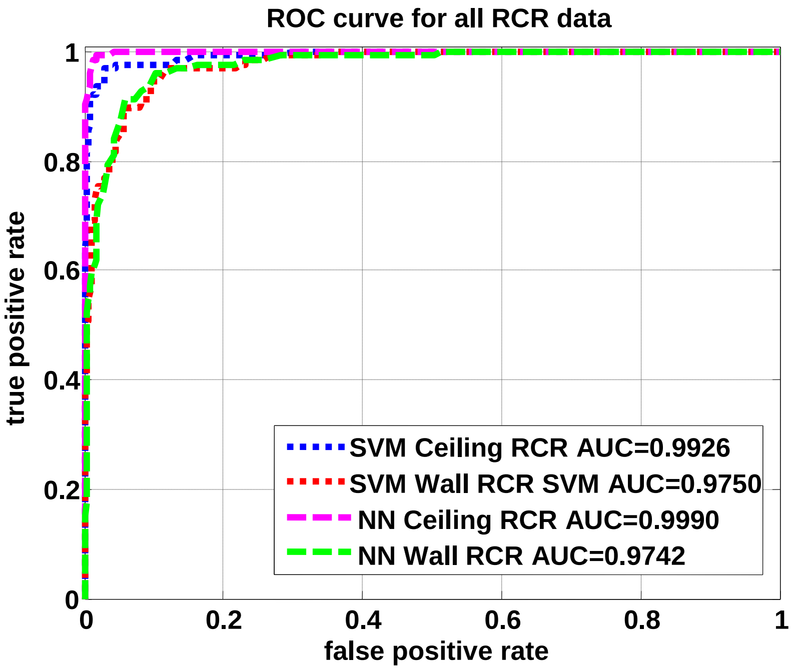


Figure 5.3 Fall detection in lab with CW_SYS.

5.1.3 Results with Configuration III

The CF_SYS has 105 falls for both ceiling and floor RCRs. The ceiling RCR recorded 349 non-falls and the floor RCR recorded 361 non-falls.

Performance of fall detection with CF_SYS is shown in Figure 5.4, the ceiling RCR almost detects all the falls at the false positive rate of 20%. By contrast, the floor

RCR can detect all the fall when the false positive rate (FPR) is above 80%. At zero FPR, the ceiling RCR had a 10% detection improvement compared to the floor RCR.

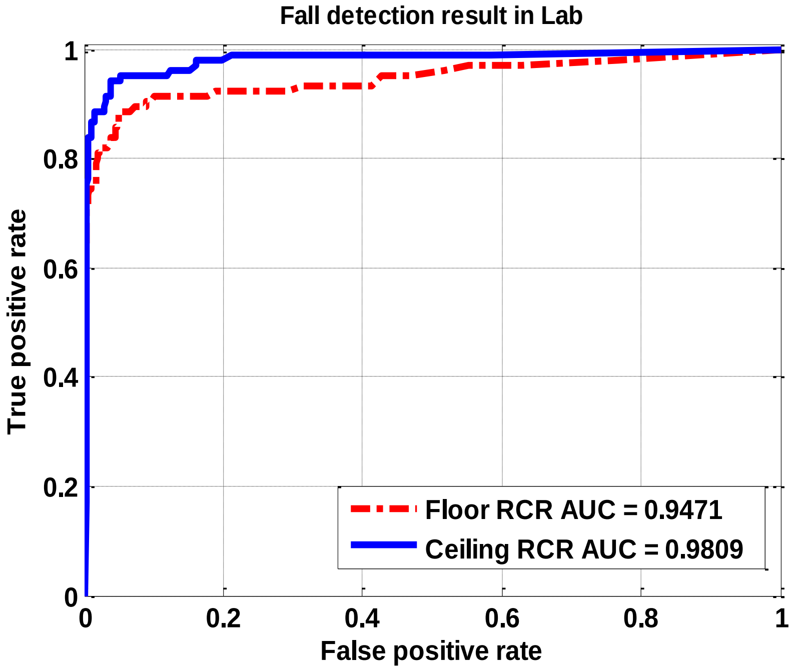


Figure 5.4 Fall detection in lab with CF_SYS using SVM.

5.2 Fall Detection with Real Apartment Dataset

By assigning the same detection task in senior apartments, we test the different training dataset with CF_SYS and DATA_TP described in Chapter 3.

Figure 5.5 shows that the fall detector trained with DATA_TP gives the larger AUC than detector trained with CF_SYS for both RCRs. For the solid curves, the ceiling RCR has a straight 3% increase in AUC. For the floor RCR with the dashed curves, the AUC of the apartment trained detector still improved by 2%. However, this improvement is more complex. The true positive rate (TPR) of the apartment trained detector was lower before the false positive rate (FPR) achieved 17%. The detector trained with DATA_TP only improved the detection rate with the false alarm rate above 17%.

Considering the small size of DATA_TP, we combined the DATA_Lab and DATA_TP together as the training dataset. The ROC curves in Figure 5.6 are similar to those in Figure 5.4. We notice that the fall number in CF_SYS is 5.25 times the fall number in DATA_TP for both RCRs. The non-fall number in CF_SYS is 10.91 times more than the number in DATA_TP for ceiling RCR and 12.44 times more for the floor RCR, respectively. As a result, the performance of the combined dataset was close to the single CF_SYS due to the dominant role of CF_SYS. However, a slightly better detection rate was achieved by including the DATA_TP when the ROC curve reached 16% FPR for ceiling RCR and 60% FPR for floor RCR.

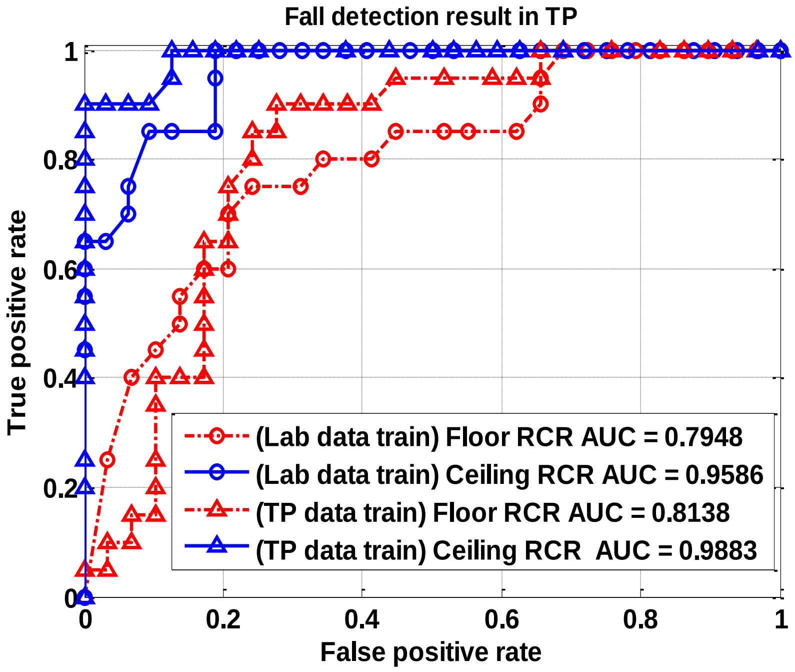


Figure 5.5 Fall detection in senior apartment environment with DATA_TP by using different training datasets, CF_SYS and DATA_TP, respectively.

With Equations (4.39)-(4.42), we calculated the statistical measures to the detection results on each operating point of a ROC curve for above experiments as shown in Figure 5.7. The x-axis experiment number corresponds to the experiments of fall detection in the lab (Figure 5.4), in apartment with the detector trained by CF_SYS (circle marker lines in Figure 5.5), in apartment with the detector trained by DATA_TP (triangle marker lines in Figure 5.5), and in combined dataset (Figure 5.6), respectively. The second and the third experiments have the same detection task. We notice that the false negative increases for ceiling RCR since SEN dropped from 100% to 90%. The false positive increased for floor RCR because SPC decreased from 79.31% to 72.41%. All other measures improved after the detector training set was switched from CF_SYS to DATA_TP.

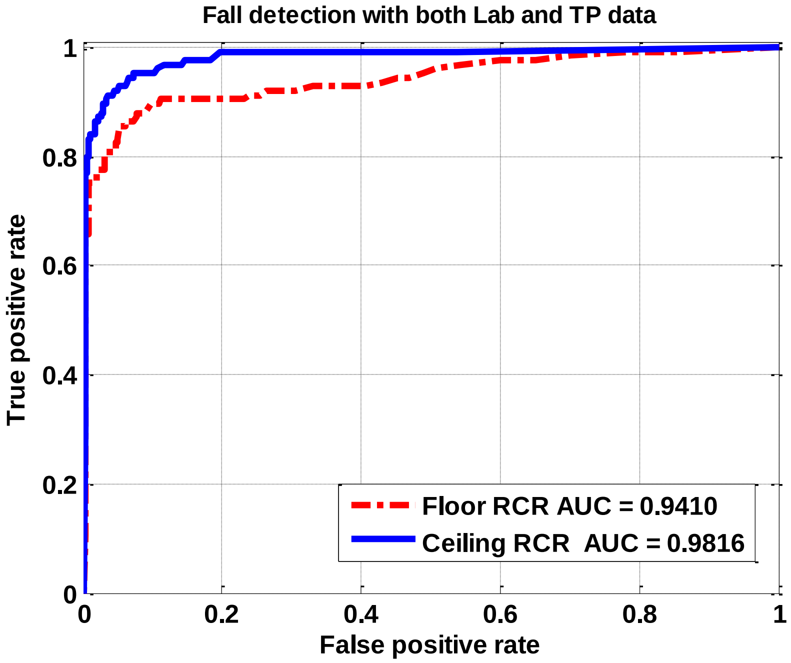


Figure 5.6 Fall detection with CF_SYS and DATA_TP together.

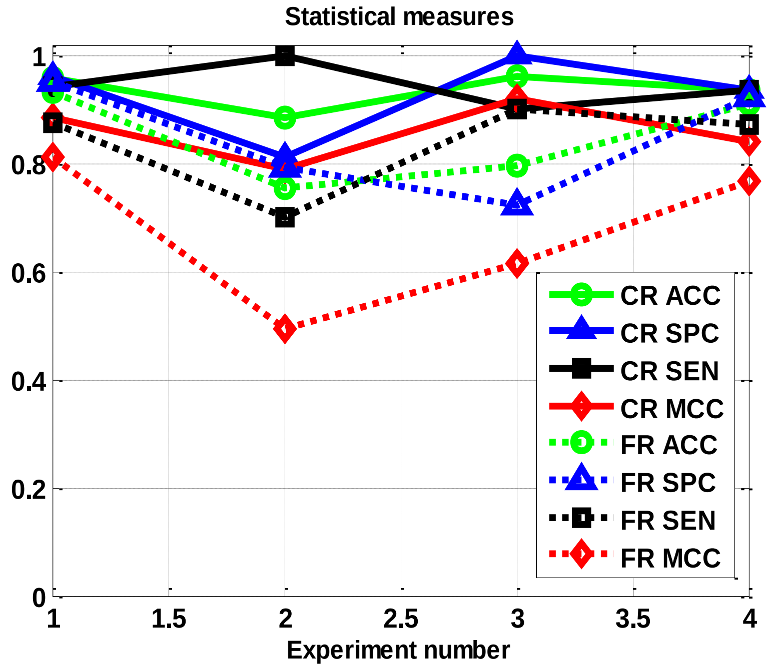


Figure 5.7 Statistical measures for each experiment conducted in lab. CR denotes ceiling RCR, and FR represents floor RCR. Those measures are described in Equations (4.39)-(4.42).

We find that the fall detection of ceiling RCR is always better than the floor RCRs in Figure 5.4-5.6. The best performance is achieved when the testing and training data are from the same environment as presented by Figures 5.4 and 5.5. Considering the limited data collected in Tiger Place apartment, we combined both CF_SYS and DATA_TP as the training data.

We ran the fall detection with the floor RCR in each of the six senior apartments. We used the fall detectors trained by CF_SYS and DATA_TP to detect falls with the two weeks data for each floor RCR. Figure 5.8 shows the number of false alarms per hour for each RCR in each apartment. The 2 weeks data included the day that the stunt actor performed several falls near the floor radar. So there were some falls in each apartment during those two weeks as well as the opportunity for false alarms.

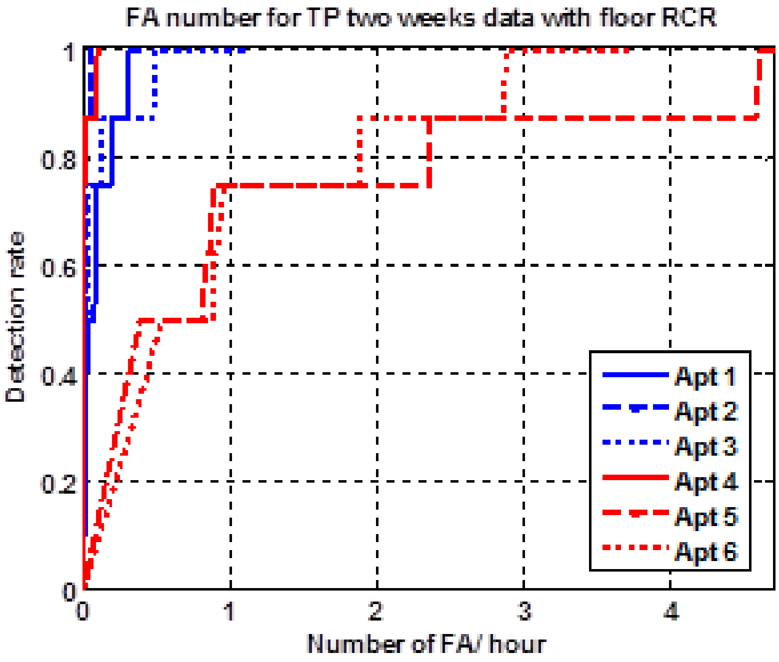


Figure 5.8 Fall detection and false alarm rate of floor RCR with two weeks of data from senior apartment at six different apartments.

We learned in the lab (Figure 5.7) that locating the radar on the floor is not the optimal location. For the real-world senior apartment data, we could better understand the limitation of the floor radar as displayed in Figure 5.8. The falls can be detected with a good degree of accuracy, but with a relatively high false alarm rate when using the radar at floor level.

The signal strength of floor RCR varied in each apartment because the RCR antenna was shielded with slightly different foil sizes, which was adjusted according to the specific apartment surrounding settings, such as room size, furniture and building materials. In Figure 5.8, more false alarms were detected in both apartments 5 and 6. This could be due to more daily activities in Apartment 5 since a couple lived there. The female resident was diagnosed with Parkinson’s and used a walker a lot in the room. In

Apartment 6, the female resident had severe osteoarthritis and she was always in a slumped position during walking. Rehabilitation or nursing staff visits to her apartment generated more activities.

Next, we installed a ceiling RCR above several rooms, particularly the bathroom where privacy is a big concern, to validate the consistency of fall detection performance with ceiling RCRs. Meanwhile, we collected more data from senior apartments to enlarge our dataset. It should be noted that only data points or silhouettes were obtained during data collection, which protected the identity and privacy of the person being monitored.

5.3 Summary

This chapter has presented the fall detection results on three different configurations in the MU elder studies research lab and continuous real-world data were collected from six different senior apartments. The data collection details were introduced in Chapter 3 with methodologies described in Chapter 4.

Chapter 6 Data Fusion

We notice that different classifiers may produce different decisions for the same radar signatures. In this chapter, we propose a fusion methodology based on the Choquet integral that combines partial decision information from each sensor and each classifier to form a final fall/non-fall decision. We employ Mel-frequency cepstral coefficients (MFCC) to represent the Doppler signatures of various human activities such as walking, bending down, and falling. Then we use three different classifiers, NN, SVM and Bayes, to detect falls based on the extracted MFCC features. Each partial decision from a classifier was represented as a fall confidence. We applied our fusion method to the lab Configuration I dataset made up of 450 activity samples (109 falls and 341 non-falls).

We realized that a data fusion of the results from different classifiers of each RCR is a key issue in making a final decision for the fall detection system. Hence, fuzzy integrals and fuzzy logic measure were implemented to fuse these multiple information sources.

6.1 Fuzzy Integration

Choquet fuzzy integral has been widely used in multiple source fusion cases [21]. Fuzzy measure is a representation of uncertainty. This is different from the assignment of membership grades in fuzzy sets. Membership grades represent the degree of membership in a particular set with unsharp boundaries for a value, which is assigned to each element of the universal set. Fuzzy measure represents the degree of evidence or belief that a particular element belongs to this set value by assigning a value to each crisp image of the universal set. Let $X = \{x_1, x_2, ..., x_n\}$ be any finite set and $\lambda > -1$. **Sugeno λ measure** is defined by the function *g*: $2^X \rightarrow [0,1]$ which have the following properties:

$g(\Phi) = 0$ and $g(X) = 1$;

if $A, B \subseteq X$ with $A \cap B \subseteq \Phi$, then

$$g(A \cup B) = g(A) + g(B) + \lambda g(A) g(B) \cdot \qquad (6.1)$$

A set function satisfying the above conditions is a fuzzy measure. **Fuzzy density** can be defined by $g^i = g(\{x_i\})$.

The value of λ satisfies

$$\lambda + 1 = \prod_{i=1}^{n} (1 + \lambda g^i), \qquad (6.2)$$

where $\lambda \neq 0$ and $\lambda > -1$.

To specify a Sugeno λ measure on a set *X* with *n* elements only requires *n* different densities, thus the number of free parameters is reduced from $2^n - 2$ to *n*.

Fuzzy Choquet integral was used to fuse *n* different information sources from a discrete fuzzy set *X*. Let *h* be a function from *X* to [0, 1]. Let $\{x_{(1)}, x_{(2)}, ..., x_{(n)}\}$ represent the reordering of the set *X* such that $h(x_{(1)}) \geq h(x_{(2)}) \geq ... \geq h(x_{(n)})$. Hence, the Choquet integral of *h* with respect to a fuzzy measure g on set *X* is defined by

$$\int_C hog = \sum_{i=1}^{n} h(x_{(i)})[g({}^{\alpha_i}H) - g({}^{\alpha_{i-1}}H)], \qquad (6.3)$$

where ${}^{\alpha_i}H = \{x_{(1)}, ..., x_{(i)}\}$ and $g({}^{\alpha_0}H) = 0$.

Another general form of Equation (6.1) is

$$g(\{{}^{\alpha_i}H\}) = g^i + g(\{{}^{\alpha_{i-1}}H\}) + \lambda g^i g(\{{}^{\alpha_{i-1}}H\}). \qquad (6.4)$$

6.2 Fusion System of Multiple Information Sources

The dataset used for this project has 450 samples including 109 falls and 341 non-falls. Two RCRs were recoding data at the same time. MFCC features of the i-th sample were denoted with $\mathrm{S_i} = [\mathrm{S_{i1}},\ \mathrm{S_{i2}}, \dots, \mathrm{S_{ik}}]$, i=1, …, 450 and k is the dimension of the feature vector, which is related to the length of the signal. "Leave one out" for testing is used. In i-th iteration, i-th sample $\mathrm{S_i}$ was left for testing and the other 449 samples were used for training.

Two types of data fusion systems were designed and are shown in Figure 6.1. System I has a two-stage data fusion structure, with which the classification outputs are fused for each sensor in the first stage and the two sensors are fused together in the second stage. The purpose is of this design is to see the difference between each sensor in the fall detection system. System II fused all six information sources directly. This is more convenient when a system has lots of sensors and multiple classifiers are involved. In these two diagrams, the total sample number M equals to 450, and three parallel-acting classifiers were used to produce fall confidence for each sensor. Six sources of information were needed so they could be fused together to let the fall detection system make a final decision.

6.2.1 A Two-stage Fusion System

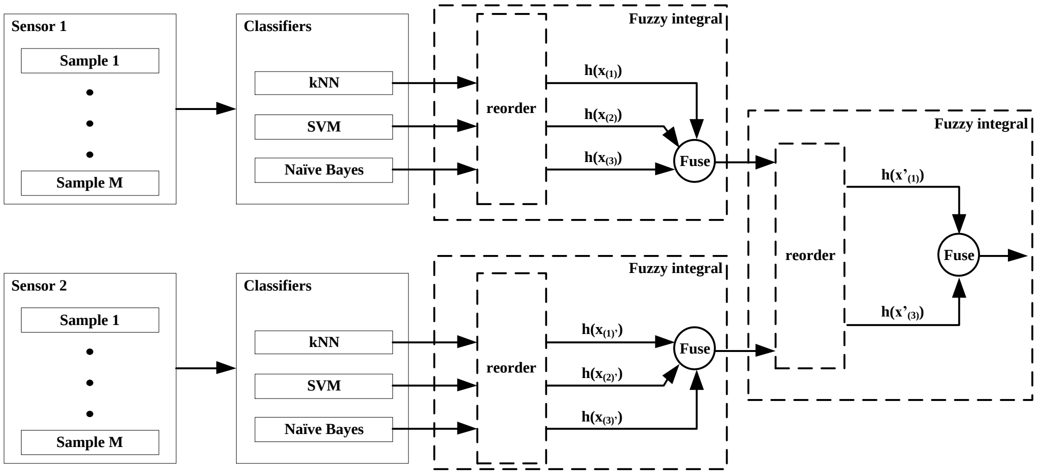


Figure 6.1 The diagram of the fall detection system with fuzzy integrals using a two-stage data fusion system. (a) The outputs of three parallel-acting classifiers are fused together in the first stage for each sensor; (b) Two sensors' fused confidence are fused in the second stage.

6.2.2 A Direct Fusion System

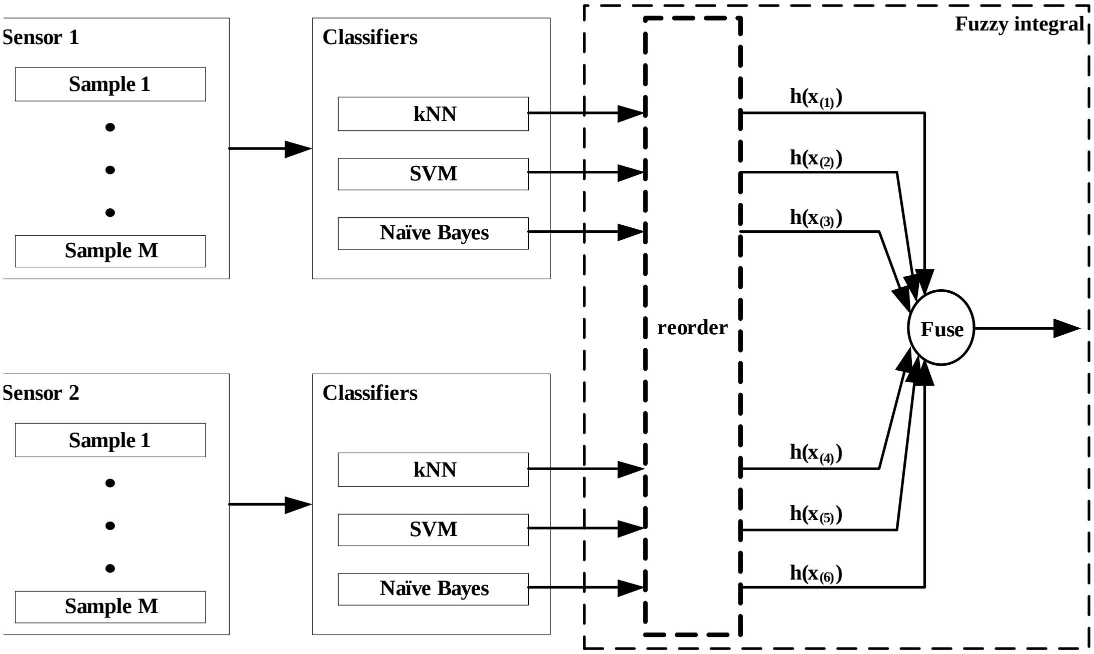


Figure 6.2 The diagram of the fall detection system with fuzzy integrals with a direct data fusion system.

6.3 Learning the Fuzzy Measure

A gradient descent method was used to learn the fuzzy measure. Minimum square error is considered for a two-class classification problem. The cost function is defined as

$$E^2 = \frac{1}{2}\sum_{w\in C_1}\Big(\int_C hog(h_w) - T_{C_1}\Big)^2 + \frac{1}{2}\sum_{w\in C_2}\Big(\int_C hog(h_w) - T_{C_2}\Big)^2 , \qquad (6.5)$$

where T_{C_1} is the desired output for *i*th class and $\int_C hog(h_w)$ are minimized under constraints.

By taking a partial derivative with respect to each of the density g^j, we get

$$\begin{aligned} \frac{\partial E^2}{\partial g^j} = {} & \sum_{w\in C_1}\Big(\int_C hog(h_w) - T_{C_1}\Big)\frac{\partial}{\partial g^j}\int_C hog(h_w) + \\ & \sum_{w\in C_2}\Big(\int_C hog(h_w) - T_{C_2}\Big)\frac{\partial}{\partial g^j}\int_C hog(h_w) \end{aligned} . \qquad (6.6)$$

Gradient of the discrete Choquet integral with respect to Sugeno λ measure can be obtained by differentiating Equation (6.3).

The partial derivative of (6.4) with respect to g^j is

$$\begin{aligned} \frac{\partial g(\{^{\alpha_i}H\})}{\partial g^j} = {} & \frac{\partial g_{(i)}}{\partial g^j} + \frac{\partial g(\{^{\alpha_{i-1}}H\})}{\partial g^j} + \frac{\partial \lambda}{\partial g^j} g_{(i)} g(\{^{\alpha_{i-1}}H\}) \\ & + \lambda \frac{\partial g_{(i)}}{\partial g^j} g(\{^{\alpha_{i-1}}H\}) + \lambda g_{(i)} \frac{\partial g(\{^{\alpha_{i-1}}H\})}{\partial g^j} \end{aligned} , \qquad (6.7)$$

where $\partial\lambda/(\partial g^j)$ is deducted by the author in [21] by

$$\frac{\partial \lambda}{\partial g^j} = \frac{\lambda^2 + \lambda}{1 + \lambda g^j [1 - (1+\lambda)\sum_{i=1}^{n}(\frac{g^j}{1+g^j\lambda})]} . \qquad (6.8)$$

6.4 Algorithm

For a fall detection system composed by *M* sample, *N* sensor, and *L* types of classifier, the flowchart of the algorithm is shown by Figure 6.3. In our experiment, *M*, *N* and L are 450, 2 and 3, respectively. So the number of information sources is N x L, which equals 6. Some major steps of the algorithm are given as follows.

Step 1: Compute the spectrogram with short time Fourier transform STFT(f, t) to raw radar signal.

Step 2: With an assumed human torso motion frequency range [25 Hz, 50 Hz], the energy burst curve is computed by $EB(t) = \sum_{f=25\text{Hz}}^{50\text{Hz}} \text{STFT}(f, \text{t})$, and its peak location is at t_{peak}.

Step 3: Extract MFCC features from the raw radar signal with a 2-second window, which includes the peak location t_{peak} on the energy burst curve.

Step 4: Classify the extracted MFCC features from each sample by three classifiers, NN, SVM and naïve Bayes, respectively.

Step 5: The outputs of the classification result generate six groups of fall confidence $Conf_{\text{ji}} \epsilon [0, 1]$. Considering the difference among classifiers, fall confidence for NN and naïve Bayes can be obtained by

$$Conf_{\text{kNN/NB}} = normalized(\frac{\text{Dist}_{\text{fall}} - \text{Dist}_{\text{nonfall}}}{\text{Dist}_{\text{fall}}}) \qquad (6.9)$$

where $\text{Dist}_{\text{fall}}$ is the distance to the nearest fall from current testing sample, and $\text{Dist}_{\text{nonfall}}$ is the distance to the nearest non fall from current sample.

For SVM, the probability for the current classification result is taken as the fall confidence.

Step 6: Fuse $Conf_{\text{ji}}$ by Choquet fuzzy integral with learning fuzzy measure to obtain the aggregated fall confidence $Conf_{\text{i}}$.

Step 6.0: Random initialize the fuzzy densities: g^j , j=1,…6. Calculate the corresponding λ by Equation (6.2).

Step 6.1: Compute the fuzzy integral with Equation (6.3).

Step 6.3: Update fuzzy density by $g_{new}^{j} = g_{old}^{j} - \alpha\frac{\partial E^2}{\partial g^j}$, α is the learning rate.

Step 6.4: Calculate the total fuzzy density error from two continuous steps with $\sum_{j=1}^{6} |g_{new}^{j} - g_{old}^{j}| < \zeta$ where ζ is the calculation accuracy for fuzzy measure.

Step 6.5: The iteration stops when the termination condition $\sum_{j=1}^{6} |g_{new}^{j} - g_{old}^{j}| < \zeta$ is met or the maximum iteration step is achieved and Step 6 is exited with the current fuzzy integral as the aggregated fall confidence $Conf_{\mathrm{i}}$,. Otherwise, go to Step 6.1 and continue the learning.

Step 7: Threshold the aggregated fall confidence $Conf_{\mathrm{i}}$, a final decision can be made and a ROC curve is drawn.

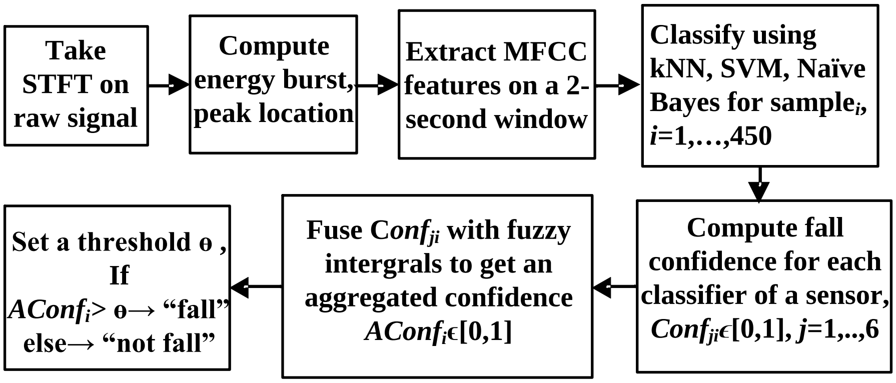


Figure 6.3 The flowchart of fall detection algorithm.

6.5 Experimental Results and Analysis

For this data fusion of six information sources, problem experiments were designed by different fuzzy measure acquiring methods where AUC values were taken from each information source as the fuzzy measure g^j, j=1,…,6 for both fusion system I and II; furthermore, we used learned fuzzy measure for direct fusion System II.

6.5.1 Two-stage Data Fusion System

At each data fusion stage, Sugeno fuzzy integral and Choquet integral were applied, respectively. The fusion result of different classifiers for the same sensor with Sugeno fuzzy integral and Choquet integral is shown by Figure 6.4, which is accomplished in the first stage. With the results from first stage, the information from two sensors was fused in the second stage with Sugeno fuzzy integral and Choquet integral again.

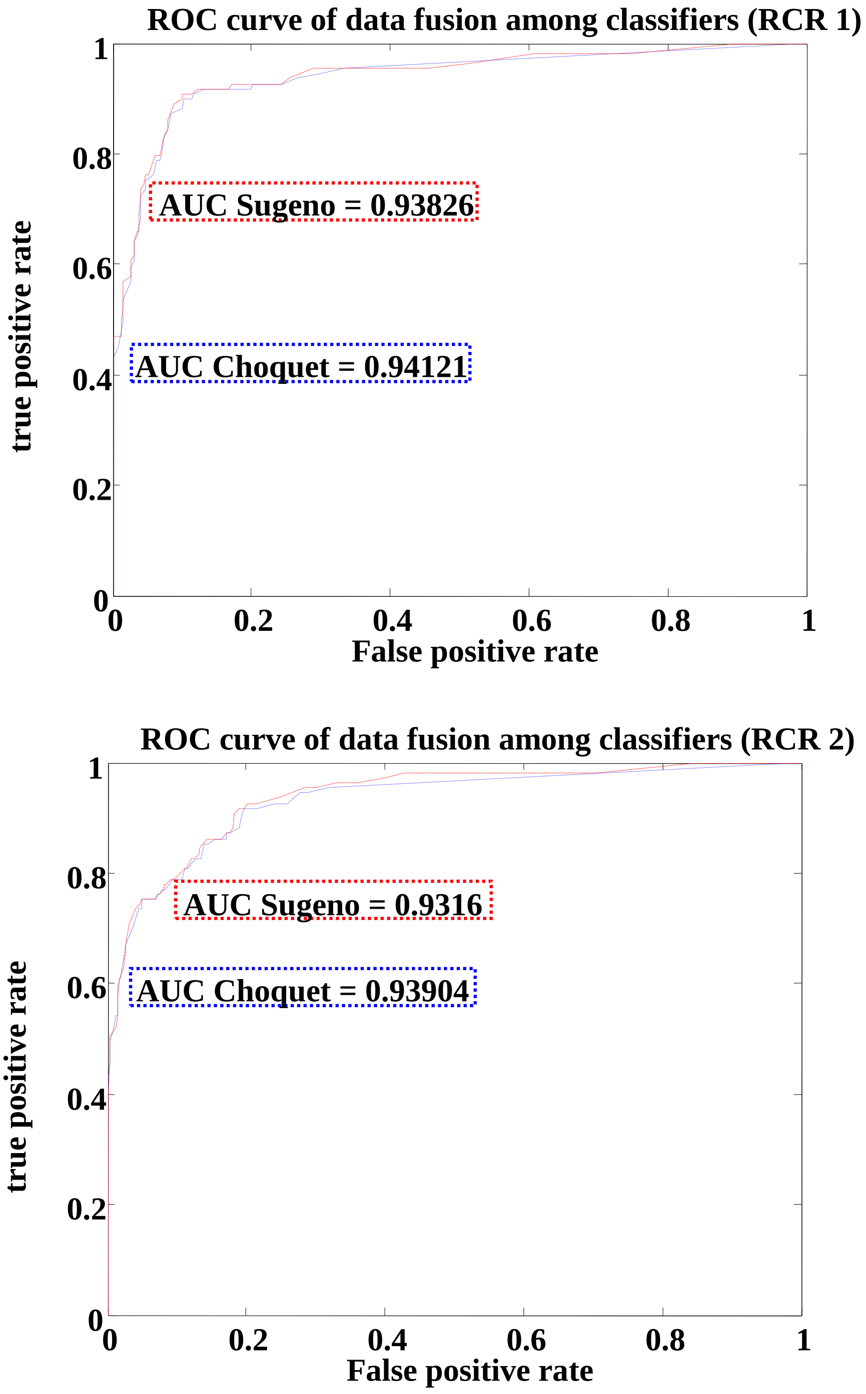
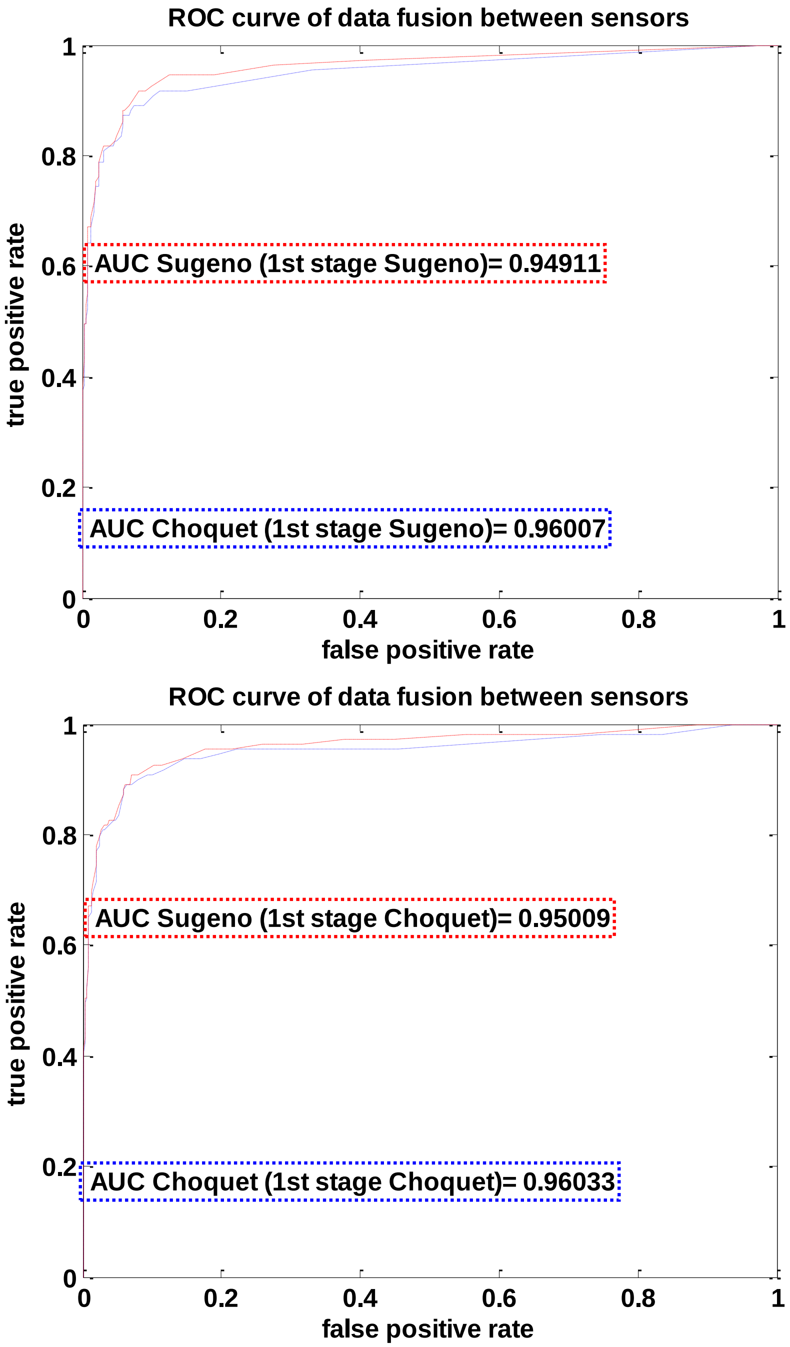


Figure 6.4 The first stage data fusion for each sensor (dash - Sugeno; dot - Choquet)



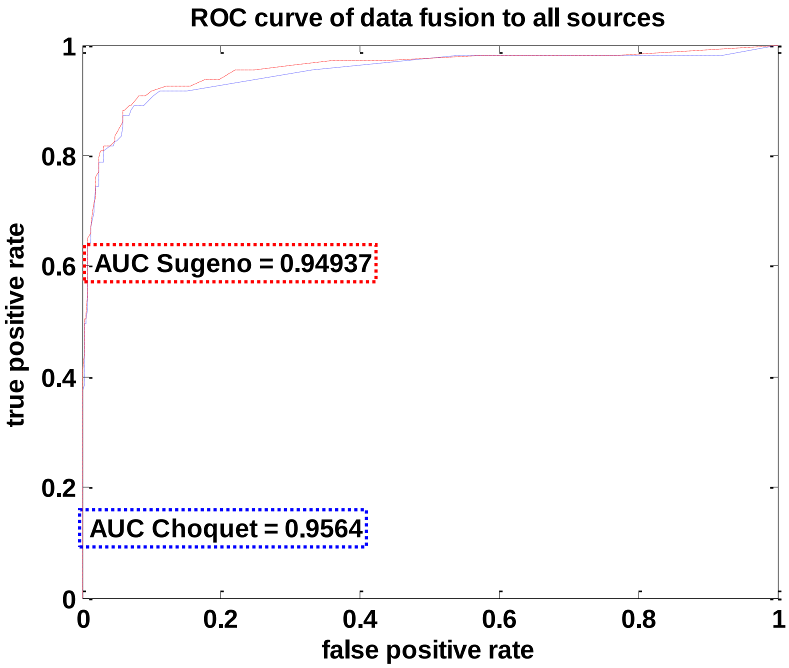
ROC curve of data fusion between sensors

ROC curve of data fusion between sensors

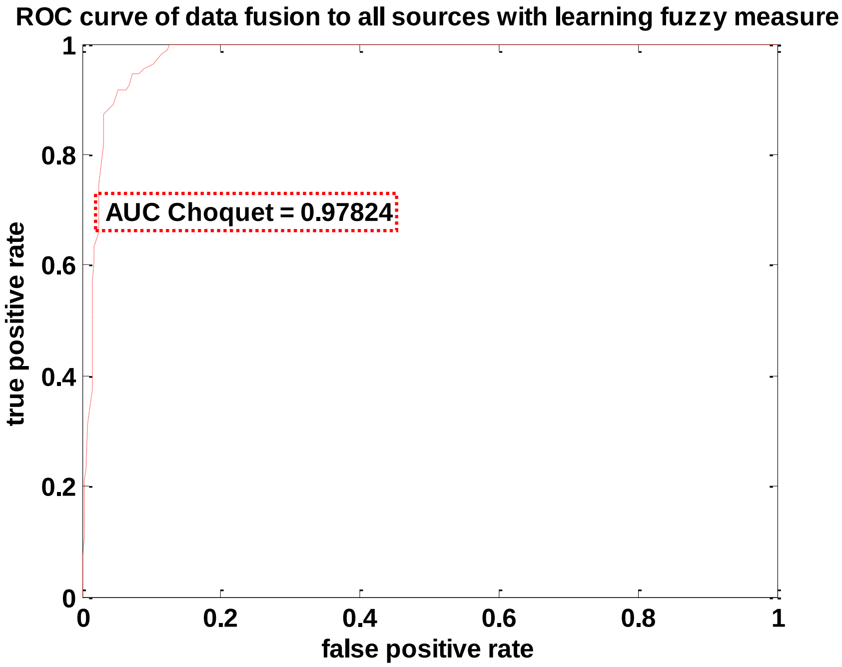
Figure 6.5 The second stage data fusion of two sensors (red dash – Sugeno integral, blue dot – Choquet integral)

6.5.2 Direct Data Fusion System

Firstly, the information sources are fused with Sugeno fuzzy integral and Choquet integral using the fixed fuzzy measure, AUC values. The fused results are shown in Figure 6.6 (a). Then a Choquet integral was implemented with the learned fuzzy measure as shown in Figure 6.6 (b).



(a) Fixed fuzzy measure



(b) Learned fuzzy measure

Figure 6.6 Direct data fusion to all information sources (red dash – Sugeno integral, blue dot – Choquet integral)

6.6 Summary

To sum up the above results, Table 6.1 shows the performance of data fusion with different designed fusion systems and fuzzy measure acquiring methods.

It is observed that Choquet integral performs better than Sugeno integral by taking the same AUC values as the fuzzy measure. The data fusion result of Choquet integral with the learned fuzzy measure gives a more encouraging result, which has an AUC as 0.97824.

After the aggregation operation with fuzzy integrals to the six information sources, the fall detection of the system performed better than the detection results by using a single classifier. We also found that the obtained AUC values various significant integrals from 0.87888 to 0.97415 for each classifier of a sensor before fusion. Fusion result produces a more reliable result with multiple classifier applications and lets the fall detection system make a decision based on all available information.

Table 6.1 Comparison of area under ROC curve (AUC)

			Sugeno integral		Choquet integral	
			Sensor1	Sensor2	Sensor1	Sensor2
System1	1st stage		0.93826	0.9316	0.94121	0.93904
	2nd stage	Sugeno integral	0.94911		0.95009	
		Choquet integral	0.96007		0.96033	
System2	Direct fuse		0.94937		0.9564	
		Choquet integral with learned fuzzy measure	0.97824			

Chapter 7 Effective Detection Range

This chapter evaluates the effective detection range around a radar sensor. Thisdetection range comes from in–front–of–radar sensors placed at different heights; hence, floor radar and wall radar will be examined as well as ceiling radar. The influence of relative falling direction and the distance from objects surveyed to the sensor is discussed. The influence of multiple falling positions under the ceiling radar is investigated. The effect of different subjects on detection results is also considered. We reorganized the dataset from previous configurations and combined the datasets for experimental purposes.

7.1 Theoretical Simulations and Experiments

Configuration II was designed in our MU lab to study the effective detection range around radar sensors. This experiment measured the effective detection ranges in front of the floor and wall RCRs and below the ceiling RCR. The negative elevation angle is the angle between the horizontal plane and the line of sight. If the RCR is higher h, the detection range x also grows by keeping the same negative elevation angle as shown in Figure 3.1. Thus, we used a wall RCR in the lab instead of floor RCR to measure the effective detection range in front of the RCR. The wall RCR was attached to the middle of one short wall with a height of 1.27 meters, which is about the chest height of a falling subject. The ceiling RCR was mounted on a beam in the center of the room. Both RCRs faced toward the center of the room. At each of the 21 locations, two subjects were performing three types of falls around the room's center. Figure 3.7 shows their different positions.

In a Cartesian coordinate system as shown in Figure 7.1, we assume that the subject falls on ground with a velocity vector $\bar{v} = [0, 0, -c]$. The unit vector is defined by $\hat{u} = u / \|u\|$. It is used to represent the direction vector from falling subject to the sensor with a unit length. In this case, the radial velocity in the direction of the RCR was the inner product of the velocity $\bar{v}$ and unit vector $\hat{u}$. The Doppler frequency on the floor was computed at each location on floor based on the Doppler effect theory described by Equation (2.7). The algorithm for the simulation is given in Algorithm 7.1.

As a comparison, the maximum signal strength during a fall is displayed at each position in the experiments. Figure 7.2 shows the theoretical simulation result in (a) and the experiment signal strength in (b) for the ceiling radar. The comparison between simulation and experiment for wall radar is shown in Figure 7.3.

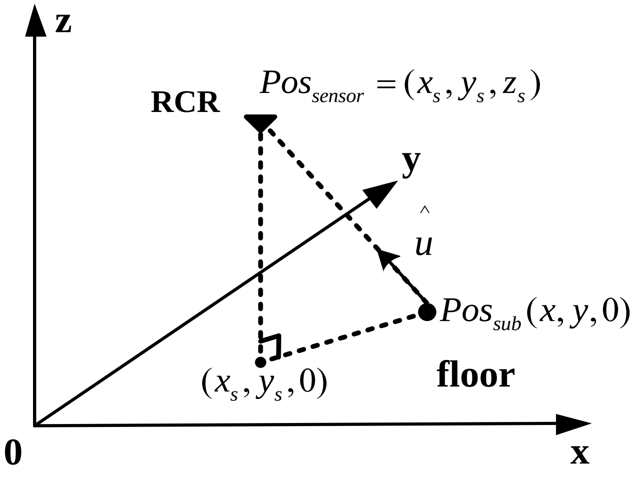


Figure 7.1 Detection range in front of the sensor with a certain height in the Cartesian coordinate system.

The meaning of the intensity map is different between the simulation and experiment. As a result, the range of the energy bar is different. However, the pattern of the intensity map can reflect the detection range on the floor. On the color bar, the more red color means a higher possibility of fall detection. The blue areas indicate a lower possibility of fall detection. By observing the results, the experimental pattern matches the simulation pattern well for both radar positions.

Algorithm 7.1 Simulation of Doppler frequency.

```
We assume that the room size is WxLxH and the fall occurs on the floor surface.
The carrier frequency is $f_0 = 5.8\times 10^9$.
The sensor position in the room is $Pos_{sensor} = (x_s, y_s, z_s)$;
We divide the floor surface WxL with small patches with an interval length of $\Delta d$:
the number of bins for width is wid=0: $\Delta d$ :W ;
the number of bins for length is len=0: $\Delta d$ :L.
We build up a table with Len_W rows and Len_L columns.
for i=1:Len_L
    for j=1:Len_W
        the current subject position is $Pos_{sub}(i, j, 0) = [\text{wid(i), len(j), 0}]$ ;
        the unit vector is $\hat{u} = (Pos_{sub}(i, j, 0) - Pos_{sensor}) / \| Pos_{sub}(i, j, 0) - Pos_{sensor} \|$;
        the radial velocity is $\bar{v} \bullet \hat{u}$ ;
        the Doppler frequency is $f_D(i, j) = f_0(2\bar{v} \bullet \hat{u}/c)$;
    end
end
```

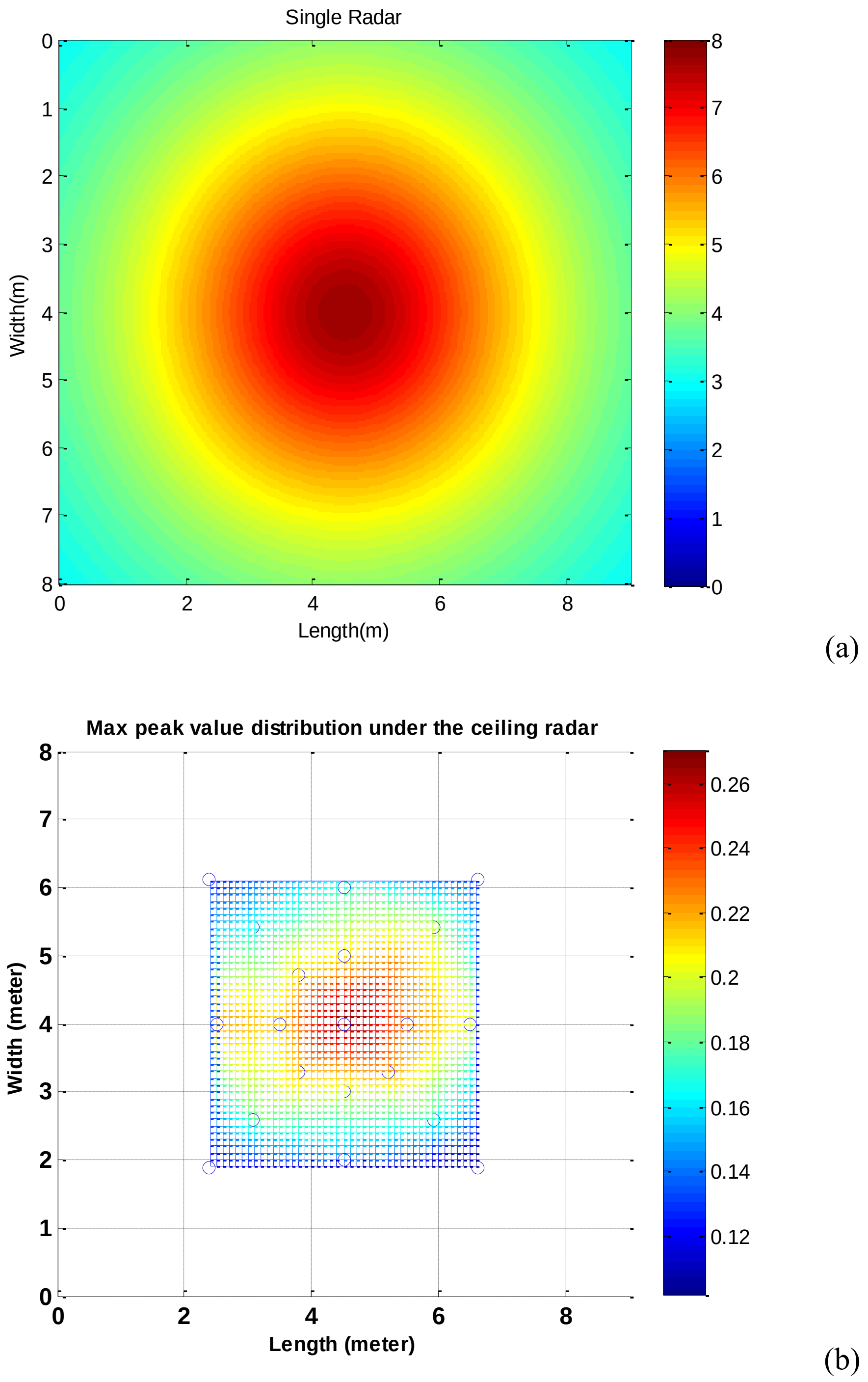
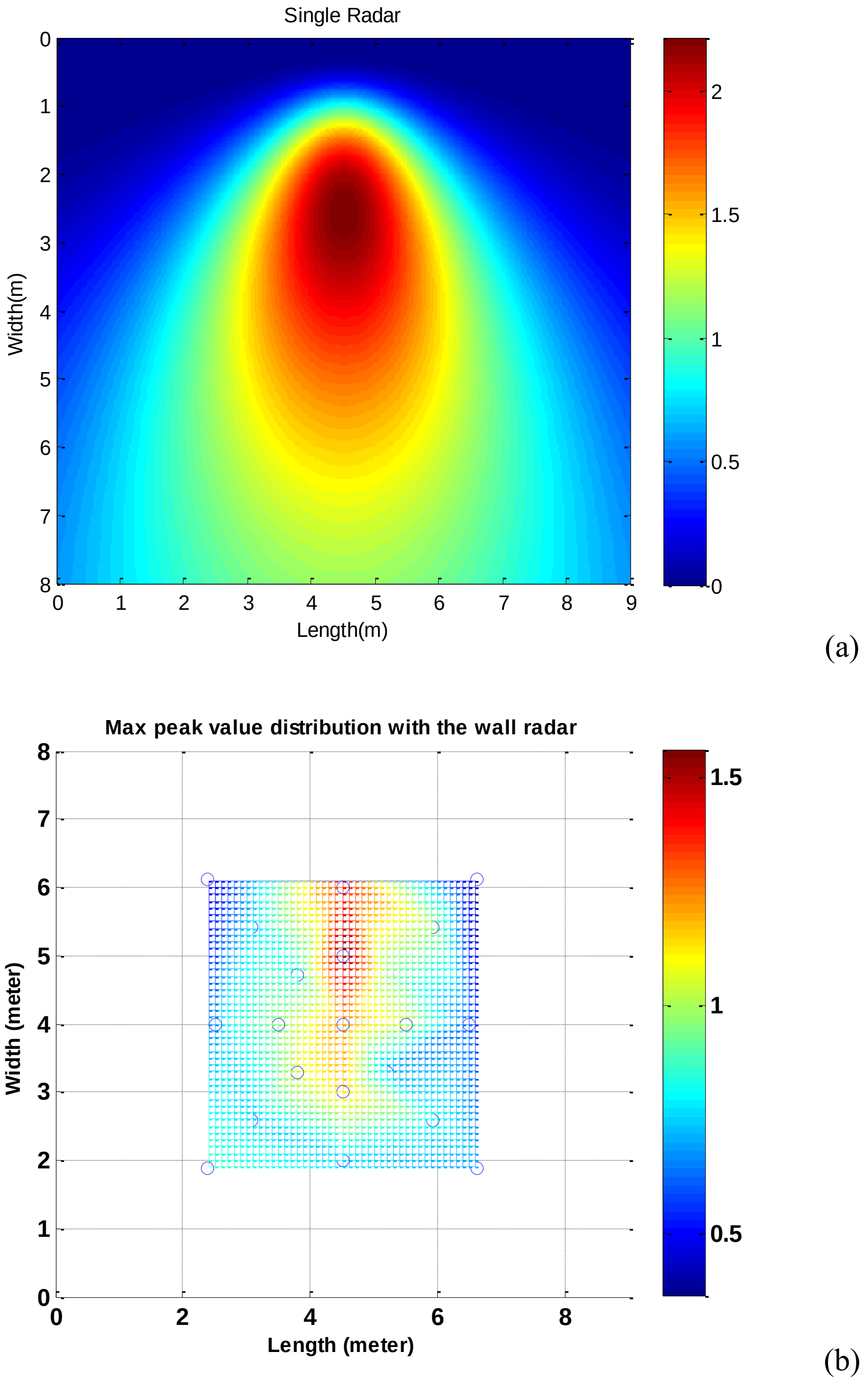


Figure 7.2 Detection range of the ceiling radar sensor. (a) Theoretical simulation; (b) Real experiment.



(a)

(b)

Figure 7.3 Detection range of the wall radar sensor. (a) Theoretical simulation; (b) Real experiment.

7.2 Influence Factors in Fall Detection Experiments

The possible factors that influence detection results are discussed here. Those influence factors include the relative falling direction, distance to the sensor and subject difference.

7.2.1 Influence of Relative Falling Direction and Distance in front of the Sensor

We employed Configuration I to explore the influence of fall direction on fall detector performance. In Figure 7.4 and Figure 7.5 we show the fall detection performance for different fall directions: towards, away, oblique and perpendicular the two sensors RCR1 and RCR2, respectively. From Figure 7.4, we see that RCR1 has the best performance (AUC=0.996) when the fall is away from the sensor and the worst one (AUC=0.889) when the fall is perpendicular to it. In Figure 7.5, the best detection of RCR2 was when the fall was towards the sensor (AUC=0.998) and the worst was when the fall was perpendicular to the sensor (AUC=0.966). We note that RCR2 detection was, in general, better than RCR1's since the fall mat was closer to RCR2.

From this experiment we can conclude that the fall direction has an important impact on detection performance, introducing about 10% variability in detection results. Best detection is obtained when the fall is along the sensor axis. Consequently, placing the radar on the floor might not be the best solution for fall detection. Instead, positioning the sensor higher up on the wall or ceiling might improve the detection performance.

7.2.2 Influence of Relative Falling Direction and Distance below the Ceiling Sensor

We used Configuration II to find the influence factor in detection for a ceiling radar sensor. We also investigated a wall radar sensor as a comparison. In this configuration, two subjects performed three types of falls at each of the 21 positions. The fall recognition results obtained for Configuration II are given in Figure 7.6. From Figure 7.6, we can conclude that the ceiling mounted RCR performed better (gave more usable results) than the wall one.

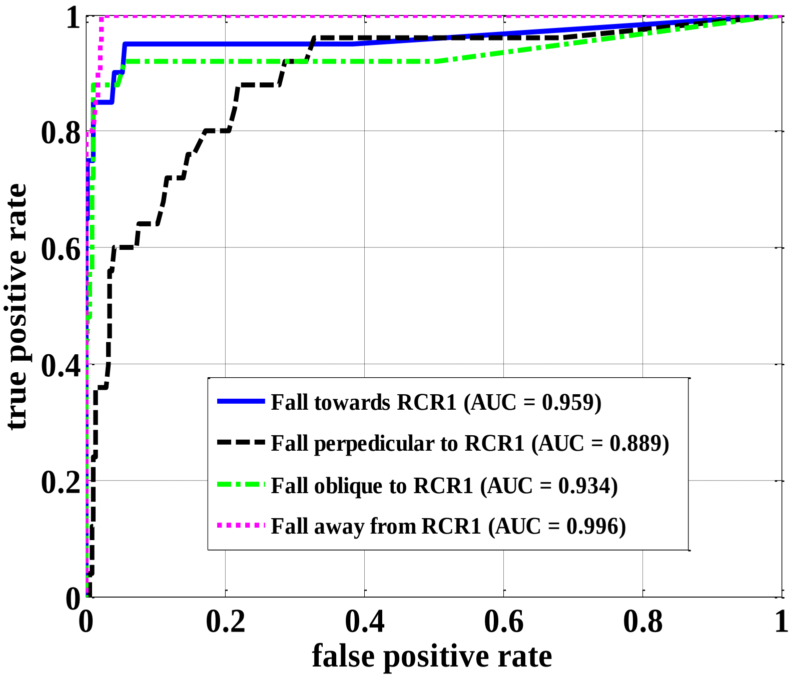


Figure 7.4 Detection results for RCR1.

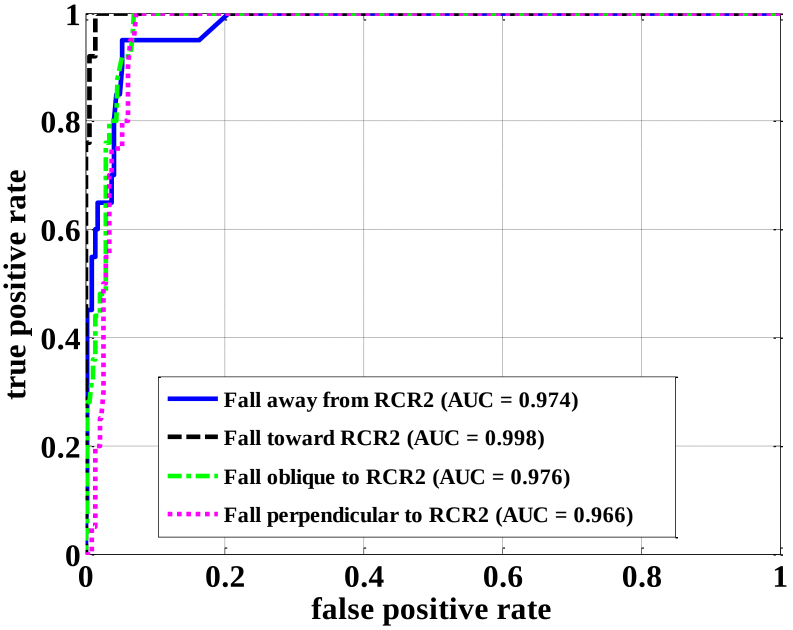


Figure 7.5 Detection results for RCR2.

Moreover, although the ceiling RCR had some variability in detection due to fall direction, representing an approximate 30% deviation from that of wall and floor mounted sensors. However, comparing Figure 7.4, 7.5 and 7.6, we see that the ceiling RCR had a slightly lower performance (3-4%) than the floor RCRs when falls are closer to the sensor.

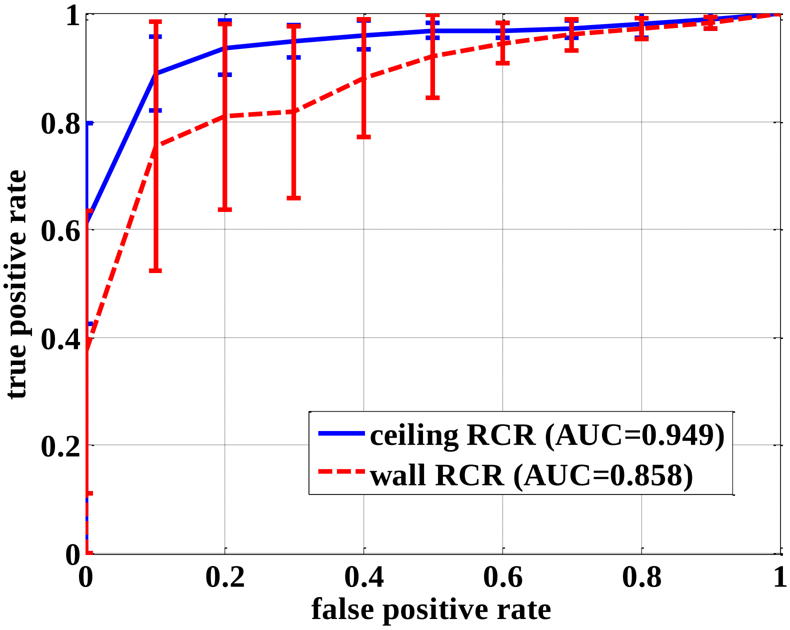


Figure 7.6 Detection results for forward fall.

The CW_SYS dataset is reorgnized by the distance from falling position to the room center on the floor surface: 0 m, 1 m, 2 m, 3 m. There are 6 falls, 48 falls, 48 falls and 24 falls at each distance. The ceiling sensor collected 342 non-falls and wall sensor collected 349 non-falls. The results for ceiling sensor are shownin Figure 7.7. As a comparison, the detection results with the wall sensor are shown in Figure 7.8.

We find that the AUC decreases when the distance increases from 1 m to 3 meters below the ceiling radar. We also notice that the detection result at room center is not as good as the results at the distances of 1m and 2m. The reason is that the fall number at room center is 8 times smaller than the fall numbers at 1 m and 2 m. The detetor trained by 6 falls at room center is still better than the detector trained by 24 falls at a distance of 3 m. The conclusion drawn is that a better performance is achieved when the distance is shorter from the falling subject to the room center under the ceiling radar sensor. This conclusion also fits the conclusions reflected in Figure 7.2.

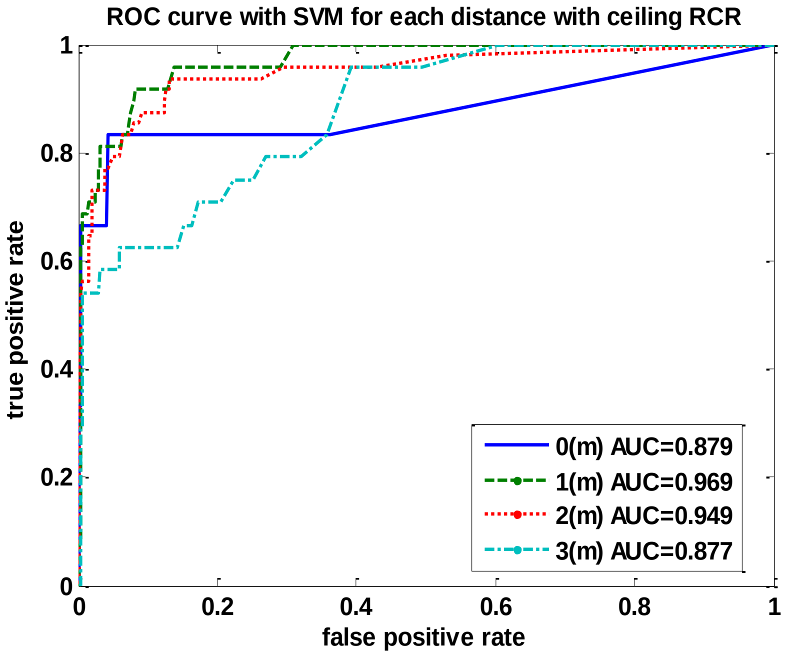


Figure 7.7 Detection results for ceiling radar sensor.

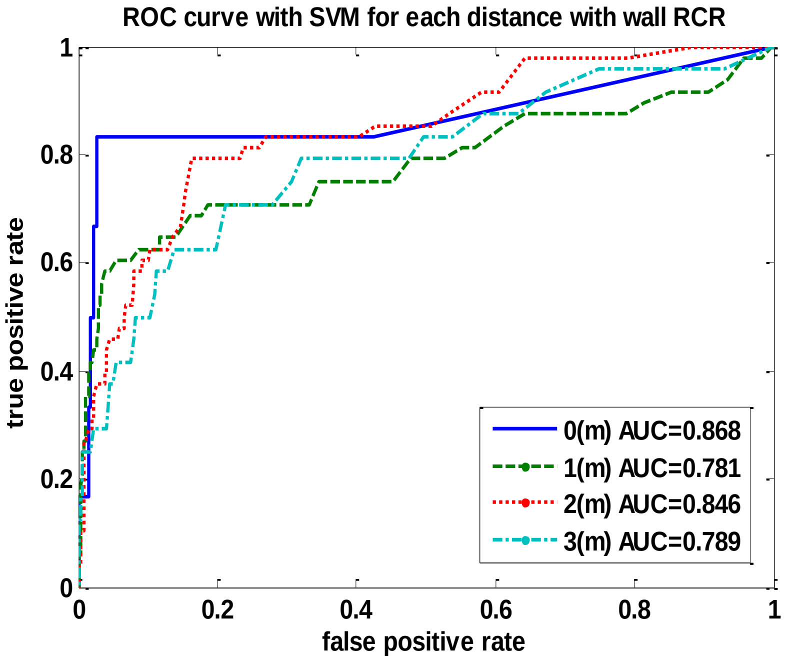


Figure 7.8 Detection results for wall radar sensor.

For the wall radar sensor, the detection performance should be distributed on an elliptical contour line in front of the sensor as shown in Figure 7.3. However, we still evaluated the performance at four distances since most of the activities were around the room center in Figure 7.8. In comparison, the ceiling sensor performed better than the wall sensor at each distance.

7.2.3 Influence of Different Falling Subject

We considered two experiments based on inter-level and intra-level falls. The inter-level falls were performed by different subjects with same skill level for falling, based on training and aptitude of students. The intra-level falls were performed by subjects with different skill levels for falling such as stunt actors and students.

7.2.3.1 Different subjects—Student1 and Student2

For inter-level experiments, CW_SYS dataset is used. The student Subject 1 had 63 falls and 160 non-falls captured by the ceiling RCR and 170 non-falls as captured by the wall RCR. The student Subject 2 had 63 falls and 175 non-falls detected by the ceiling RCR and 183 non-falls recorded by the wall RCR. Each student fall was detected with the detector trained by another student subject, as shown in Figure 7.9. Each student fall was also detected with the detector trained by the same student in Figure 7.10.

Comparing the results in Figure 7.9 and Figure 7.10, we find that the fall detector performs better when the fall is from the same subject who generates the training data for the detector. The ceiling RCR outperformed the wall RCR in all cases as shown in Figure 7.9 and Figure 7.10.

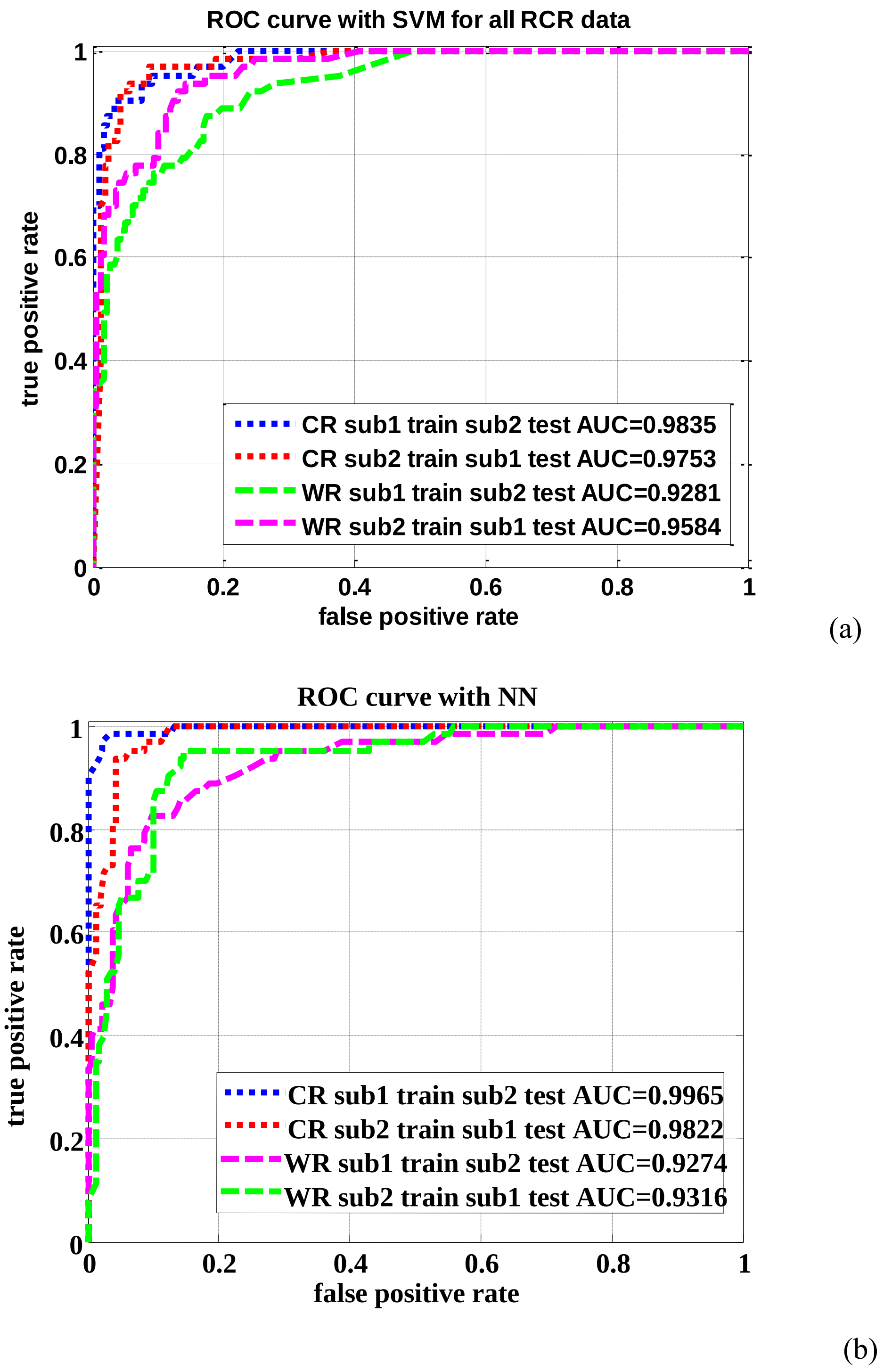


Figure 7.9 Detection results for the detector trained and tested with different subjects. (a) SVM; (b) NN.

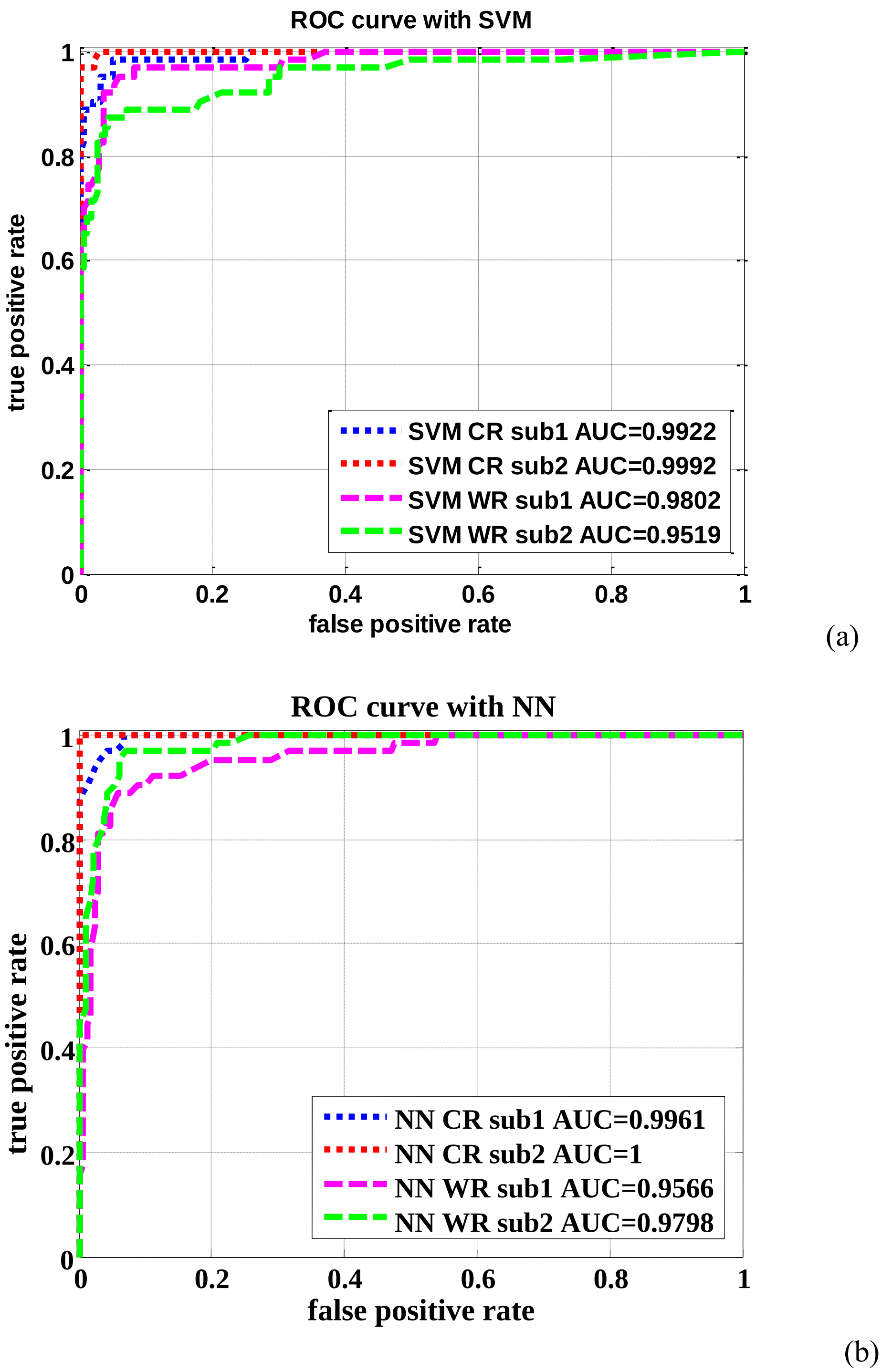


Figure 7.10 Detection results for the detector trained and tested with same subject. (a) SVM; (b)NN.

7.2.3.2 Different Subject—Students and Stunt Actors

For intra-level experiments, the ceiling radar data in CW_SYS and CF_SYS datasets were used. The students had 126 falls and 335 non-falls. The stunt actors had 105 falls and 375 non-falls.

From Figure 7.11, we find that the detector trained with stunt actor falls to detect student falls performed better than the detector trained with student fall data to detect stunt actor falls. The reason is that the stunt actor had 21 different fall types, which covered the three types of falls from other students.

Figure 7.12 shows that the detector trained with student data to detect student falls performed better than the detector trained with stunt actor data to detect stunt actor falls. The reason is that the ratio between the fall type and total fall number is quite different in each dataset. The ratio is 3/126=0.02 for the student fall dataset and 21/105=0.2 for the stunt actor fall dataset. For the leave one out cross validation, the better performance was generated when the ratio was low as shown in Figure 7.12.

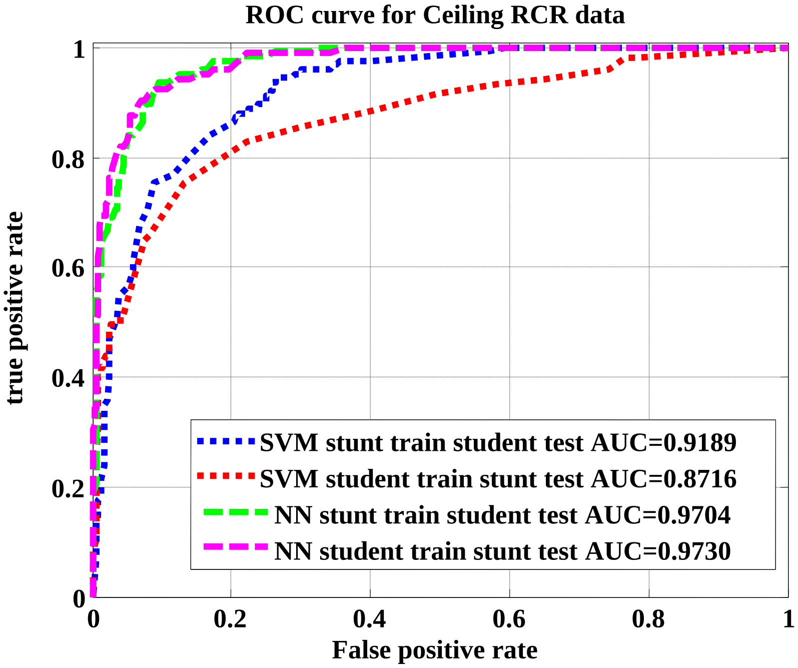


Figure 7.11 Detection results for the detector trained and tested with different subject fall skill levels.

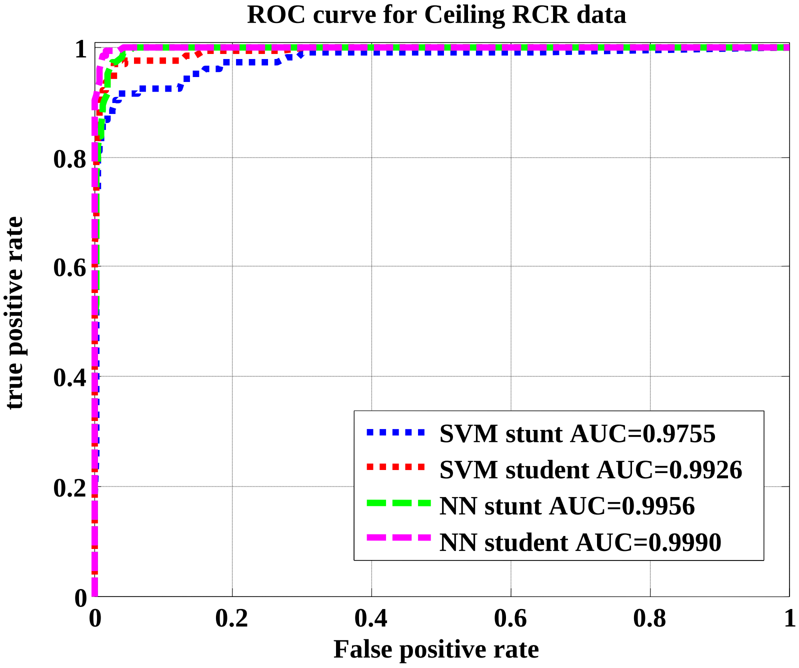


Figure 7.12 Detection results for the detector trained and tested with same subject fall skill level.

7.3 Summary

This chapter discusses the effective detection range around the radar sensor. We did the comparison between the computational model and the maximum signal strength. The results show us that the computational model matches the real experiments since they have the same distribution pattern for the detection range around the sensor. After we validated the experiments with the theoretical simulation, we investigated possible factors that influence the detection results. For FF_SYS, the best detection was obtained when the fall was along the sensor axis and closer to the sensor. For CW_SYS, the ceiling mounted RCR had better performance than the wall mounted one with lower detection variants due to different fall directions. The ceiling sensor performed better than the wall sensor at each distance below the ceiling. Although different subject skill levels is one factor that influenced the detection results when evaluating falls below the ceiling, a more comprehensive training dataset would be helpful to improve the performance.

Chapter 8 Fall Detection in Real Senior Home

Our aim is to detect falls of the elderly resident with a higher accuracy, especially for the case when the resident stays alone at home and no one is there to take care of the resident after a fatal fall occurs. For the real application in the senior apartment, the Doppler radar sensor is sensitive to multiple types of motion generated by visitors, such as repairman, housekeeper, family, friends, *etc*. Those high energy activities add noise to the resident's daily activity data and generate false alarms in the fall detection algorithm. In this chapter, we further reduce the false alarms by proposing and developing several methods to improve the fall detection system performance in real-world data.

8.1 Performance Improvement Methods

8.1.1 Sensor Fusion

Besides the Doppler radar sensor, the Tiger Place senior home was also equipped with other sensors such as motion sensor networks, Kinect, *etc***.** Each sensor technology had its merits and shortcomings. Sensor fusion could increase the system accuracy for detection [29]. We used the motion sensor network [30] to fuse the results from the Doppler radar sensor. Later, we validated the fusion system performance with the Kinect depth image.

8.1.1.1 Motion Sensors Placement in the Home Environment

We deployed ceiling radar systems to six different apartments in Tiger Place. The typical floor plan of a Tiger Place apartment is shown in Figure 8.1. Furniture placement is not shown. There were seven motion sensors (located at the end of each cone of blue lines) and one Doppler radar facing down to the floor (marked by a red cross in Figure 8.1) placed above the ceiling at the center of the living room and dining room. The detection range of the radar is about 6.1 meters with a 90 degree cone angle. Since the height of the room is about 3 meters, the cover range on the floor was a circle 6 meters wide. The longer dimension of the room had a 6.4 meter coverage shown as 21 ft in Figure 8.1. The sofa and other furniture occupied at least 1 meter of floor space in this longer dimension. The Doppler radar covered the remaining 5.4 meters of floor surface. The data logger for each sensor was synchronized with the same data server in Tiger Place.

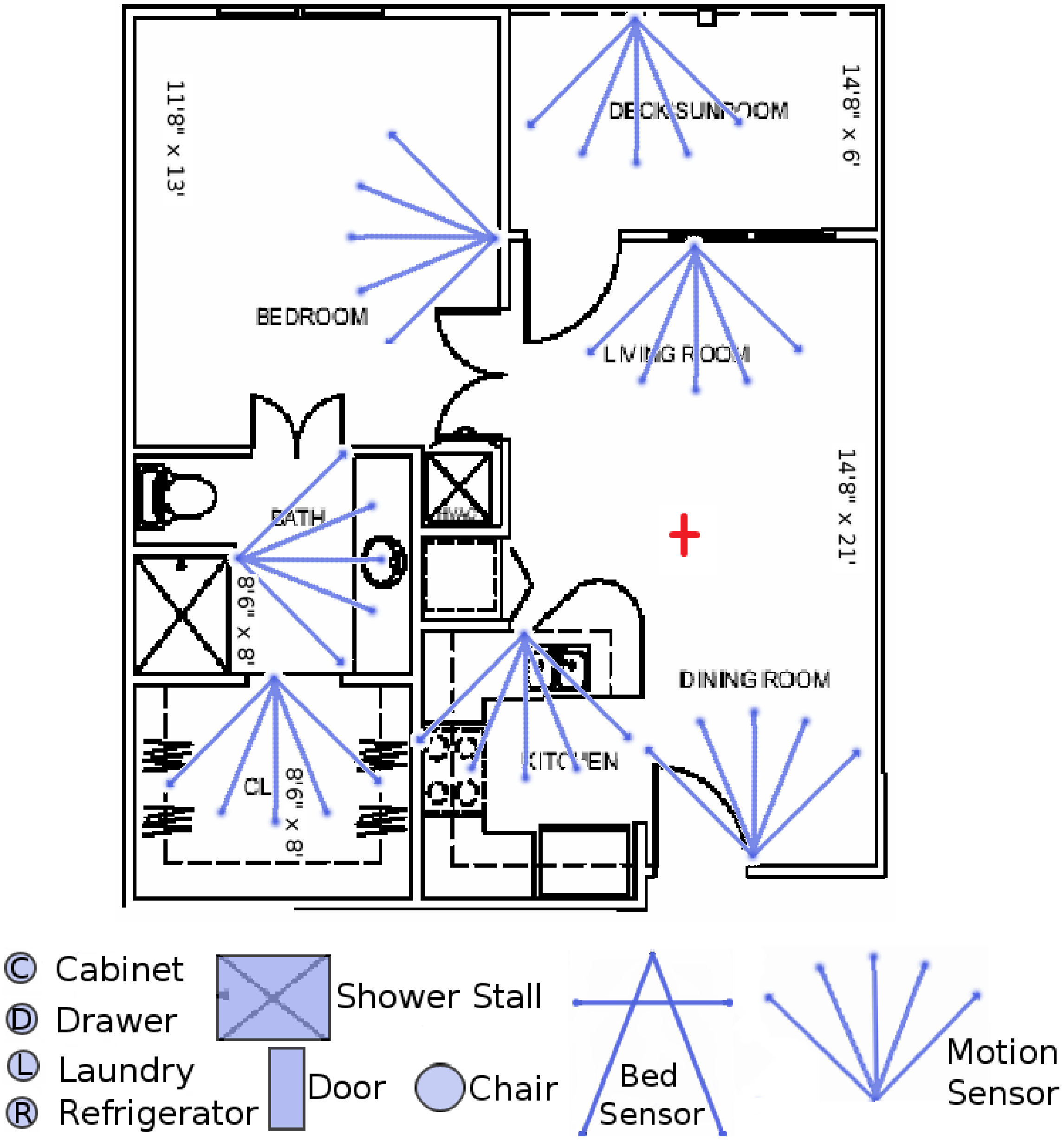


Figure 8.1 The floor plan showing motion sensors placement in a senior apartment.

8.1.1.2 Activity Density Map Generated by the Sensor Network

In the resident activity density map shown in Figure 8.2, the x-axis represents days from left to right and y-axis denotes hours in a day. Each color represents the different motion density levels. The black on the density map means that the resident was away from home. The white represents very low density. The color bar illustrates the density range from 50–99 events/h to >550 events/h using different color blocks from gray to blue sequentially.

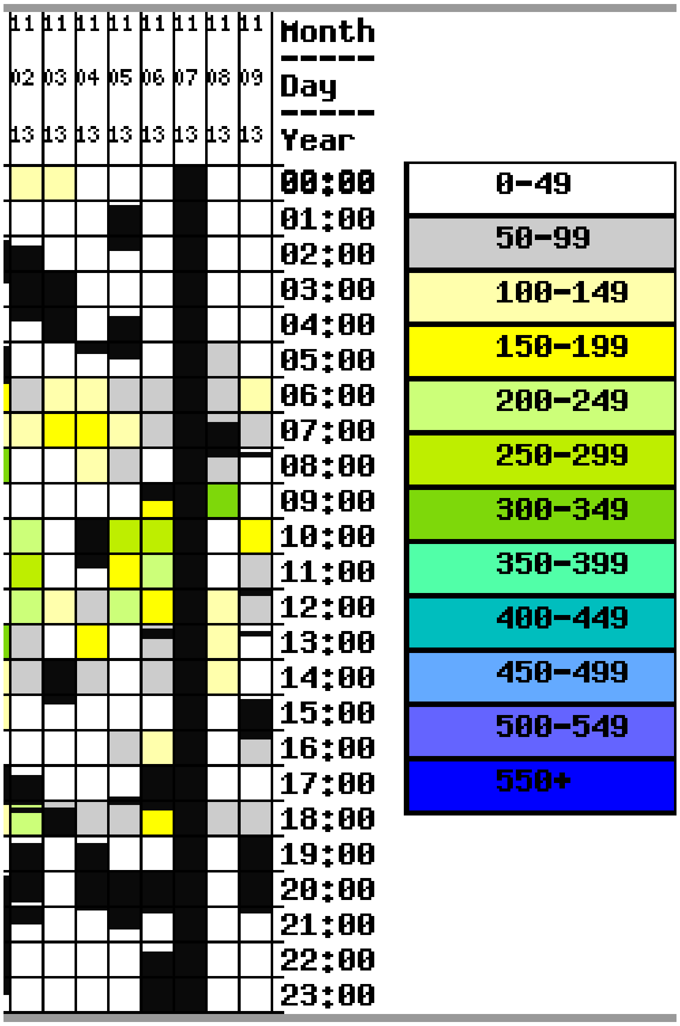


Figure 8.2 Activity density map

8.1.1.3 Description of Doppler Radar Sensor Feature

Since Chapter 4 presented detailed algorithm for the system, this chapter will only briefly cover the data processing procedure.For the Doppler radar sensor signal segment $r(n)$, we first take the short time Fourier transform (STFT) by

$$STFT(m,\omega) = \sum_{n=-\infty}^{\infty} r(n)w(n-m)e^{-i\omega n}. \quad (8.1)$$

Then, we compute the spectrogram by taking the magnitude square to the STFT,

$$spectrogram\{r(n)\} \equiv |STFT(m,\omega)|^2. \quad (8.2)$$

Next, we calculate the energy burst curve using

$$EB(m) = \sum_{\omega=25/(2\pi)}^{50/(2\pi)} STFT(m,\omega), \quad (8.3)$$

and smooth the curve over K bursts to reduce noise:

$$\widehat{EB}(m) = \sum_{i=0}^{K-1} EB(m-i). \quad (8.4)$$

The peaks on this curve are the located potential falls. More detailed examples are presented in the author's and colleagues' article published in a conference proceedings for the IEEE Engineering in Medicine and Biology Society in 2012 [22].

We extracted MFCC features for each 2-second window which contained the located possible fall activities. Each 2-second window data segment was divided into 166 sub-frames with an overlap rate of 0.5. Seven coefficients were extracted from each sub-frame. After throwing the dominant coefficient away, we used 6×166=966 MFCC features to represent this potential fall.

The radar signatures were classified by support vector machine (SVM) into two classes: fall and non-falls. We employed LibSVM [22] to produce a score, fall confidence $conf_{fall}$. For computational efficiency, we used only a linear kernel for SVM in all our experiments. We generated a receiver operating characteristic (ROC) curve by thresholding the SVM scores to evaluate the performance of our fall detection algorithms.

8.1.1.4 Sensor Data Fusion Schema

The outputs from above Doppler radar system were in the form of a radar fall confidence $conf_{fall}$ associated with the corresponding time stamp t_{radar}. The sensors in the motion sensor networks are installed above the door of each room and the main facility area, such as kitchen, bathroom, and closet. An event from the motion sensor network means someone is moving around the sensor. A fall is unlikely to occur if an event from a motion sensor is recorded immediately after it. Figure 3 illustrates the idea of this assumption. The t_{dist} represents the time lapse between t_{radar} and the closest event time stamp t_{motion} thereafter defined as $t_{dist} = t_{motion} - t_{radar}$.

A predefined parameter $Delta$ was used to determine whether the t_{dist} was among the reasonable range for a real fall. If the time lapse is shorter than $Delta$, it is a false alarm. Otherwise, it is a fall. The rule is represented by

$$\begin{cases} if\ t_{dist} < Delta, & false\ alarm \\ if\ t_{dist} \geq Delta, & fall \end{cases} \tag{8.5}$$

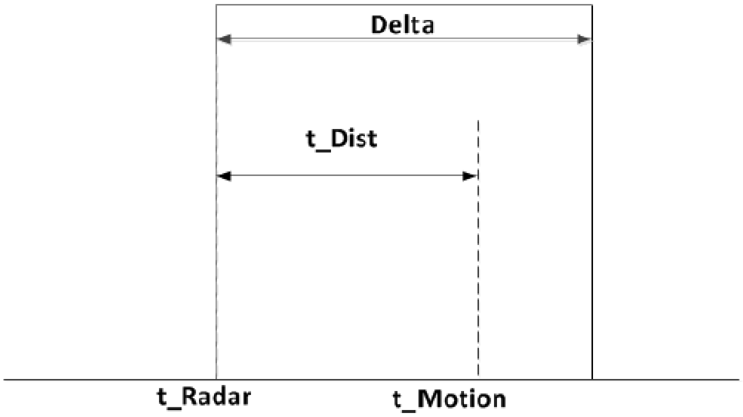


Figure 8.3 Data fusion schema between Doppler radar and sensor network

8.1.1.5 Description of Data Collected by Ceiling Radar

The experimental data includes training data from stunt actors and the continuous testing data from Tiger Place. Due to the large dataset and limited space, only one week's data from an actual elderly person who frequently falls are presented in this paper.

The dataset in Table 8.1 was collected with ceiling radar at senior apartments in Tiger Place. The stunt actor came to senior apartments monthly to perform different types of falls and non-falls. The falls performed were 21 types including losing balance, losing consciousness, trip & fall, reach & fall in different directions - forward, backwards, left side, right side falls, and fall from couch. Non-falls were also collected as part of data on daily activities and could be easily confused with fall activities causing false alarms in fall recognition, such as bending down to pick up item from floor, retrieving items dropped on the floor, sitting down on floor, kneeling down to tighten shoe, *etc*.

Table 8.1 Stunt actor dataset

Environment	Fall #	Non-fall #
Tiger Place	72	98

8.1.1.6 Experimental Results

(1) Improving the performance by including the sensor network: In Figure 8.4, the blue solid line gives the fall confidence generated by Doppler radar fall detection system. The red dotted line presents the event activated by the sensor network. The x-axis denotes the timestamp for both sensor systems.

Figure 8.4 (a) shows that a fall is recognized by the radar with a confidence of 0.96. No event was reported from the sensor network in the following 55 seconds shown. Figure 4 (b) presents the typical false alarms from the senior daily activities: fast opening and closing of door near ceiling radar (0.98 fall confidence); fast turning around and shifting walker direction (0.83 fall confidence). Multiple events from sensor network after Non-fall-2 reflect that the resident is active or a visitor is in the apartment. Although motion is not frequent after Non-fall-2, this false alarm can still be removed with a larger $Delta$ value.

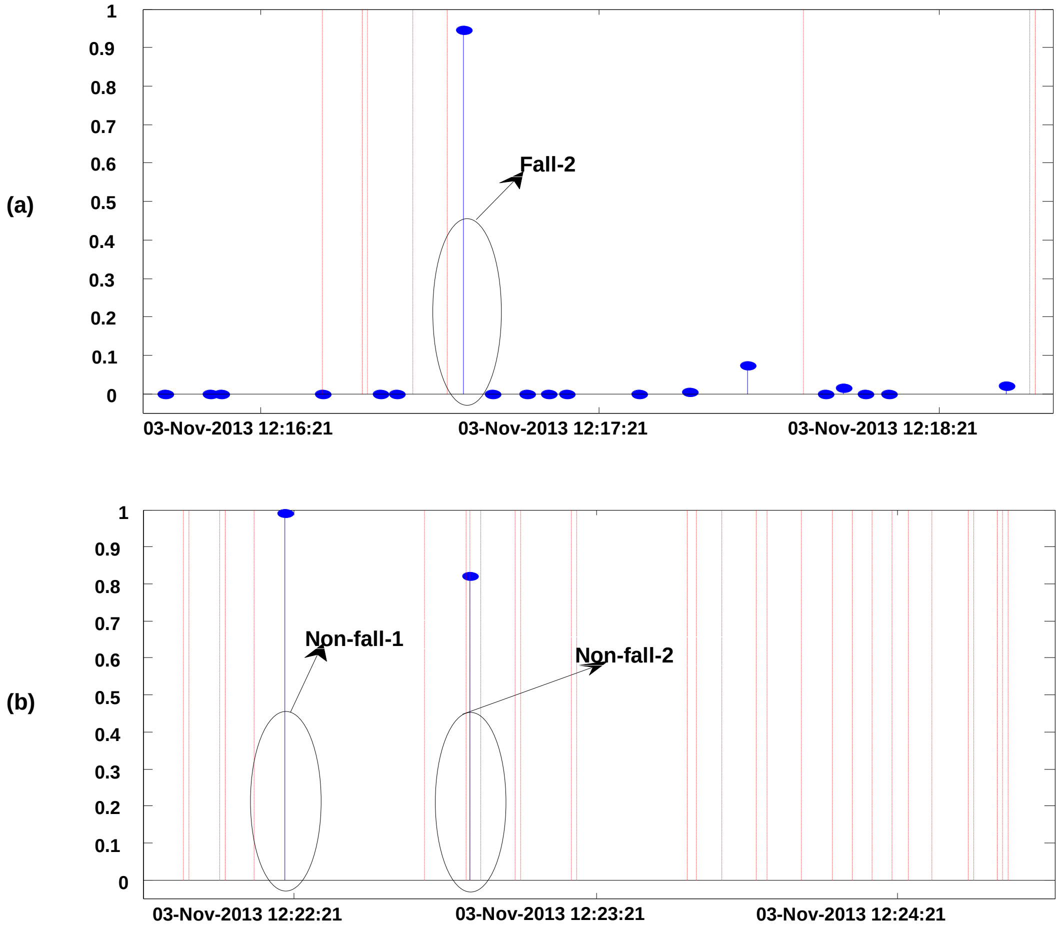


Figure 8.4 The examples fall confidence with events from the sensor network: (a) a natural fall (top); (b) non-fall (bottom)

(2) Leave-one-out cross validation for the stunt actor dataset collected with ceiling Doppler radar sensor

The leave-one-out cross validation was applied on the stunt actor data in Table 8.1. For N=170 samples, each experiment of this validation used N-1 samples for training and the remaining samples for testing. In Figure 8.5 we show the classification results of the radar signature library (see Table I) where we achieved an area under the ROC curve (AUC) of 0.98. While good results were obtained on the signature library, we couldn't replicate them on the continuous datasets.

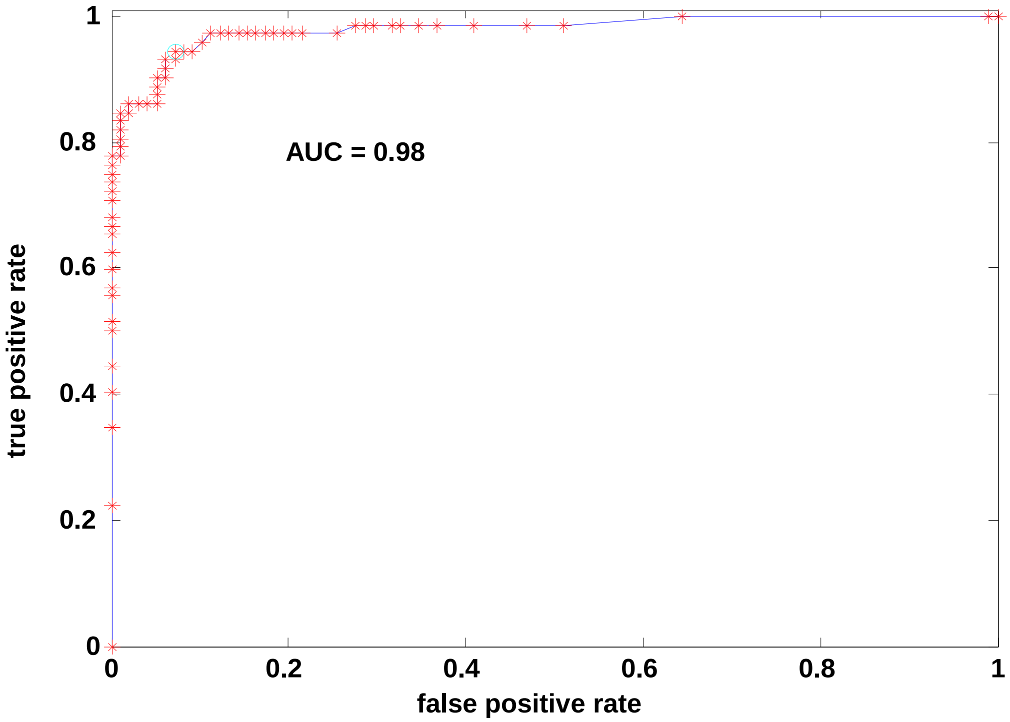


Figure 8.5 The cross validation of the stunt actor dataset

(3) Data fusion results for the actual faller: Using the above stunt actor data for training, we tested our fusion method on one week of continuous data collection in a Tiger Place apartment. Figure 8.6 shows false alarms decreased by 63% while the

parameter $Delta$ increased from 0 to 53 seconds (from right to left). Without the motion sensor network, the fall detection system generated 172 false alarms per day in order to detect all the actual falls. The best performance of the improved system could achieve 62 false alarms per day without losing any falls when $Delta$ equaled 53 seconds. If $Delta$ was larger than 53 second, the false alarm rate kept decreasing but there was some missed falls.

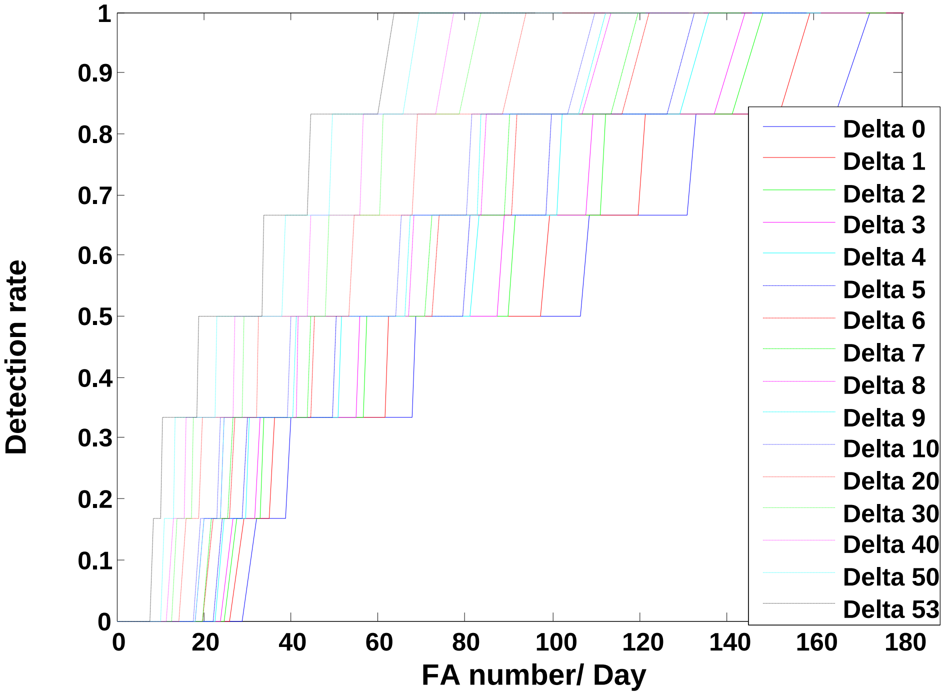


Figure 8.6 The performance of the improved fall detection system on one week of elderly home data

It is reasonable to choose the best $Delta$ value as about 50 seconds. An actual elderly faller could probably not get up by herself to activate the sensor network within 50 seconds after a fall occurs. If any event is generated within 50 seconds, the detected possible fall is most likely to be a false alarm and will be discarded. The remaining fall confidence is kept to generate the ROC curve as seen in Figure 8.6. We did not consider the case that the elderly adult had a real fall and a visitor was there to generate events in the sensor network. In this case we assumed that visitor would help the person who fell or call a staff member.

In Table 8.2, four falls were detected with a higher fall confidence (>0.9). Fall-1 and Fall-4 generated a lower confidence (<0.5) due to the weak radar signals. Fall-1 occurred around the detection range of the ceiling radar. Fall-4 is a kind of slow motion style fall, for which we assume the resident usually would not get a fatal injury. By applying the sensor network fusion framework, we could detect all those natural elderly falls by generating a lower false alarm number (even the fall confidence is small).

Table 8.2 Description of detected resident falls in a week

# Fall	Fall and accompanying activities (observed with Kinect depth image)
1	close front door
2	resident kneeled first, then fell to the side
3	visitor behind kitchen counter, cat run
4	kneeled first fall, sit on leg and fall
5	one leg kneel first
6	sliding away the wheel chair and fall

8.1.2 Extended Features

8.1.2.1 LBP Features

Besides the MFCC features, we also developed other features to improve the fall detection performance. One example is Local Binary Pattern (LBP) features. Local Binary Pattern features for signal processing are modified from the 2-D LBP [44, 45]. For the signal segment, x[i] is the selected center among its P neighboring samples. The LBP code on x[i] is computed with

$$LBP_P(x[i]) = \sum_{p=0}^{\frac{P}{2}-1} \left(S\left(x\left[i+p-\frac{P}{2}\right]-x[i]\right)2^p + S(x[i+p+1]-x[i])2^{p+\frac{P}{2}}\right), \quad (8.6)$$

where S is the Sign function and defined as

$$S[x] = \begin{cases} 1\ for\ x \geq 0 \\ 0\ for\ x < 0 \end{cases}. \quad (8.7)$$

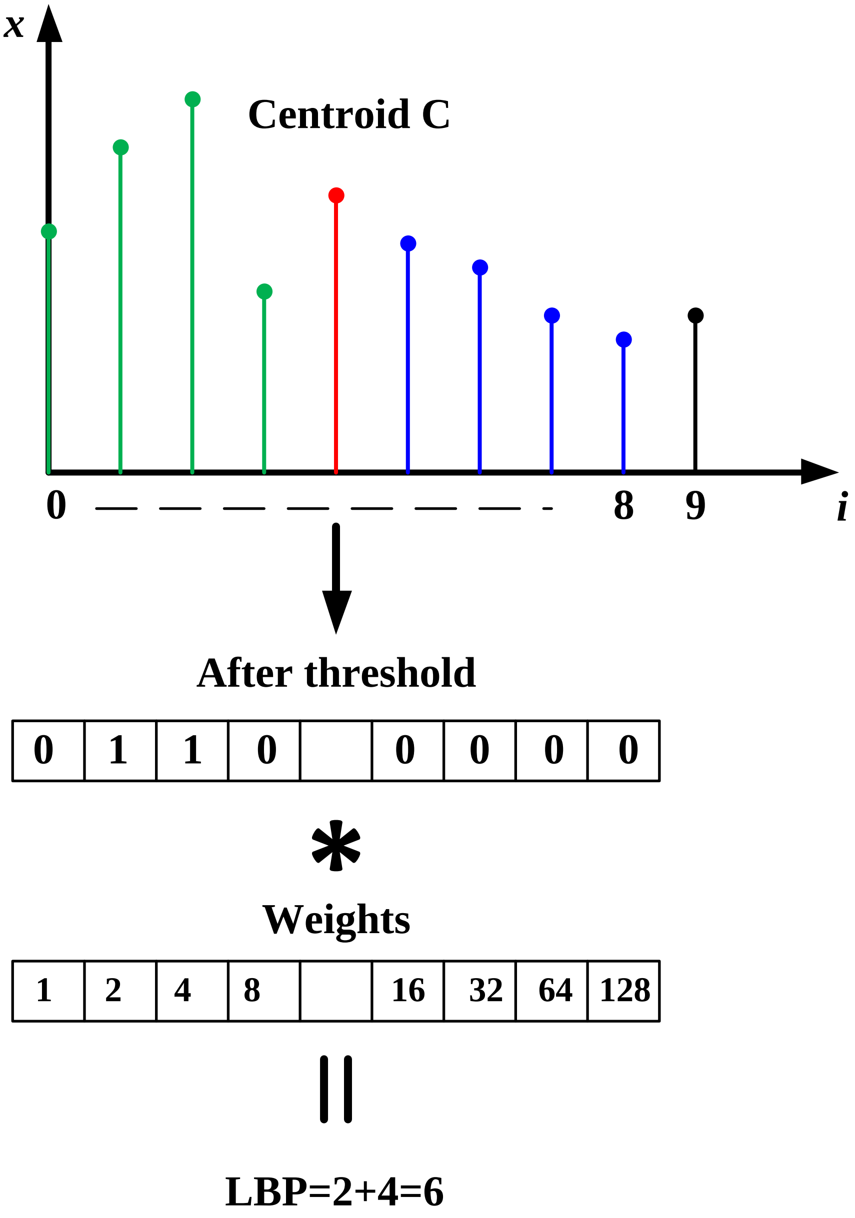


Figure 8.7 The generation of 1-D LBP code with P neighbors

We set P=8 which has eight neighbors around the center sample x[i] as shown in Figure 8.7. For the signal segment with a data length of L, i is in the range [P/2: L-P/2]. For 2-s radar sensor data, we had L=960*2=1920 and i=[4: 1916]. The difference of the P neighbors to the center sample was converted to a P-bit binary code by equation (8.7). The LBP code reflects the difference of the local sample to its neighbors. For a less varying signal, those differences are close to zero. For peaks and valleys, the difference is relative large. At edges, the difference in some directions could be larger than those from other directions. The local patterns could be defined with the distribution of LBP codes

$$H_k = \sum_{\frac{P}{2} \leq i \leq L-P/2} \delta(LBP_P(x[i]), k), \quad k = 1, \ldots, n. \qquad (8.7)$$

where k and n are the number of the histogram bins. Each bin corresponds to a LBP code. δ is the Kronecker delta function which is given by

$$\delta(i,j) = \begin{cases} 1 & i == j \\ 0 & i \neq j \end{cases}. \qquad (8.8)$$

The LBP_P generates 2^P different LBP codes corresponding to 2^Pdifferent binary patterns formed by P neighbors.

The rotation invariant patterns LBP_P^r is defined by

$$LBP_P^r = \min\{ROR(LBP_P, i)\}, \qquad i = 0, \ldots, P-1. \quad (8.9)$$

$ROR(LBP_P, i)$ performed a bit-wise right shift on P-bit number LBP_P i times. The LBP_P^r is the minimum value was found by shifting the LBP_P code for the P neighbors. The processed window produced the same code for all shifted codes which is invariant to rotation. Figure 8.8 shows the same example in Figure 8.7. After the rotation, the minimum value was found as 3. So the rotation invariant binary patterns are 1100_0000.

Rotation

0	**1**	**1**	**0**		**0**	**0**	**0**	**0**	**6**
0	**0**	**1**	**1**		**0**	**0**	**0**	**0**	**12**
0	**0**	**0**	**1**		**1**	**0**	**0**	**0**	**24**
0	**0**	**0**	**0**		**1**	**1**	**0**	**0**	**48**
0	**0**	**0**	**0**		**0**	**1**	**1**	**0**	**96**
0	**0**	**0**	**0**		**0**	**0**	**1**	**1**	**192**
1	**0**	**0**	**0**		**0**	**0**	**0**	**1**	**129**
1	**1**	**0**	**0**		**0**	**0**	**0**	**0**	**3**

Figure 8.8 The procedure to generate 1-D rotation invariant and uniform patterns for LBP code with P neighbors $LBP_P^{r,u} = 3$.

The uniform patterns LBP_P^u define the LBP code which has at most two one-to-zero or zero-to-one transitions. Each uniformed pattern was assigned to a different histogram bin and all the un-uniformed patterns were assigned to the same histogram bin.

A uniformity measure U was introduced to describe the number of spatial 0/1 transitions in the “patterns.” Patterns 00101000 and 00000000 have U value of 4 and 0, respectively. The first pattern is non-uniform pattern since the U value is larger than 2. Thus, we defined this operator as follows

$$LBP_P^{r,u}(x[i]) =$$

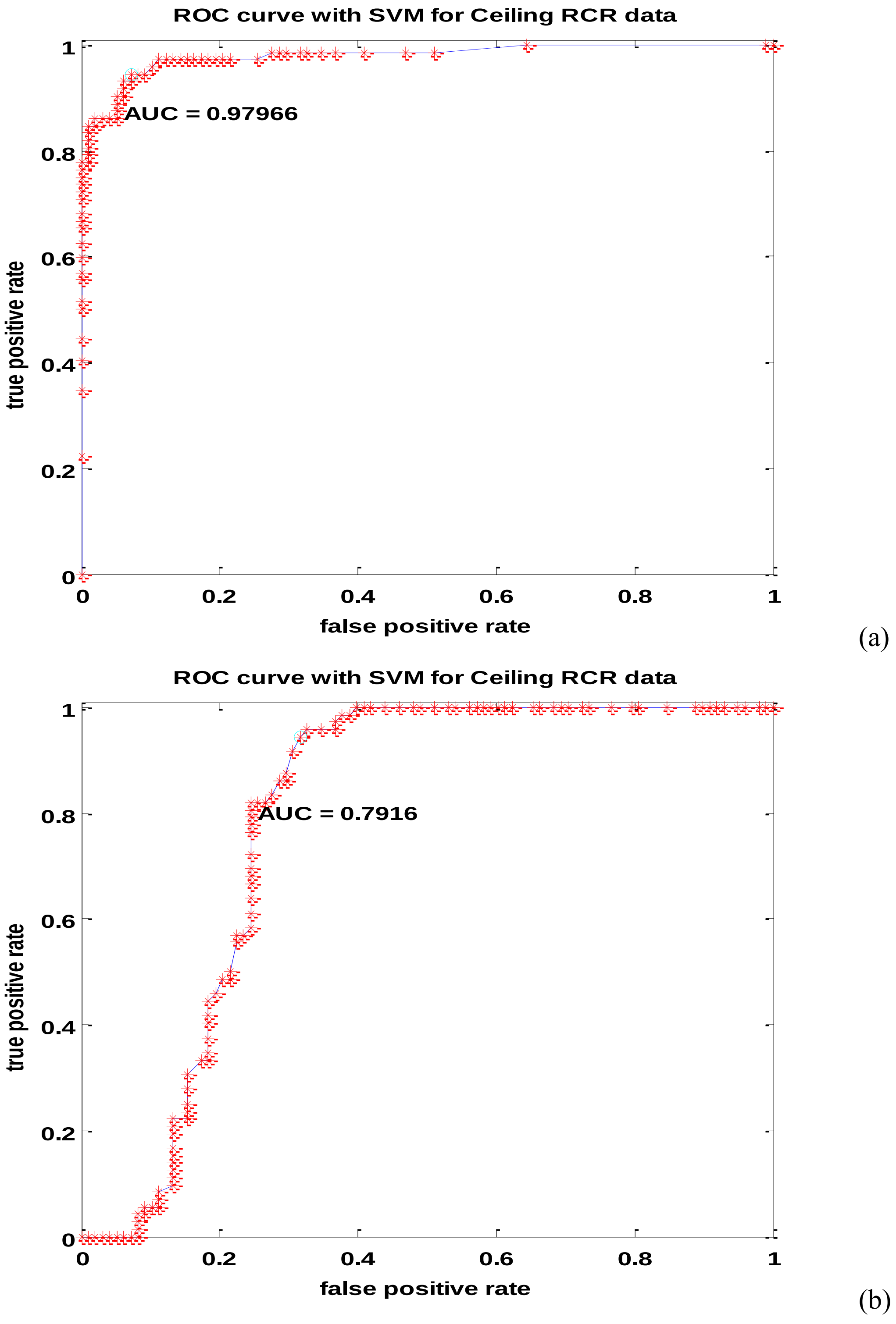
$$\begin{cases} \sum_{p=0}^{P/2-1}(S(x\left[i+p-\frac{P}{2}\right]-x[i])+S(x[i+p+1]-x[i])) & if\ U(LBP_P) \leq 2 \\ P+1 & otherwise \end{cases}. \quad (8.10)$$

The example in Figure 8.8 has two 0/1 transitions and it is a uniformed pattern. The $LBP_P^{r,u}$ binary codes shift until they achieve the minimum values and end up with a histogram having P+1 bins for uniform patterns plus one bin for non-uniform patterns. We used the normalized histograms for $LBP_P^{r,u}$with P+2=10 bins. So our implementation of the LBP features hadthe dimension of 10.

8.1.2.2 Experimental Results

We used the dataset described in Table 8.1 to test the performance of LBP features. For comparison, the MFCC feature is presented also in Figure 8.9. Each 2-s signal could generate 330 MFCC features and 10 LBP features.

We find that the MFCC features outperformed the LBP features on the same dataset since the MFCC-AUC is 0.98 which is larger than LBP-AUC of 0.79. MFCC features system could detect 97% falls with a lower false alarm rate at 10%. However, we notice that LBP features enabled the system to detect all the falls with a false alarm rate above 40% whereas the system with MFCC features could only detect all the falls if the false alarm rate was above 65%. Based on not missing any falls, the LBP features performed better.



(a)

(b)

Figure 8.9 The comparison of different features of the senior home data. (a) MFCC feature and (b) LBP feature.

8.1.2.3 Features Selection

Considering the results in Figure 8.8, we have several reasons to consider both of the features in our applications. First, the starting point of the two features is different. The MFCC features represent the short-term power spectrum of the signal. The LBP features could reveal the varying of the signal. Second, the feature dimension is quite different since MFCC features are 33 times larger than LBP features. Although the feature dimension is not directly proportional to the classification performance, the larger dimension may provide more information.

To solve this dilemma, we need to estimate the quality of features at each dimension, also called attribute. RELIFE has been used to fulfill this task [128]. We combined the MFCC and LBP features together and evaluated the efficiency or quality of features with the RELIFE algorithm developed by Kira and Rendell for estimating attributes. It reports how well the attribute values distinguish among instances that are near to each other. RELIEF searches the two nearest neighbors for the given instance. One distance is from the same class (nearest hit) and the other is from a different class (nearest miss). Good attributes should differentiate between instances from different classes and should be equal to each other for the instances from the same class. The estimation of the attribute A is approximated in the form of the probability differences.

$$W(A) = Prob(different\ value\ of\ A|nearest\ instance\ from\ different\ class) - P(different\ value\ of\ A|nearest\ instance\ from\ same\ class)\ . \qquad (8.11)$$

In Figure 8.10, the RELIEF algorithm has been implemented in two ways to improve the system performance. The exhaustive searching range for feature dimensions is from 5 to 340 with an interval of 5. The RELIEF algorithm first performs as a weighting device shown on the blue curve. The high quality of an attribute corresponds to the large weight. For the example at the 5th feature dimension on x-axis, the 5th dimension features have the five largest weights among all 340 features. The big improvement happens at dimension 250 with an AUC value as 0.9744. The largest AUC 0.9803 occurs at the dimension length of 340. The RELIEF algorithm acts as a factor to emphasize the efficiency of each attribute. Those attributes are still selected according to the order of the weights. The selected attributes multiply their corresponding weights to form the new features. The performance shows with the red curve. The performance improves dramatically at the dimension of 55 with an AUC of 0.8250. The best performance appears with the dimension of 340 having an AUC value of 0.9811.

Both of the two above featured handling methods could improve system performance. The largest AUC increased from 0.9797 in Figure 8.8 to 0.9811 in Figure 8.10.

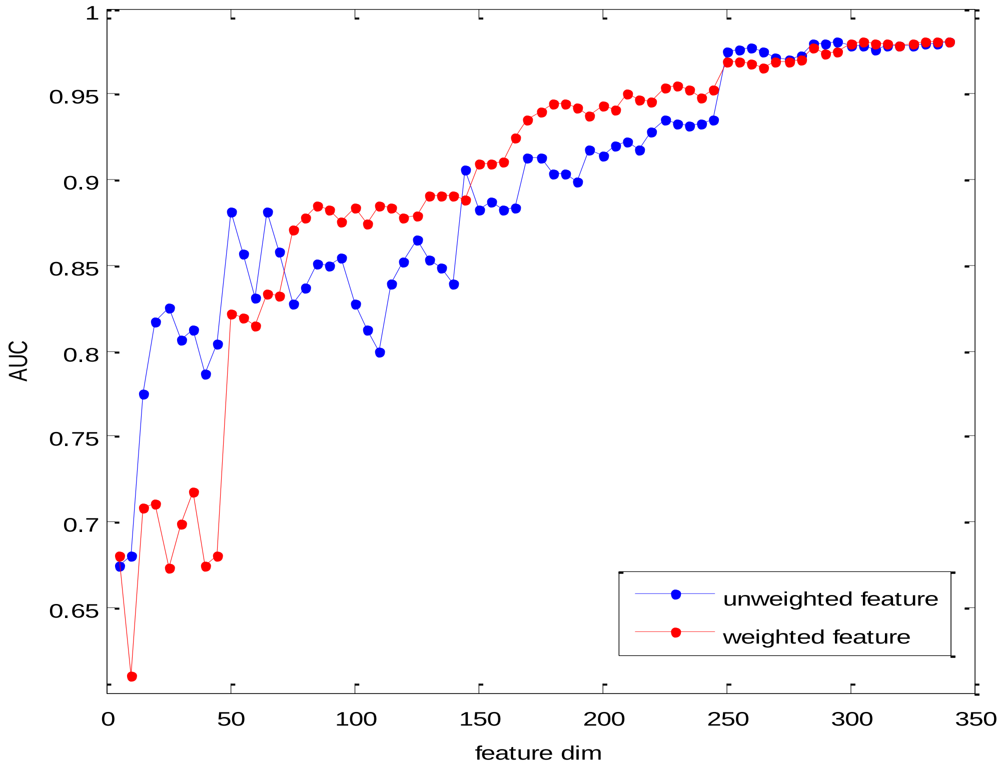


Figure 8.10 The feature selection using RELIEF algorithm

8.2 Real-world Data Processing

After we deployed the Doppler radar sensors to the senior homes, each senior home generated continuous radar signals with a 960 Hz sampling frequency. To process those real-world data more efficiently, we first filtered the data to find out the possible data segments where falls are most likely to happen.

8.2.1 Filtering

For efficiency, the raw Doppler radar signal needs to be pre-screened before entering the classifier for further processing. We also call this pre-screen step “filtering out the interest points which have the higher possibility of containing falls.” We developed the pre-screener based on the energy burst curve.

The energy burst curve was introduced in chapter 4 as a tool used in locating potential falls on the coming unknown signal. The lab collected data generated the energy burst curve using a frequency range from 25 to 50 Hz, which represents the fall on the corresponding energy burst curve. Ideally, those potential falls could be pointed out by those positions of the detected peaks on the energy burst curve. However, the effective frequency range for generating the energy burst curve can differ when the radar environment changes from lab to senior homes.

After the Doppler radar was moved to the senior apartments at Tiger Place, we re-investigated different frequency ranges on those fall data sequences collected in the new environment. Figures 8.11-8.13 show signal segments collected each month from three different bathrooms in those apartments. The data was collected with Sinclair School of Nursing trained stunt actors came from results processed in four different months. On each figure, the upper subplot shows the energy burst curve marked with red stars, indicating the detected possible falls after running the peak detection. The bottom curve is the Doppler radar signal waveform with the ground truth of the fall occurrence timestamps indicated by those green dotted lines.

The selection of different frequency ranges produced different energy burst curves. A good range was established after exhaustive searching for both maximum and minimum boundaries on the frequency spectrogram. The parameter searching was performed from four months of bathroom site data in three different senior homes. The structure, floor plans and surroundings are unique for each the bathroom. The frequency range of the desirable performance is between 30 and 300Hz. The pre-screen results of t Subjects 1, 2 and 3 are presented in Figures 8.11, 8.12 and 8.13, respectively. The sub caption from (a) to (d) in each figure represents the data collected in different months from Jan to May in 2013. The April data is absent due to the abnormal device status caused by discontinuous power supplies.

For each signal segment threshold, the energy burst curve was programmed to throw away all the non-fall activities under the same conditions such as temperature, surrounding staff, furniture displacements, *etc*. The threshold value is computed based on the amplitude range of the energy burst curve on the current signal segment. The threshold value for energy burst curve is selected by using

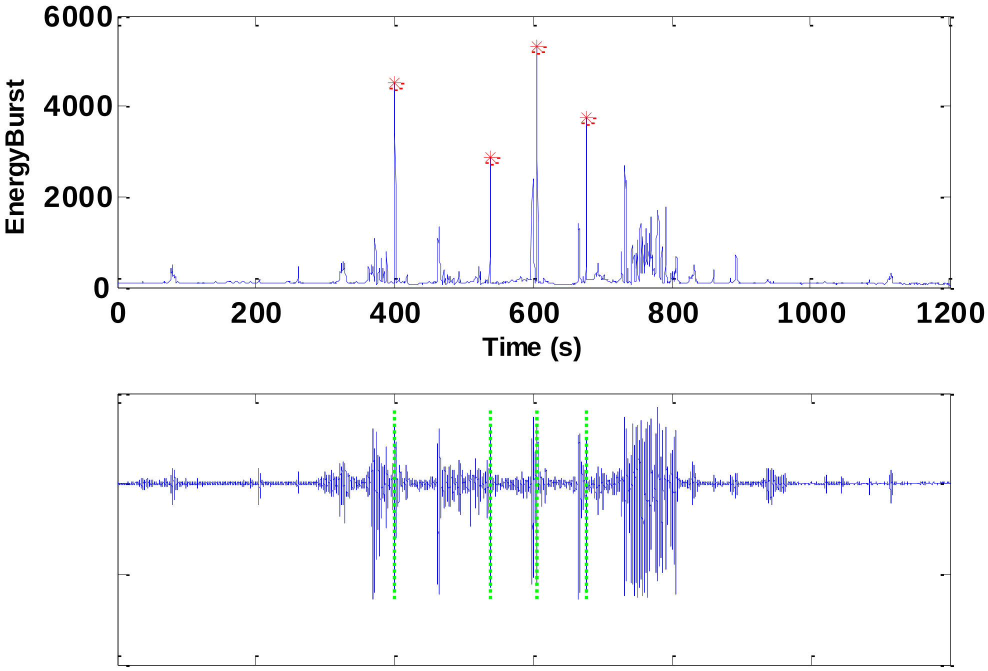
$$threshold = \beta[\max(y) - \min(y)] + \min(y), \quad (8.12)$$

where β is the weight factor with a constraint as $0 \leq \beta \leq 1$.

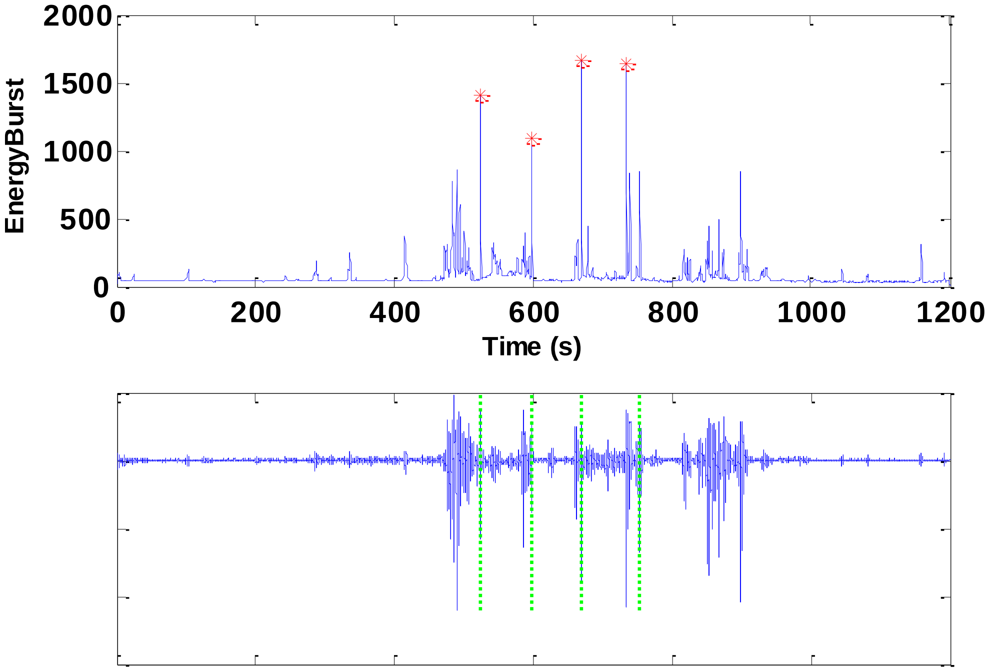
Among those pre-screen results, three signal segments have shown the perfect pre-screen results in Figure 8.11(a) - (b) and in Figure 8.12 (d). They locate all the falls accurately without including any false alarms. One extreme case is shown in Figure 8.13(c). The first fall showed a very weak strength on the energy burst curve. In order to include this fall, the β is set lower as 0.1. In this case, the false alarms increasedto three. The last false alarm of around 1000 s would not appear if the threshold was higher.

Another special case is the weak signal strength during the data collection period in Figure 8.12(b). The signal amplitude is around ±0.3V in this Feb. 15 data segment. In order to obtain the second fall after pre-screen, the β was set as 0.2. The same radar had a signal strength around ±2 V in the May 24 data segment as shown in Figure 8.12(d). The radar was setup above the living room ceiling. The temperature on Feb 15 was below -1°

F before noon. The attic could get warmer instantly. The working temperature was lower around zero. However, the operating temperature for radar was between 0˚ F and 50˚F. For the remaining seven data sequences, all the falls could be located exactly with 1-3 false alarms.


6000
4000
2000
0
EnergyBurst
0
200
400
600
800
1000
1200
Time (s)

(a)


2000
1500
1000
500
0
EnergyBurst
0
200
400
600
800
1000
1200
Time (s)

(b)

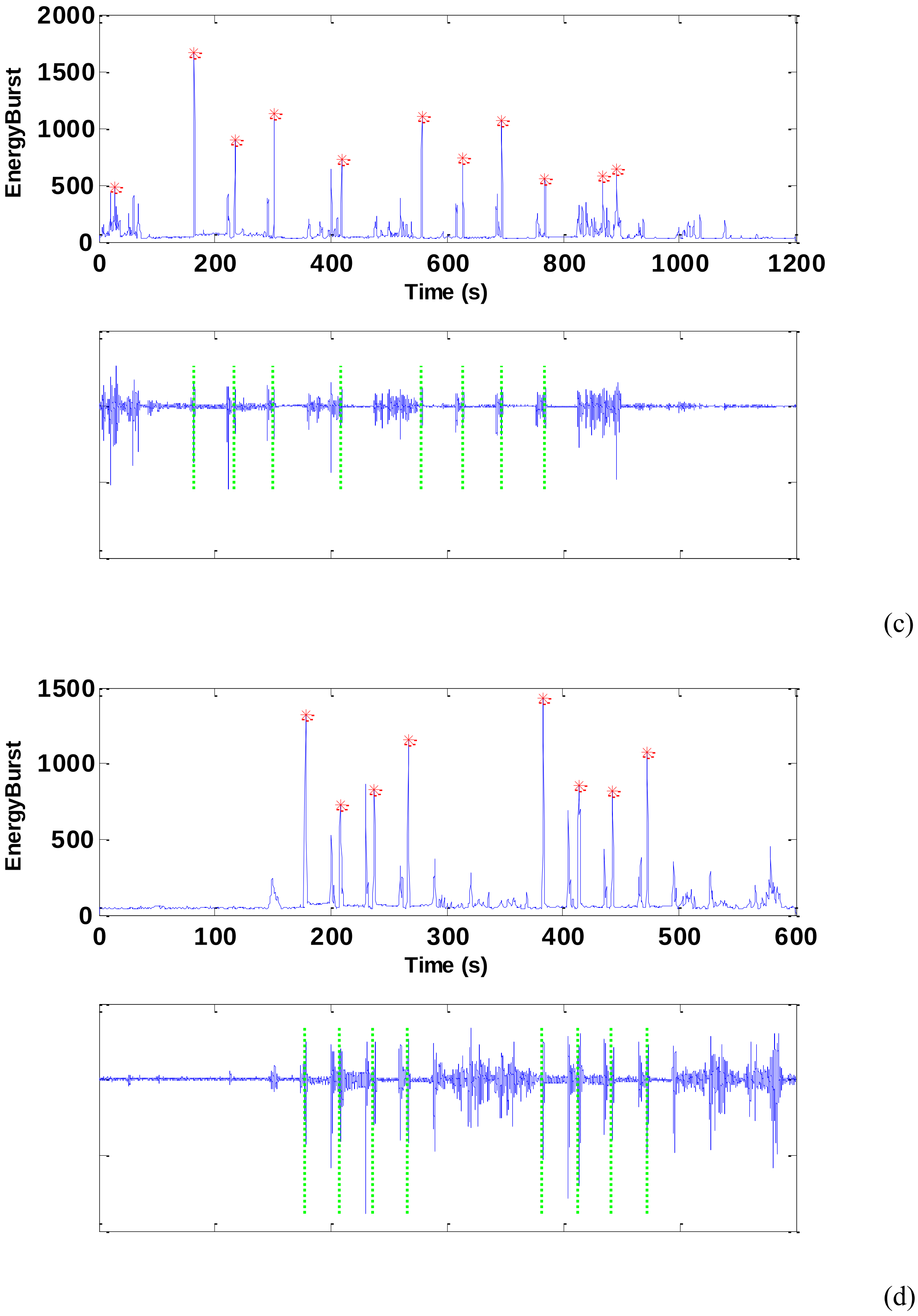
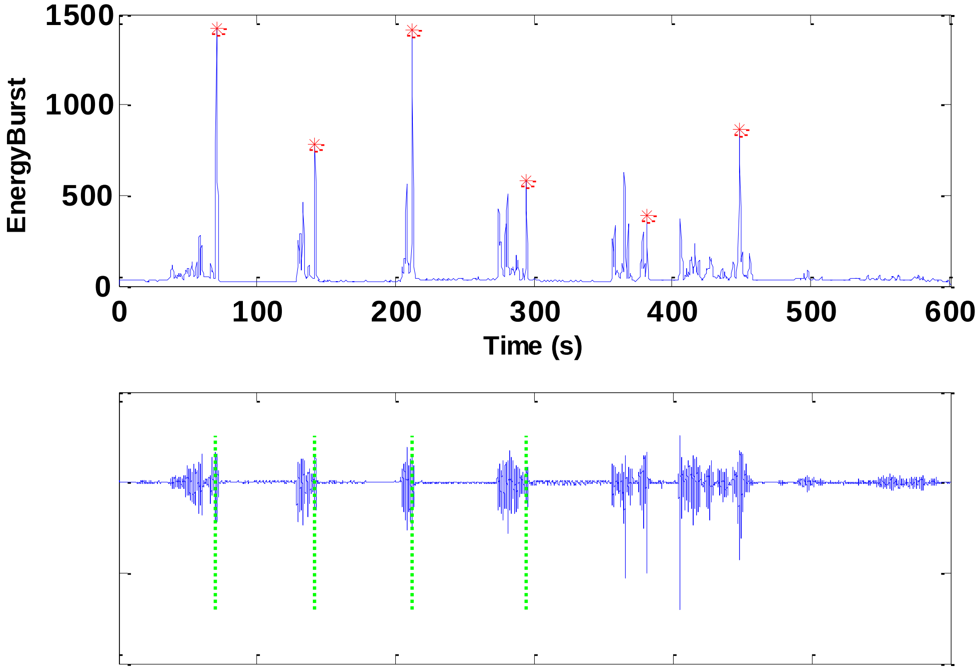
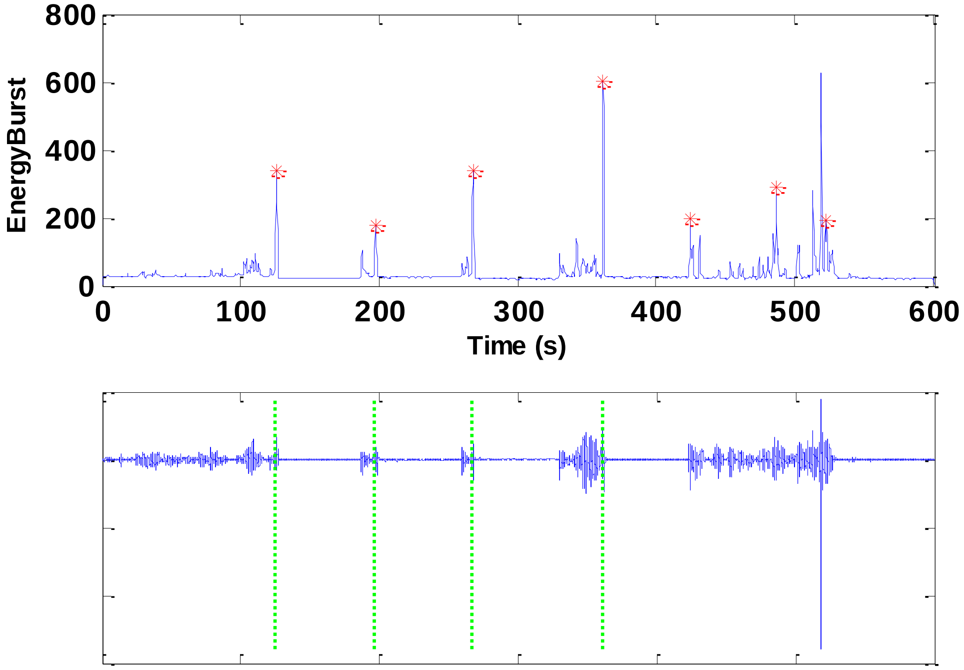


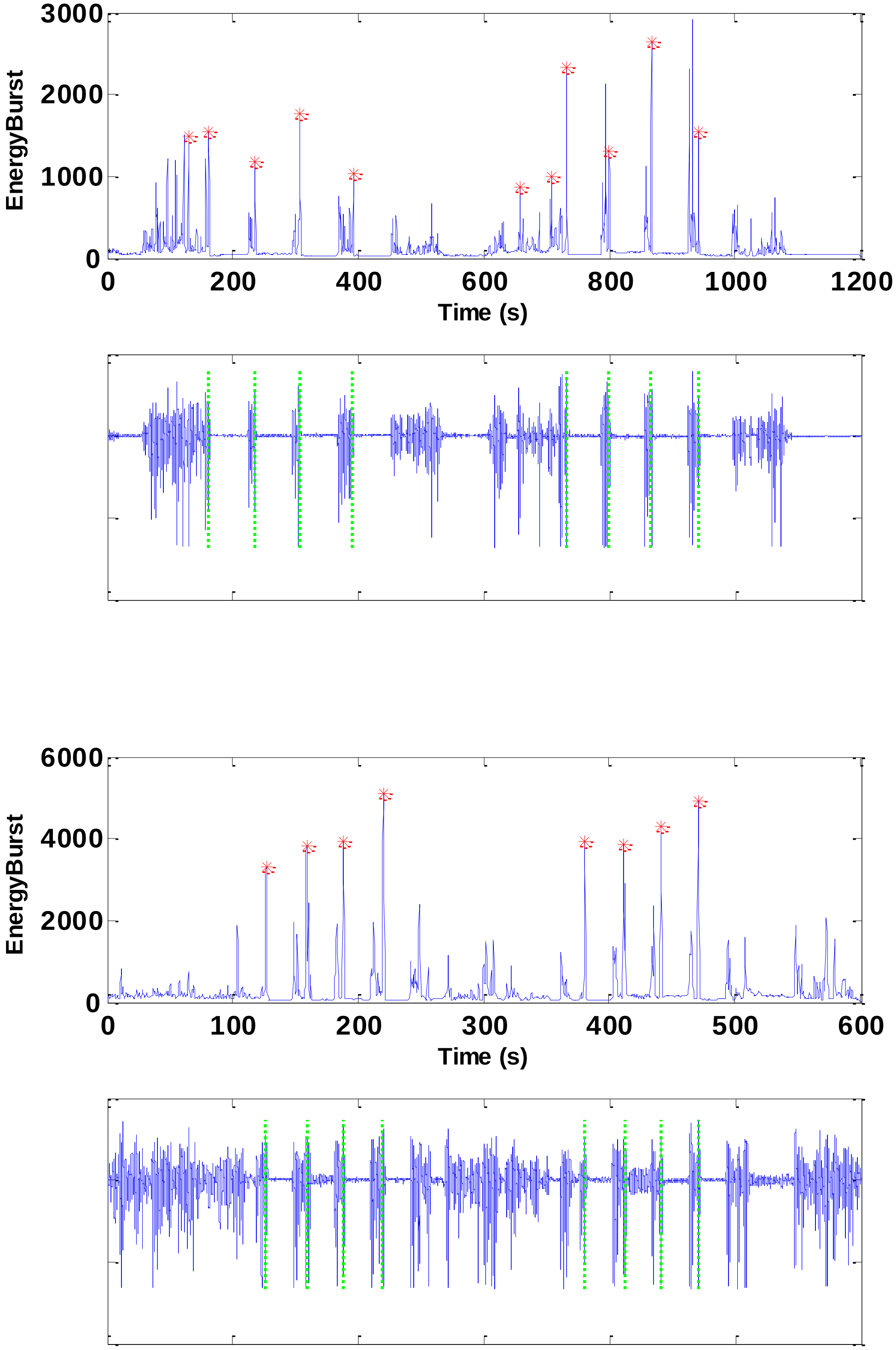
Figure 8.11 The pre-screening with energy burst curve for subject 3013


1500
1000
500
0
EnergyBurst
0
100
200
300
400
500
600
Time (s)

(a)


800
600
400
200
0
EnergyBurst
0
100
200
300
400
500
600
Time (s)

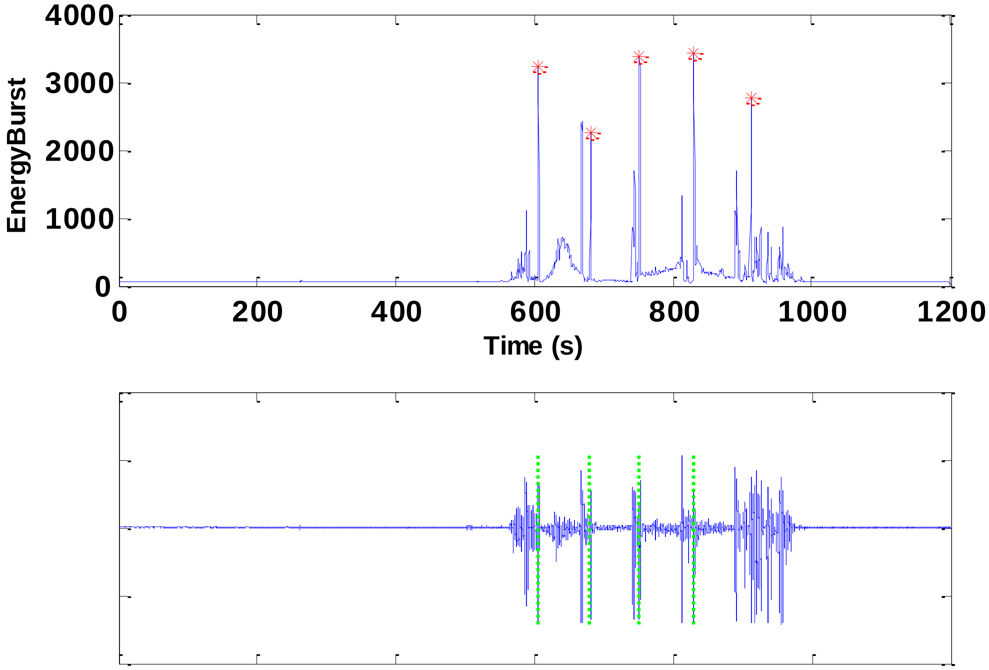
(b)



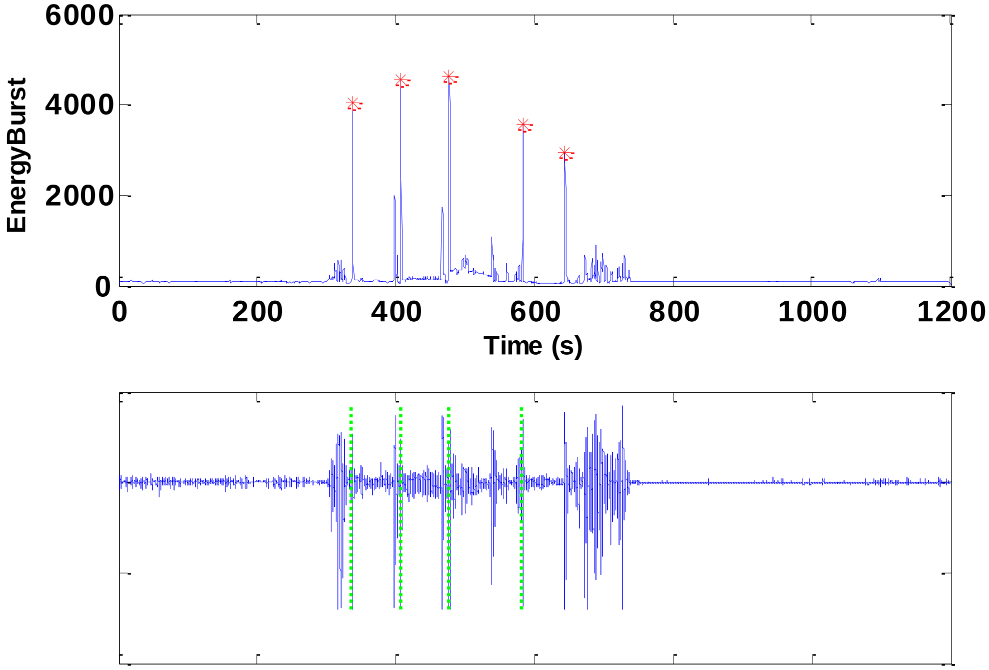
(c)

(d)

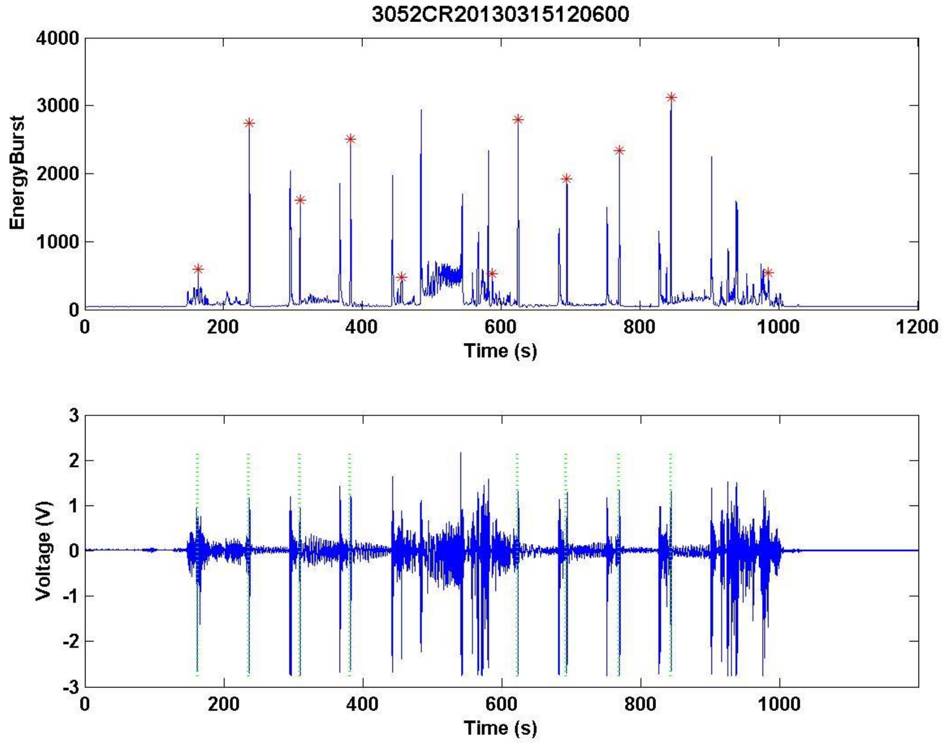
Figure 8.12 The pre-screening with energy burst curve for subject 3017


4000
3000
2000
1000
0
EnergyBurst
0
200
400
600
800
1000
1200
Time (s)

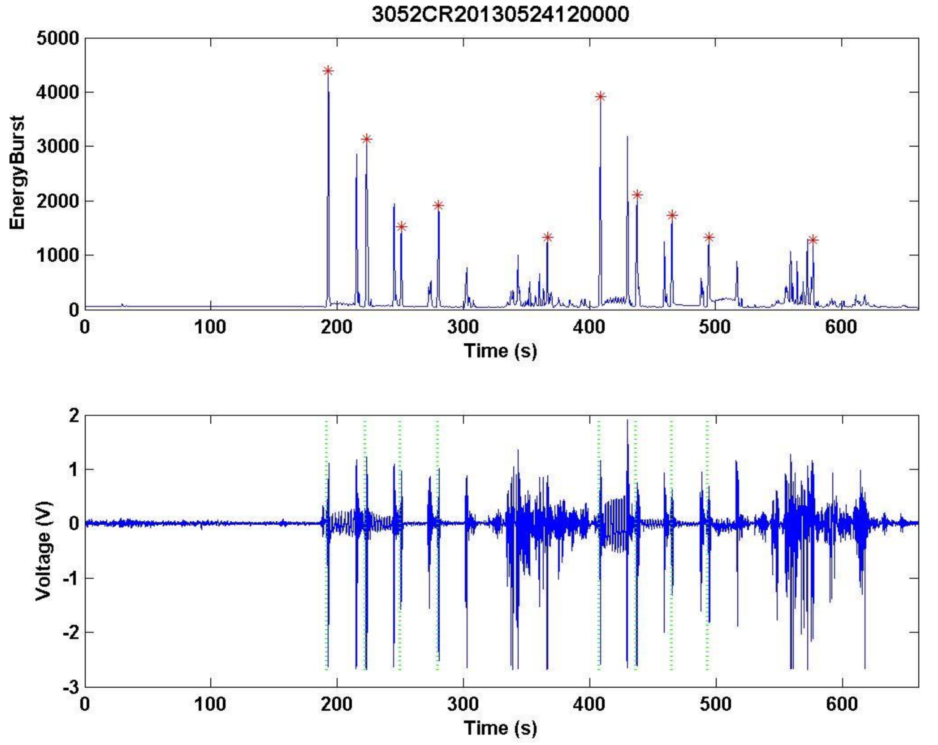
(a)


6000
4000
2000
0
EnergyBurst
0
200
400
600
800
1000
1200
Time (s)

(b)



(c)



(d)

Figure 8.13 The pre-screening with energy burst curve for subject 3052

Table 8.3 Summary of the result from the fall pre-screener

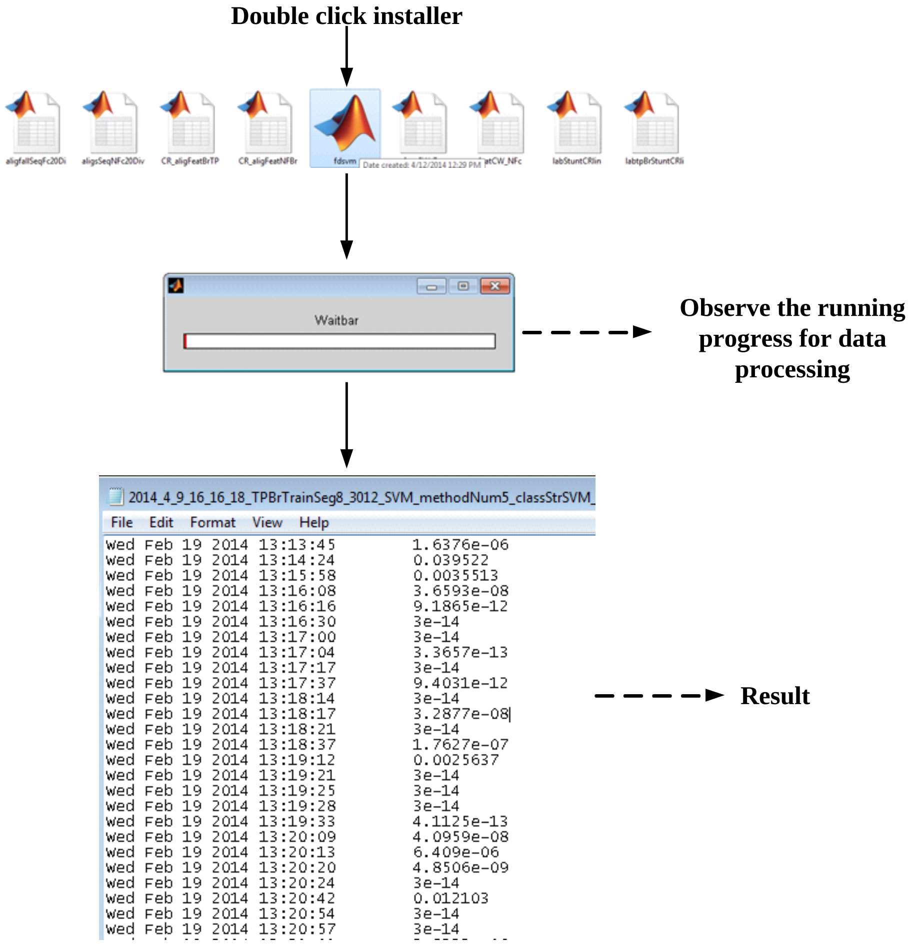
Num	Subject	#Actual fall (Ground truth)	#Detected fall	#False alarm
1	3013	4	4	0
		4	4	0
		8	11	3
		8	9	1
2	3017	4	6	2
		4	7	3
		8	11	3
		8	8	0
3	3052	4	5	1
		4	5	1
		8	11	3
		8	10	2
	Total	72	91	19

Table 8.3 gives the summary on the results from the fall pre-screener. The total data length is about 191 minutes. So we have (19 false alarms /191min) = 9.95% false alarms/ min. Although the false alarm/min does not seem very low, we should notice that the signal segments have lots of motions from the research staff around data collection. Many things can confuse radar, such as the stunt actor moving up and down the mattress and furniture, stretching legs and arms, quickly stooping to pick up things on floor, *etc*. These activities would not happen a lot in a normal private bathroom. By increasing the denominator for time, the false alarm/min would be very smaller in daily life.

8.2.2 The Running Radar Fall Detection System

A running radar fall detection system has been implemented and verified in a standalone working mode on a desktop computer. The standalone software could continuously process data recorded from senior homes. The input data format is in a ".wdq" file. The system output result is saved in a text file. The text file is then imported to the server, where the user can check the fall confidence for a specific subject during certain periods or at a specific time via a website. For nursing staff, this kind of fall information is valuable to find the cause of a fatal fall or to discover the possibility of a potential senior fall victim.

(1) Standalone mode of the fall detection system: To generate the standalone mode of the monitoring system, a MATLAB Compiler generates the software installer. The installer could run on the desktop without installing the MATLAB and thereby avoiding the software license issue for multiple machines. Each of those desktop would need MATLAB Compiler Runtime (MCR) to run the code.

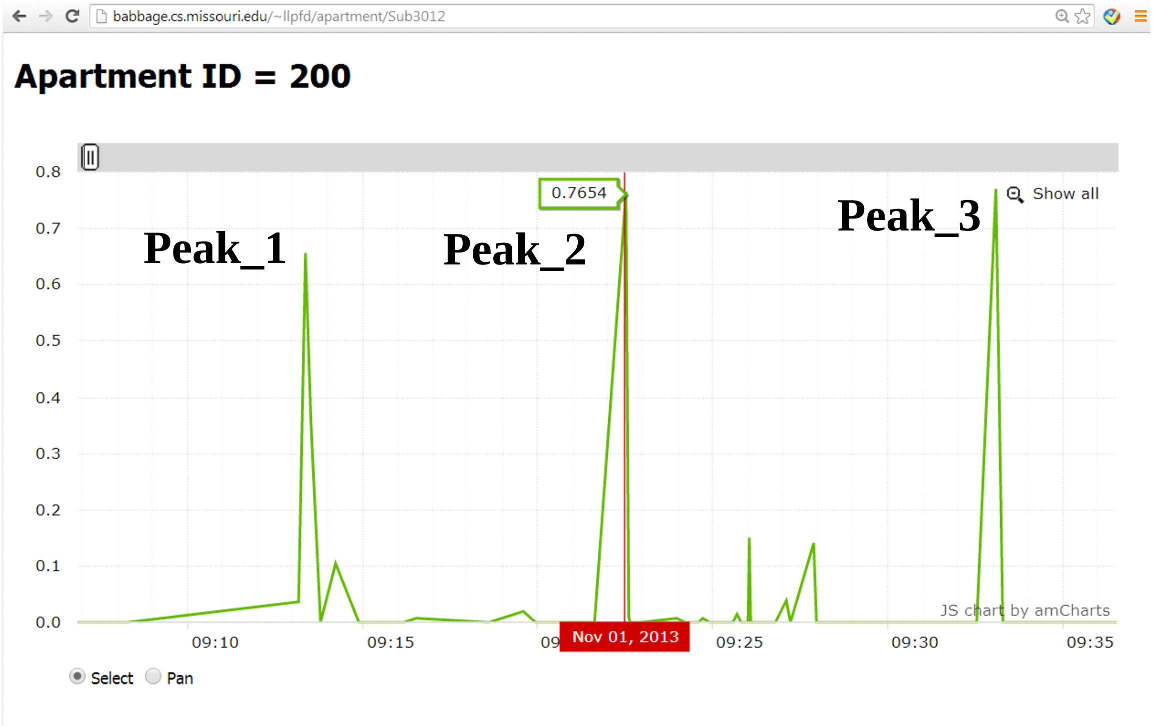


```
Wed Feb 19 2014 13:13:45          1.6376e-06
Wed Feb 19 2014 13:14:24          0.039522
Wed Feb 19 2014 13:15:58          0.0035513
Wed Feb 19 2014 13:16:08          3.6593e-08
Wed Feb 19 2014 13:16:16          9.1865e-12
Wed Feb 19 2014 13:16:30          3e-14
Wed Feb 19 2014 13:17:00          3e-14
Wed Feb 19 2014 13:17:04          3.3657e-13
Wed Feb 19 2014 13:17:17          3e-14
Wed Feb 19 2014 13:17:37          9.4031e-12
Wed Feb 19 2014 13:18:14          3e-14
Wed Feb 19 2014 13:18:17          3.2877e-08
Wed Feb 19 2014 13:18:21          3e-14
Wed Feb 19 2014 13:18:37          1.7627e-07
Wed Feb 19 2014 13:19:12          0.0025637
Wed Feb 19 2014 13:19:21          3e-14
Wed Feb 19 2014 13:19:25          3e-14
Wed Feb 19 2014 13:19:28          3e-14
Wed Feb 19 2014 13:19:33          4.1125e-13
Wed Feb 19 2014 13:20:09          4.0959e-08
Wed Feb 19 2014 13:20:13          6.409e-06
Wed Feb 19 2014 13:20:20          4.8506e-09
Wed Feb 19 2014 13:20:24          3e-14
Wed Feb 19 2014 13:20:42          0.012103
Wed Feb 19 2014 13:20:54          3e-14
Wed Feb 19 2014 13:20:57          3e-14
```

Figure 8.14 A running fall detection system in standalone mode

The recorded Doppler Radar data could be processed automatically by clicking the installer (dark orange convex surface with light blue background icon) in Figure 8.14. The progress of the data processing could be observed with the Waitbar. The result is saved in the format of the fall confidence with timestamp. The minimum timestamp interval is one second.

(2) Fall confidence displayed on a web interface: In Figure 8.15, a web interface presents the fall detection result using Java Script. In the selected time duration, there are three confidence peaks above 0.5. The red cursor shows the time stamp on a fall confidence for the second peak. This peak happens at 9:22 AM on Nov/01/2013 with a fall confidence at 0.7654. The first peak has a confidence 0.6566. The confidence of the third peak is 0.7715. We can select the range for displaying the user interested segment and check the confidence value on the curve.



Apartment ID = 200

Figure 8.15 The fall confidence for a resident in senior home displayed on a web interface

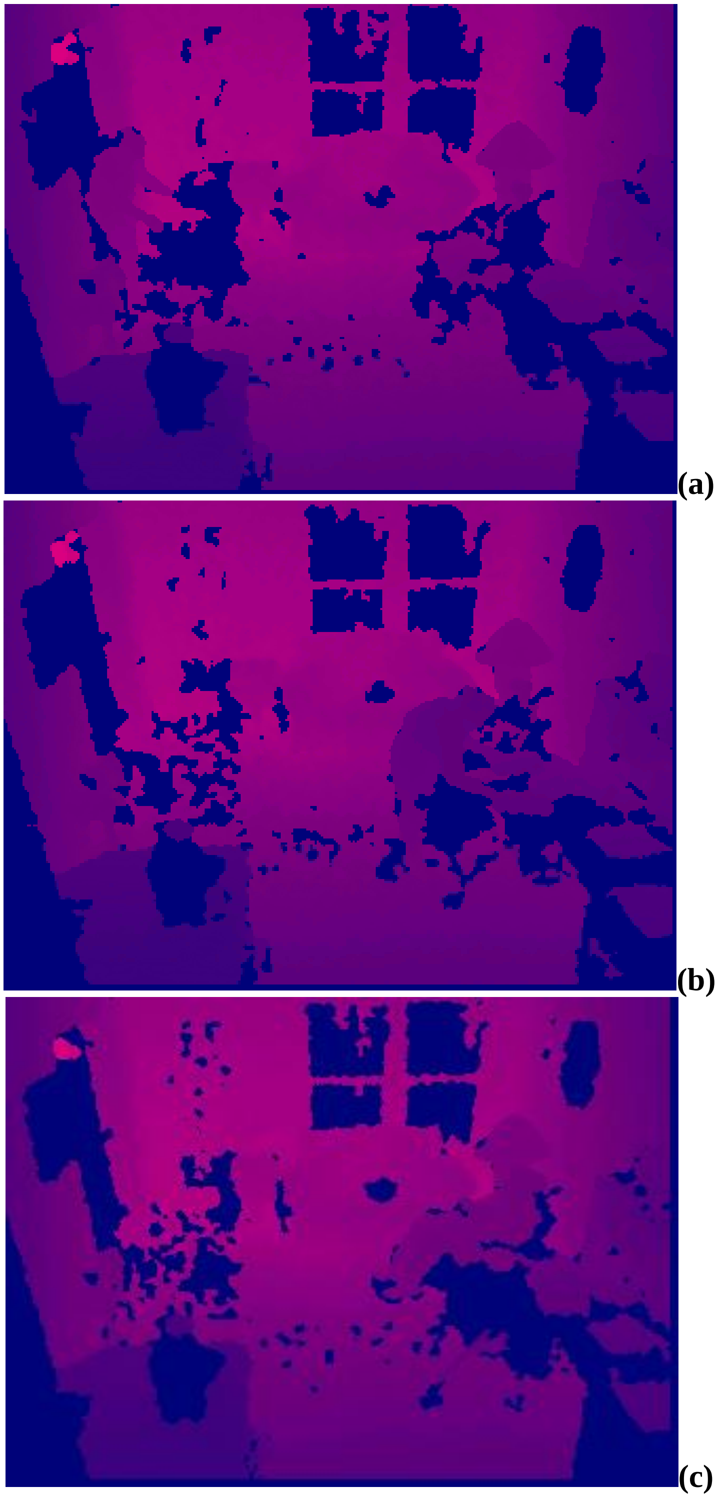


Figure 8.16 The activities corresponding to the high fall confidence values: (a)-(c) show the scene of Peak_1, Peak_2 and Peak_3, respectively.

The higher confidence means the detected data point has a larger possibility to be detected as a fall. We look back to the Kinect depth image to find out the ground truth for each of those peaks. In Figure 8.16 (a), the resident is walking with a walker from the bedroom to the living room at the left side of the image. Figure 8.16(b) shows that the resident is reaching something under a big cover below the ceiling radar on the middle to right image. In Figure 8.16 (c), the resident is sitting on a rolling chair and has raised s up both of her legs from floor to lean on the chair back.

8.3 Summary

I implemented an automatic fall detection system for an actual retired community apartment/home. I investigated different features: MFCC, LBP and the combined version of features with RELIEF algorithm. I also improved the fall detection performance for both pre-screener and features selection for the senior home data. I fused the radar fall detection system with other sensors, specifically motion sensors. I developed a standalone fall detection system and generated a result to display on a designed webpage. These results are quite promising. This system was designed to help nursing staff better understand information of the health status as it relates to falling for senior residents. In this way, seniors can be more healthy and safer in their independent living.

Bibliography

[1] S.L. Murhy, “Deaths: Final Data for 1998,” *National Vital Statistics Reports*, vol. 48, no. 11. Hyattsville, Maryland: National Center for Health Statistics, 2000.

[2] J.A. Stevens, G. Ryan, M. Kresnow, “Fatalities and injuries from falls among older adults – United States, 1993–2003 and 2001–2005,” MMWR, Nov. 2006, 55(45).

[3] R.J. Gurley, N. Lum, M. Sande, B. Lo, M.H. Katz, “Persons found in their homes helpless or dead,” *N Engl J Med*. 334(26):1710-6, 1996.

[4] C.G. Moran, R.T. Wenn, M. Sikand, A.M. Taylor, "Early mortality after hip fracture: is delay before surgery important," *J. of Bone and Joint Surgery*, pp. 483-9, 2005.

[5] "A Report to the National Livable Communities: Creating Environments for Successful Aging"[online]. Retrieved 2014-03-10.

[6] D. Giansanti, G. Maccioni and V. Macellari, “The development and test of a device for the reconstruction of 3D position and orientation by means of a kinematic sensor assembly with rate gyroscopes and accelerometers,” *IEEE Trans. Biomed. Eng.* vol. 52, no. 7, pp 1271-1277, July 2005.

[7] N. Noury, A. Fleury, P. Rumeau, *et al*., "Fall detection-principles and methods," *Proc. of the 29th IEEE EMBS*, Lyon, France, Aug. 23-26,

[8] M. Alwan, P.J. Rajendran, S. Kell, D. Mack, S. Dalal, M. Wolfe, R. Felder, "A smart and passive floor-vibration based fall detector for elderly," *2nd IEEE Int. Conf. on Inf. & Comm. Tech.: from Theory to Applications - ICTTA’06*, April 24 - 28, 2006, Damascus, Syria.

[9] D. Anderson, R.H. Luke, J. Keller, M. Skubic, M. Rantz, M. Aud, "Linguistic summarization of activities from video for fall detection using voxel person and fuzzy logic," *Computer Vision and Image Understanding*, vol. 113 (1), January 2009, pp. 80-89

[10] A. Sixsmith, N. Johnson, R. Whatmore, "Pyrolitic IR sensor arrays for fall detection in the older population," *J. Phys. IV France*, vol. 128.

[11] M. Addlesee, A. Jones, F. Livesey, and F. Samaria, "Orl active floor," in *IEEE Personal Communications*, vol. 4.5, 1997, pp. 35 – 41.

[12] M. Popescu, S. Coupland, "A fuzzy logic system for fall detection," AAAI Fall Symposium, Washington DC, Nov 7-9 2008, pp. 78-83.

[13] M. Popescu, A. Mahnot, "Acoustic fall detection using one-class classifier," 31th Int. IEEE EMBS Conf., Vancouver, BC, 2009, pp. 3505-3508.

[14] M. Popescu, Y. Li, M. Skubic, M. Rantz, "An acoustic fall detector system that uses sound height information to reduce the false alarm rate," 30th Int. IEEE EMBS Conf., Vancouver, BC, Aug. 20-24, 2008, pp. 4628-4631.

[15] J.L. Geisheimer, W.S. Marshall, E. Greneker, "A continuous-wave (CW) radar for gait analysis," *Signals, Systems and Computers*, vol. 1, pp. 834-838, 2001.

[16] Youngwook Kim, Hao Ling, "Human Activity Classification Based on Micro-Doppler Signatures Using a Support Vector Machine," *Geoscience and Remote Sensing, IEEE Transactions on* , vol.47, no.5, pp.1328-1337, May 2009

[17] S. Dura-Bernal, G. Garreau, C. Andreou, A. Andreou, J. Georgiou, T. Wennekers, S. Denham. "Human action categorization using ultrasound micro-Doppler signatures," in *Human Behavior Understanding Lecture Notes in Computer*

Science, *Vol. 7065*, 2011, Eds., A. A. Salah and B. Lepri, HBU 2011, LNCS 7065, Springer-Verlag, Berlin Heidelberg, pp 1-15.

[18] E. J. Hughes, M. Lewis, "The Application of Speech Recognition Techniques to Radar Target Doppler Recognition: A Case Study," *High Resolution Imaging and Target Classification, 2006. The Institution of Engineering and Technology Seminar on* , vol., no., pp.145-152, 21-21 Nov. 2006

[19] T. Yardibi, P. Cuddihy, S. Genc, C. Bufi, M. Skubic, M. Rantz, L. Liu; C. Phillips, "Gait characterization via pulse-Doppler radar," *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on* , vol., no., pp.662-667, 21-25 March 2011

[20] L. Liu, M. Popescu, M. Skubic, M. Rantz, T. Yardibi, P. Cuddihy, "Automatic Fall Detection Based on Doppler Radar Motion," *Proceedings, 5th International Conference on Pervasive Computing Technologies for Healthcare*, pp. 222-225, Dublin, Ireland, May 23-26, 2011.

[21] L. Liu, M. Popescu, M. Rantz, M. Skubic, "Fall Detection using Doppler Radar and Classifier Fusion," *Proceedings of EMBS on BHI*, pp. 180-183, Shenzhen, Jan 2-7, 2012.

[22] L. Liu, M. Popescu, M. Rantz, M. Skubic, "Doppler Radar Sensor Positioning in a Fall Detection System," *Proceedings of 34th EMBS*, pp. 256-259, San Diego, August 28- Septemper 1, 2012.

[23] L. Liu, M. Popescu, M. Skubic, M. Rantz, "An automatic fall detection framework using data fusion of Doppler radar and motion sensor network," *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE* , pp.5940-5943, 26-30 Aug. 2014

[24] M. Mercuri, P.J. Soh, G. Pandey, P. Karsmakers, G.A.E. Vandenbosch, P. Leroux, D. Schreurs, "Analysis of an Indoor Biomedical Radar-Based System for Health Monitoring," *Microwave Theory and Techniques, IEEE Transactions on* , vol.61, no.5, pp.2061,2068, May 2013

[25] C. Garripoli, M. Mercuri, P. Karsmakers, P. Soh, G. Crupi, G. Vandenbosch, C. Pace, P. Leroux, D. Schreurs, "Embedded DSP-based Telehealth Radar System for Remote In-door Fall Detection," *Biomedical and Health Informatics, IEEE Journal of* , vol.PP, no.99, pp.1,1

[26] B. Mager, N. Patwari, M. Bocca, "Fall detection using RF sensor networks," *Personal Indoor and Mobile Radio Communications (PIMRC), 2013 IEEE 24th International Symposium on* , pp.3472-3476, 8-11 Sept. 2013

[27] O. Kaltiokallio, M. Bocca, and N. Patwari, "Follow @grandma: Long-term device-free localization for residential monitoring," *Local Computer Networks Workshops (LCN Workshops), 2012 IEEE 37th Conference on* , vol., no., pp.991,998, 22-25 Oct. 2012

[28] J. Wilson and N. Patwari, "A fade level skew-Laplace signal strength model for device-free localization with wireless networks," IEEE Trans. Mobile Computing, vol. 11, no. 6, pp. 947-958, June 2012, appeared online 12 May 2011.

[29] S. Cagnoni, G. Matrella, M. Mordonini, F. Sassi, L. Ascari, "Sensor Fusion-Oriented Fall Detection for Assistive Technologies Applications," *Intelligent Systems Design and Applications, 2009. ISDA '09. Ninth International Conference on* , vol., no., pp.673,678, Nov. 30 2009-Dec. 2 2009

[30] S. Wang, M. Skubic, Y. Zhu, "Activity Density Map Visualization and Dis-similarity Comparison for Eldercare Monitoring," IEEE Journal of Biomedical and Health Informatics, 2012, 16(4):607-614.

[31] F. Wang, M. Skubic, M. Rantz, T. Yardibi & PE Cuddihy, "Quantitative Gait Measurement with Pulse-Doppler Radar for Passive In-home Gait Assessment," IEEE Transactions on Biomedical Engineering, 2014, 61(9):2434-2443.

[32] David Brandwood, Fourier transforms in Radar and Signal Processing, 2nd ed. Artech House, 2011.

[33] T. Ganchev, N. Fakotakis, G. Kokkinakis, "Comparative Evaluation of Various MFCC Implementations on the Speaker Verification Task," Proc. of the SPECOM-2005, October 17-19, 2005. Patras, Greece. Vol. 1, pp. 191-194

[34] http://neural.cs.nthu.edu.tw/jang/books/audioSignalProcessing/speechFeatureMfcc.asp?title=12-2 MFCC

[35] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[36] M. Rantz, M. Aud, G. Alexander, B. Wakefield, M. Skubic, R.H. Luke, D. Anderson, and J.K. Keller, "Falls, Technology, and Stunt Actors: new Approaches to Fall Detection and Fall Risk Assessment," Journal of Nursing Care Quality, 23(3):195-201, 2008.

[37] http://www.americareusa.net/independent_living/columbia_mo/zip_65201/americare/1335.html

[38] G.E. Smith, K. Woodbridge and C.J. Baker, "Naïve Bayesian radar micro-Doppler recognition," *Proc. Int. Conf. Radar*, p.111 , 2008.

[39] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification*, 2nd ed. John Wiley & Sons Inc., 2005

[40] M.J. Rantz, K.D. Marek, M.A. Aud, R.A. Johnson, D. Otto & R. Porter, "TigerPlace: A New Future for Older Adults," *Journal of Nursing Care Quality*, vol. 20, no. 1, pp. 1-4, January-March, 2005.

[41] P.E. Cuddihy, T. Yardibi, Z.J. Legenzoff, L. Liu, C.E. Phillips, C. Abbott, C. Galambos, J. Keller, M. Popescu, J. Back, M. Skubic & MJ Rantz, "Radar Walking Speed Measurements of Seniors in their Apartments: Technology for Fall Prevention," Proceedings of 34th EMBS, San Diego CA, August 28-September 1, 2012, pp 260-263.

[42] D.J. Lipman, S.F. Altschul, J.D. Kececioglu, "A tool for multiple sequence alignment," Proc Natl Acad Sci USA June 1989; 86(12): 4412–4415.

[43] J.C. Augusto, H. Nakashima, H. Aghajan, “Ambient Intelligence and Smart Environments: A State of the Art,” in the Handbook of Ambient Intelligence and Smart Environments, pp 3-31, 2010.

[44] M. Heikkila, M. Pietikainen, "A texture-based method for modeling the background and detecting moving objects," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* , vol.28, no.4, pp.657,662, April 2006

[45] Q. Zhu, N. Chatlani, J.J. Soraghan, "1-D Local binary patterns based VAD used INHMM-based improved speech recognition," Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European , vol., no., pp.1633,1637, 27-31 Aug. 2012

[46] J.Y. Hwang, J.M. Kang, Y.W. Jang, H.C. Kim, "Development of novel algorithm and real-time monitoring ambulatory system using Bluetooth module for fall detection in the elderly," Engineering in Medicine and Biology Society, 2004. IEMBS '04. 26th Annual International Conference of the IEEE, vol.1, no., pp.2204-2207, 1-5 Sept. 2004

[47] M. Wu, X. Dai, Y.D. Zhang, B. Davidson, M.G. Amin, J. Zhang, "Fall Detection Based on Sequential Modeling of Radar Signal Time-Frequency Features," *Healthcare Informatics (ICHI), 2013 IEEE International Conference on* , vol., no., pp.169,174, 9-11 Sept. 2013

[48] J. Chen, K. Kwong, D. Chang, J. Luk, R. Bajcsy, "Wearable Sensors for Reliable Fall Detection," *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the* , vol., no., pp.3551,3554, 17-18 Jan. 2006

[49] V. Vishwakarma, C. Mandal, S. Sural, “Automatic detection of human fall in video,” Lect Notes Comput Sci Pattern Recognit Mach Intell. 2007;4815:616–623.

[50] G. Mastorakis, D. Makris, “Fall detection system using Kinect’s infrared sensor,” Journal of Real-Time Image Processing. 2014. vol9, issue4, pp635–646

[51] R. Cucchiara, A. Prati, R. Vezzani, “A multi-camera vision system for fall detection and alarm generation,” Expert Systems, 2007;24:334–345.

[52] A. Edgcomb, F. Vahid, “Privacy perception and fall detection accuracy for in-home video assistive monitoring with privacy enhancements,” Spec Interest Group Health Inform Rec. 2012;2(2):6–15.

[53] K.M. Culhane, M. O’Connor, D. Lyons, G.M Lyons, “Accelerometers in rehabilitation medicine for older adults,” Age Ageing. 2005;34:556–560.

[54] A.K. Bourke, J.V. O’Brien, G.M. Lyons, “Evaluation of a threshold-based triaxial accelerometer fall detection algorithm,” Gait Posture. July 2007, vol. 26, page194–199.

[55] M. Kangas, I. Vikman, J. Wiklander, P. Lindgren, L. Nyberg, T. Jämsä, “Sensitivity and specificity of fall detection in people aged 40 years and over,” Gait Posture, June 2009, vol.29, pp571–574.

[56] F. Bianchi, S.J. Redmond, M.R Narayanan, S. Cerutti, N.H. Lovell, "Barometric Pressure and Triaxial Accelerometry-Based Falls Event Detection," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* , vol.18, no.6, pp.619,627, Dec. 2010

[57] M. Yuwono, B.D. Moulton, S.W. Su, B.G. Celler, H. T. Nguyen, “Unsupervised machine-learning method for improving the performance of ambulatory fall-detection systems,” *BioMedical Engineering OnLine* 2012;11:9.

[58] H. Kerdegari, K. Samsudin, A.R. Ramli, S. Mokaram, "Evaluation of fall detection classification approaches," *Intelligent and Advanced Systems (ICIAS), 2012 4th International Conference on* , vol.1, no., pp.131,136, 12-14 June 2012

[59] J. Cheng, X. Chen, M. Shen, "A Framework for Daily Activity Monitoring and Fall Detection Based on Surface Electromyography and Accelerometer Signals," *Biomedical and Health Informatics, IEEE Journal of* , vol.17, no.1, pp.38,45, Jan. 2013

[60] N.D. Lane, E. Miluzzo, H. Lu, D. Peeble, T. Choudhury, A.T. Campbell, "A survey of mobile phone sensing," *Communications Magazine, IEEE* , vol.48, no.9, pp.140,150, Sept. 2010

[61] M.V. Albert, K. Kording, M. Herrmann, A. Jayaraman, “Fall classification by machine learning using mobile phones,” PLoS One. 2012;7:e36556.

[62] I. Plaza, L. Martin, S. Martin, C. Medrano, "Mobile applications in an aging society: Status and trends," Journal of Systems Software, 2011, volume 84, issue 11: pp1977–1988.

[63] M. Kangas, I. Vikman, L. Nyberg, R. Korpelainen, J. Lindblom, T. Jms, "Comparison of real-life accidental falls in older people with experimental falls in middle-aged test subjects," Gait Posture, 2012;35:500–505.

[64] G. Diraco, A. Leone, P. Siciliano, "An active vision system for fall detection and posture recognition in elderly healthcare," *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2010* , vol., no., pp.1536,1541, 8-12 March 2010

[65] L. Hazelhoff, J. Han, and P.H. With, "Video-Based Fall Detection in the Home Using Principal Component Analysis," In *Proceedings of the 10th International Conference on Advanced Concepts for Intelligent Vision Systems* (ACIVS '08), Jacques Blanc-Talon, Salah Bourennane, Wilfried Philips, Dan Popescu, and Paul Scheunders (Eds.). Springer-Verlag, Berlin, Heidelberg, 298-309.

[66] A. Pantelopoulos, N.G. Bourbakis, "A Survey on Wearable Sensor-Based Systems for Health Monitoring and Prognosis," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* , vol.40, no.1, pp.1,12, Jan. 2010

[67] A. Lanata, E.P. Scilingo, E. Nardini, G. Loriga, R. Paradiso, D. De-Rossi, "Comparative evaluation of susceptibility to motion artifact in different wearable systems for monitoring respiratory rate," *Information Technology in Biomedicine, IEEE Transactions on* 2010, 14:378-386.

[68] L. Fernández, J.M. Blasco, J.F. Hernández, "Wireless sensor networks in ambient intelligence," In *Book Wireless Sensor Networks in Ambient Intelligence*. City: Univ. Politécnica de Valencia: Technologies for Health and Well-being Institute ITACA; 2009.

[69] H.G. Kang, D.F. Mahoney, H. Hoenig, V.A. Hirth, P. Bonato, I. Hajjar, LA. Lipsitz, “In situ monitoring of health in older adults: technologies and issues,” *J Am Geriatr Soc* 2010, 58:1579-1586.

[70] D.H. Stefanov, Z. Bien, W.C. Bang, “The smart house for older persons and persons with physical disabilities: structure, technology arrangements, and perspectives,” *Neural Systems and Rehabilitation Engineering, IEEE Transactions on* 2004, 12:228-250.

[71] A. Muro-de-la-Herran, B. Garcia-Zapirain, A. Mendez-Zorrilla, “Gait Analysis Methods: An Overview of Wearable and Non-Wearable Systems, Highlighting Clinical Applications,” *Sensors* 2014, *14*, 3362-3394.

[72] AT. Özdemir, B. Barshan, “Detecting Falls with Wearable Sensors Using Machine Learning Techniques,” *Sensors*. 2014; 14(6):10691-10708.

[73] H. Banaee, M.U. Ahmed, A. Loutfi, “Data Mining for Wearable Sensors in Health Monitoring Systems: A Review of Recent Trends and Challenges,” *Sensors* 2013, *13*, 17472-17500.

[74] S. Begum, S. Barua, M. U. Ahmed, "Physiological Sensor Signals Classification for Healthcare Using Sensor Data Fusion and Case-Based Reasoning," *Sensors* 14, no. 7: 11770-11785.

[75] S. Tao, M. Kudo, H. Nonaka, “Privacy-Preserved Behavior Analysis and Fall Detection by an Infrared Ceiling Sensor Network,” *Sensors* 2012, *12*, 16920-16936.

[76] M.T. Yang, M.W. Chuang, “Fall Risk Assessment and Early-Warning for Toddler Behaviors at Home,” *Sensors* 2013, 13, 16985

[77] A. Darwish, A.E. Hassanien, “Wearable and Implantable Wireless Sensor Network Solutions for Healthcare Monitoring,” *Sensors* 2011, *11*, 5561-5595.

[78] V. Custodio, F.J. Herrera, G. López, J.I. Moreno, “A Review on Architectures and Communications Technologies for Wearable Health-Monitoring Systems,” *Sensors* 2012, *12*, 13907-13946.

[79] M.A. Habib, M.S. Mohktar, S.B. Kamaruzzaman, K.S. Lim, T.M. Pin, F. Ibrahim, “Smartphone-Based Solutions for Fall Detection and Prevention: Challenges and Open Issues,” *Sensors* 2014, *14*, 7181-7208.

[80] S. Pate, H. Park, P. Bonato, L. Chan and M. Rodgers, “A review of wearable sensors and systems with application in rehabilitation,” *Journal of NeuroEngineering and Rehabilitation* 2012, 9:21

[81] E. Jovanov, A. Milenkovic, “Body area networks for ubiquitous healthcare applications: opportunities and challenges,” *J Med Syst* 2011, 35:1245-1254

[82] M.J. Mathie, A.C. Coster, N.H. Lovell, B.G. Celler, S.R. Lord, A. Tiedemann, ”A pilot study of long-term monitoring of human movements in the home using accelerometry,” *J Telemed Telecare* 2004, 10:144-151.

[83] E.S. Sazonov, G. Fulk, N. Sazonova, S. Schuckers, “Automatic recognition of postures and activities in stroke patients,” *Conf Proc IEEE Eng Med Biol Soc* 2009, 2009:2200-2203.

[84] D. Giansanti, G. Maccioni, S. Morelli, ”An experience of health technology assessment in new models of care for subjects with Parkinson's disease by means of a new wearable device,” *Telemed J E Health* 2008, 14:467-472.

[85] O. Amft, G. Troster “Recognition of dietary activity events using on-body sensors,” *Artif Intell Med* 2008, 42:121-136.

[86] M.G. Benedetti, A. Di Gioia, L. Conti, L. Berti, L.D. Esposti, G. Tarrini, N. Melchionda, S. Giannini, “Physical activity monitoring in obese people in the real life environment,” *J Neuroeng Rehabil* 2009, 6:47.

[87] A. Sciacqua, M. Valentini, A. Gualtieri, F. Perticone, A. Faini, G. Zacharioudakis, I. Karatzanis, F. Chiarugi, C. Assimakopoulou, P. Meriggi, *et al*. “Validation of a flexible and innovative platform for the home monitoring of heart failure patients: preliminary results,” *Computers in Cardiology* 2009, 36:97-100.

[88] J. Merilahti, J. Parkka, K. Antila, P. Paavilainen, E. Mattila, E.J. Malm, A. Saarinen, I. Korhonen, “Compliance and technical feasibility of long-term health monitoring with wearable and ambient technologies,” *J Telemed Telecare* 2009, 15:302-309

[89] M. Sung, C. Marci, A. Pentland, “Wearable feedback systems for rehabilitation,” *J Neuroeng Rehabil* 2005, 2:17

[90] U. Anliker, J.A. Ward, P. Lukowicz, G. Troster, F. Dolveck, M. Baer, F. Keita, E.B. Schenker, F. Catarsi, L. Coluccini, A. Belardinelli, D. Shklarski, M. Alon, E. Hirt, R. Schmid, M. Vuskovic, "AMON: a wearable multiparameter medical monitoring and alert system," *Information Technology in Biomedicine, IEEE Transactions on* , vol.8, no.4, pp.415,427, Dec. 2004

[91] R. Paradiso, A. Alonso, D. Cianflone, A. Milsis, T. Vavouras, C. Malliopoulos, "Remote health monitoring with wearable non-invasive mobile system: The Healthwear project," *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE* , vol., no., pp.1699,1702, 20-25 Aug. 2008

[92] D.P. Tobón, T.H. Falk, M. Maier, "Context awareness in WBANs: a survey on medical and non-medical applications," *Wireless Communications, IEEE* , vol.20, no.4, pp.30,37, August 2013

[93] K. Ozcan, A.K. Mahabalagiri, M. Casares, S. Velipasalar, "Automatic Fall Detection and Activity Classification by a Wearable Embedded Smart Camera," *Emerging and Selected Topics in Circuits and Systems, IEEE Journal on* , vol.3, no.2, pp.125,136, June 2013

[94] N. Raveendranathan, S. Galzarano, V. Loseu, R. Gravina, R. Giannantonio, M. Sgroi, R. Jafari, G. Fortino, "From Modeling to Implementation of Virtual Sensors in Body Sensor Networks," *Sensors Journal, IEEE* , vol.12, no.3, pp.583,593, March 2012

[95] A.M. Bianchi, M.O. Mendez, S. Cerutti, "Processing of Signals Recorded Through Smart Devices: Sleep-Quality Assessment," *Information Technology in Biomedicine, IEEE Transactions on* , vol.14, no.3, pp.741,747, May 2010

[96] E.S. Sazonov, J.M. Fontana, "A Sensor System for Automatic Detection of Food Intake Through Non-Invasive Monitoring of Chewing," *Sensors Journal, IEEE* , vol.12, no.5, pp.1340,1348, May 2012

[97] T.W. Chen, Y.C. Su, K.Y. Huang, Y.M. Tsai, S.Y. Chien, L.G. Chen, "Visual Vocabulary Processor Based on Binary Tree Architecture for Real-Time Object Recognition in Full-HD Resolution," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* , vol.20, no.12, pp.2329,2332, Dec. 2012

[98] O. Pujol, D. Masip, "Geometry-Based Ensembles: Toward a Structural Characterization of the Classification Boundary," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* , vol.31, no.6, pp.1140,1146, June 2009

[99] C.D. Metcalf, R. Robinson, A.J. Malpass, T.P. Bogle, T.A. Dell, C. Harris, S.H. Demain, "Markerless Motion Capture and Measurement of Hand Kinematics: Validation and Application to Home-Based Upper Limb Rehabilitation," *Biomedical Engineering, IEEE Transactions on* , vol.60, no.8, pp.2184,2192, Aug. 2013

[100] J. Munson, P. Pasqual, "Using Technology in Autism Research: The Promise and the Perils," *Computer* , vol.45, no.6, pp.89,91, June 2012

[101] S. Obdrzalek, G. Kurillo, F. Ofli, R. Bajcsy, E. Seto, H. Jimison, M. Pavel, "Accuracy and robustness of Kinect pose estimation in the context of coaching of elderly population," *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE* , vol., no., pp.1188,1193, Aug. 28 2012-Sept. 1 2012

[102] E.E. Stone, M. Skubic, "Passive in-home measurement of stride-to-stride gait variability comparing vision and Kinect sensing," *Engineering in Medicine and Biology Society,EMBC, 2011 Annual International Conference of the IEEE* , vol., no., pp.6491,6494, Aug. 30 2011-Sept. 3 2011

[103] N.A. Borghese, M. Pirovano, R. Mainetti, P.L. Lanzi, "An integrated low-cost system for at-home rehabilitation," *Virtual Systems and Multimedia (VSMM), 2012 18th International Conference on* , vol., no., pp.553,556, 2-5 Sept. 2012

[104] F. Cordella, F. Di Corato, L. Zollo, B. Siciliano, P. van der Smagt, "Patient performance evaluation using Kinect and Monte Carlo-based finger tracking," *Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS & EMBS International Conference on* , vol., no., pp.1967,1972, 24-27 June 2012

[105] E.E. Stone, M. Skubic, "Unobtrusive, Continuous, In-Home Gait Measurement Using the Microsoft Kinect," *Biomedical Engineering, IEEE Transactions on* , vol.60, no.10, pp.2925,2932, Oct. 2013

[106] H. Haggag, M. Hossny, S. Haggag, S. Nahavandi, D. Creighton, "Safety applications using Kinect technology," *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on* , vol., no., pp.2164,2169, 5-8 Oct. 2014

[107] C.F. Crispim, V. Bathrinarayanan, B. Fosty, A. Konig, R. Romdhane, M. Thonnat, F. Bremond, "Evaluation of a monitoring system for event recognition of older people," *Advanced Video and Signal Based Surveillance (AVSS), 2013 10th IEEE International Conference on* , vol., no., pp.165,170, 27-30 Aug. 2013

[108] M. Martinez, R. Stiefelhagen, "Breath rate monitoring during sleep using near-ir imagery and PCA," *Pattern Recognition (ICPR), 2012 21st International Conference on* , vol., no., pp.3472,3475, 11-15 Nov. 2012

[109] B. Ni, N.C. Dat, P. Moulin, "RGBD-camera based get-up event detection for hospital fall prevention," *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on* , vol., no., pp.1405,1408, 25-30 March 2012

[110] D. Rivera-Gutierrez, R. Ferdig, J. Li, B. Lok, "Getting the Point Across: Exploring the Effects of Dynamic Virtual Humans in an Interactive Museum Exhibit on User Perceptions," *Visualization and Computer Graphics, IEEE Transactions on* , vol.20, no.4, pp.636,643, April 2014

[111] C. Schonauer, T. Pintaric, H. Kaufmann, S. Jansen – Kosterink, M. Vollenbroek-Hutten, "Chronic pain rehabilitation with a serious game using multimodal input," *Virtual Rehabilitation (ICVR), 2011 International Conference on* , vol., no., pp.1,8, 27-29 June 2011

[112] R. Cornejo, D. Hernandez, J. Favela, M. Tentori, S. Ochoa, "Persuading older adults to socialize and exercise through ambient games," *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2012 6th International Conference on* , vol., no., pp.215,218, 21-24 May 2012

[113] H. Kayama, S. Nishiguchi, M. Yamada, T. Aoyama, K. Okamoto, T. Kuroda, "Effect of a Kinect-based exercise game on improving executive cognitive performance in community-dwelling elderly," *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2013 7th International Conference on* , vol., no., pp.362,365, 5-8 May 2013

[114] A. Ejupi, M. Brodie, Y.J. Gschwind, D. Schoene, S. Lord, K. Delbaere, "Choice stepping reaction time test using exergame technology for fall risk assessment in older people," *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE* , vol., no., pp.6957,6960, 26-30 Aug. 2014

[115] M. Yu, Y.Z. Yu; A. Rhuma, S.M.R. Naqvi, L. Wang, J.A. Chambers, "An Online One Class Support Vector Machine-Based Person-Specific Fall Detection System for Monitoring an Elderly Individual in a Room Environment," *Biomedical and Health Informatics, IEEE Journal of* , vol.17, no.6, pp.1002,1014, Nov. 2013

[116] M. Yu, A. Rhuma, S.M. Naqvi, L. Wang, J. Chambers, "A Posture Recognition-Based Fall Detection System for Monitoring an Elderly Person in a Smart Home Environment," *Information Technology in Biomedicine, IEEE Transactions on* , vol.16, no.6, pp.1274,1286, Nov. 2012

[117] M. Yu, S.M. Naqvi, A. Rhuma, J. Chambers, "One class boundary method classifiers for application in a video-based fall detection system," *Computer Vision, IET* , vol.6, no.2, pp.90,100, March 2012

[118] Y. Zigel, D. Litvak, I. Gannot, "A Method for Automatic Fall Detection of Elderly People Using Floor Vibrations and Sound—Proof of Concept on Human Mimicking Doll Falls," *Biomedical Engineering, IEEE Transactions on* , vol.56, no.12, pp.2858,2867, Dec. 2009

[119] J. Wang, Z.Q. Zhang, B. Li, S. Lee, R.S. Sherratt, "An enhanced fall detection system for elderly person monitoring using consumer home networks," *Consumer Electronics, IEEE Transactions on* , vol.60, no.1, pp.23,29, February 2014

[120] D. Naranjo-Hernandez, L.M. Roa, J. Reina-Tosina, M.A. Estudillo-Valderrama, "Personalization and Adaptation to the Medium and Context in a Fall Detection System," *Information Technology in Biomedicine, IEEE Transactions on* , vol.16, no.2, pp.264,271, March 2012

[121] Y. Li, K.C. Ho, M. Popescu, "A Microphone Array System for Automatic Fall Detection," *Biomedical Engineering, IEEE Transactions on* , vol.59, no.5, pp.1291,1301, May 2012

[122] P.J. Soh, M. Mercuri, G. Pandey, G.A.E. Vandenbosch, D.M.M. Schreurs, "Dual-Band Planar Bowtie Monopole for a Fall-Detection Radar and Telemetry System," *Antennas and Wireless Propagation Letters, IEEE* , vol.11, no., pp.1698,1701, 2012

[123] X. Ma, H. Wang, B. Xue, M. Zhou, B. Ji, Y. Li, "Depth-Based Human Fall Detection via Shape Features and Improved Extreme Learning Machine," *Biomedical and Health Informatics, IEEE Journal of* , vol.18, no.6, pp.1915,1922, Nov. 2014

[124] J. Liu, T.E. Lockhart, "Development and Evaluation of a Prior-to-Impact Fall Event Detection Algorithm," *Biomedical Engineering, IEEE Transactions on* , vol.61, no.7, pp.2135,2140, July 2014

[125] C. Gu, C. Li, "From Tumor Targeting to Speech Monitoring: Accurate Respiratory Monitoring Using Medical Continuous-Wave Radar Sensors," *Microwave Magazine, IEEE* , vol.15, no.4, pp.66,76, June 2014

[126] M. Mercuri, D. Schreurs, P. Leroux, "Optimised waveform design for radar sensor aimed at contactless health monitoring," *Electronics Letters* , vol.48, no.20, pp.1255,1257, September 27 2012

[127] P.A. Cavalcante Aguilar, J. Boudy, D. Istrate, B. Dorizzi, J.C. Moura Mota, "A Dynamic Evidential Network for Fall Detection," *Biomedical and Health Informatics, IEEE Journal of* , vol.18, no.4, pp.1103-1113, July 2014

[128] Kira, Kenji and Rendell, Larry, “The feature selection problem: traditional methods and a new algorithm,”AAAI'92 Proceedings of the tenth national conference on Artificial intelligence, Pages 129-134 .

[129] Bilmes, Jeff A., “A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models,” Berkley, CA: International Computer Science Institute. pp. 7–13, 1998

[130] O. Cappé, "H2M : A set of MATLAB/OCTAVE functions for the EM estimation of mixtures and hidden Markov models", in http://tsi.enst.fr/~cappe/mfiles/h2m.tgz, Sept. 2002.

VITA

Liang Liu is born in Xianyang, China. She received her bachelor degree in Electrical Engineering at Harbin Institute of Technology, Harbin, China in 2003. She was an engineer in Xi’an Gaoke group corporation. She received her master degree in Control Engineering at Beijing University of Technology, Beijing, China in 2007. In the same year, she attended University of Missouri at Columbia for graduate study. She received her master degree on computer vision with the department of Electrical and Computer Engineering in 2009. After that, she researched on healthcare related projects and have sucessfully defensed her PhD dissertation in 2014 with the same department. Now she is a researcher on the project of remote health monitoring and activities recognition for hospital room at Northwestern University (Evanston, IL). Her research interests include machine learning, pattern recognition, signal processing, image processing, computer vision, healthcare technologies, and smart ambient.