8-1-2013

Vision-Based 3D Human Motion Analysis for Fall Detection and Bed-Exiting

Xiaoxiao Dai
University of Denver

Vision-based 3D Human Motion Analysis for Fall Detection and Bed-exiting

---

A Dissertation

Presented to

the Faculty of the Daniel Felix Ritchie School of Engineering and Computer Science

University of Denver

---

in Partial Fulfillment

of the Requirements for the Degree

of Master of Science

---

by

Xiaoxiao Dai

August, 2013

Advisor: Jun Zhang

Author: Xiaoxiao Dai
Title: Vision-based 3D Human Motion Analysis for Fall Detection and Bed-exiting
Advisor: Jun Zhang
Degree Date: August, 2013

Abstract

Fall is one of the most dangerous and costly accidents that threaten health of elderly people, and a large portion of falls occurs when a patient is trying to exit a bed. This thesis proposes two vision-based approaches for general fall detection and bed-exiting detection for elderly people, respectively. The Kinect sensor is chosen as the major monitoring device.

The first approach exploits the Kinect sensor with its Windows SDK to detect fall activities. The recorded spatial coordinates of the human body joints from Kinect's 3D skeletal view are processed to extract posture features. Then the principle component analysis and k-means clustering algorithms are applied for dimensionality reduction, vector quantization and feature translation. HMMs are well known for their application in temporal pattern recognition, thus they are chosen for this project to classify human motion which is a temporal sequence of postures. HMMs are trained by the labelled extracted features to model and discriminate four fall motion classes and three non-fall classes.

The second approach utilizes segmented motion history image (MHI) sequences to extract spatiotemporal features of a moving human body. Eight Hu image moments are calculated to translate the spatiotemporal features of each frame into vectors to describe video frames. The k-means clustering and HMM modelling are utilized for vector quantization and classification between bed-exiting activities and

rolling-on-bed activities. In addition, likelihood probability curves are generated along the time line of all MHIs, endeavoring to predict a bed-exiting activity.

Detailed descriptions of the experiments and result evaluation are documented in this thesis. The experimental results using human subjects verifies the feasibility and effectiveness of the proposed approaches for general fall detection and bed-exiting prediction.

Acknowledgements

I would like to express my gratitude to my advisor Dr. Jun Zhang for his invaluable guidance in my research work during the passed whole year.

Also, I want to thank Dr. Bradley Davidson and Dr. Mohammad H. Mahoor for their great help and all the advises and suggestions for refinement of this thesis.

Last but not least, I would like to thank my beloved mom and my family for their endless love and measureless support for getting me through the intense work and all the hard days.

This thesis work is supported in part by the Knoebel Institute for Longevity and Health Awards of the University of Denver (No. 89317-142152) , OKT Enterprises, LLC, and National Science Foundation Industry/University Cooperative Research Center for Safety, Security and Rescue Research.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation and background

Health care of elderly people is an eternal topic, and becomes a major issue in human society nowadays, since population aging has become a global trend. Falls, specially, are one of the most dangerous accidents that threaten health of elderly people. According to [1], no less than one-third of elders aged over 65 years fall each year, and nearly 13 million falls take place per year, among which approximately 50% are recurrent falls, and about 10% to 20% can result in serious injuries such as fractures or head traumas, or even lead to death. Although some falls do not cause injury, 47% of these non-injured falls brings about immovability, that the faller cannot get up immediately without external assistance.

Along with the damages of elderly falls, indicated by [1, 2, 3], the cost of health-care for elderly fallers is substantial and fall-related injuries account for 6% of all medical expenditures in the USA. In 2005, direct fall injury cost is 23.6 billion dollars, in 2010 30.4 billion dollars , and the number is still rising.

Thus, it is important for caregivers to monitor a patient's or elders' activities for the purpose of providing in time assist when they fall or taking necessary measures

to prevent a fall. For this reason, one of the most challenging and urgent requests in geriatric healthcare is to track moving motions of an elderly person, estimate potential motion, and further to reliably detect a dangerous movement and finally issue an early warning to the caregivers. To address the above issue,we propose a vision-based fall detection algorithm based on the *"skeleton view"* offered by the Kinect sensor.

And among all the conditions, under which a fall is possible to occur, bed-exiting catches our eyes. According to [4], most falls in nursing homes take place in the residents rooms, especially during attempts to get out of a bed. Reasons of bed-exiting related falls are mainly caused by an attack of vertigo, lack of strength, a sudden lost of balance or twisted feet with bed sheets. In [4], the authors discuss the effectiveness of bed-exiting alarm systems, in which they describe the accuracy of two types of bed-exiting alarms for detecting bed-exiting body movements: pressure-sensitive and a pressure sensitive combined with infrared beam detectors (dual sensor system). The authors draw a conclusion that while the dual sensor bed-exit alarm was more accurate than the pressure sensitive alarm in identifying bed-exiting, false alarms were not eliminated altogether. Hence, besides the general fall detection algorithm, this thesis developed another vision-based algorithm specifically for the bed-exiting detection.

1.2 Contribution

This thesis represents two vision-based human motion analysis approaches, and the specific fall detection and prevention applications of them. Contributions of this work are listed below:

1. We introduce the low-cost Kinect sensor as a data collection device in a surveillance system, and make use of the data from it to build different 3D models for motion estimation.

2. The skeletal view based approach is with largely reduced computational load of data processing, and by selecting the same origin and x, y, z axis the skeletal review based approach is invariant to different view angles.

3. Taking advantage of the innovative classification of motions for fall detection application, when real-time fall prediction is realized, direction of falls can also be predicted for facilitating possible protective methods that rely on the prediction.

4. The segmented motion history image sequences based approach offers a fast and uncomplicated way to extract spatiotemporal motion information not only in lateral directions but also in the direction along the depth change. The Hu moments for feature translation is scale and rotation invariant, which can be applied to any videos recorded by a variety of video based applications.

1.3 Thesis outline

Chapter 2 reviews related works about human motion estimation, fall detection and bed-exiting detection. Chapter 3 narrates the entire experiment design, methodology, and experimental results of the 3D skeletal view based fall detection project. Chapter 4 describes how the bed-exit detection is achieved, which includes data collection experiment design, details of the algorithm, experimental results of activity classification and evaluation on the capability of bed-exiting prediction. At last, Chapter 5 concludes the work presented in the thesis, and discusses potential future work.

Chapter 2

Literature review

2.1 Categories of fall detection approaches

Tremendous efforts have been made to build a credible in-house fall detection system for elderly people. According to how a fall is detected, [5] summarizes and classifies the major fall detection technologies into three categories: wearable device, ambience device, and vision based approaches.

2.1.1 Wearable devices based approaches

The wearable device approach embeds sensors, mostly accelerometers, into garments or devices held or worn by a person to detect motion and/or posture of the body to identify potential fall events. In [6] Clifford *et al* patented a human body fall detection system, which consists of a set of accelerometers, a processor, and a wireless transmitter. The accelerometers collect and transmit acceleration measurements to the processor. The processor compares the acceleration measurements with a value range to decide if the wearer is experiencing a fall activity. The transmitter transmits to a remote receiver the signal generated by the processor when a fall is detected. In [7] Noury *et al* developed a fall detection device, which collects

three types of measurements: vertical acceleration shock obtained by a piezoelectric accelerator, body orientation monitored by a position tilt switch, and mechanical vibrations of body surface. These measurements are transmitted to and analyzed by a PC to identify fall activities. T. Tamura *et al* in [8] developed a wearable airbag which is inflated by acceleration and angular velocity signals in a fall-detection system. Their fall-detection algorithm could detect signals 300 ms before the fall and trigger the inflation of the wearable airbag. H. Ghasemzadeh *et al* in [9] presents a system using inexpensive, off-the-shelf inertial sensor nodes that constructs motion transcripts from biomedical signals and identifies movements by taking collaboration between the nodes into consideration. They use motion primitives to built the transcripts, and each primitive is labelled with a unique symbol. Representation of a particular action is fulfilled by a sequence of such symbols, known as motion template. And their action recognition is achieved using edit distance with respect to motion templates.

Most wearable devices are inexpensive and easy to set up and operate, however they are not very appealing to the wearers, and this approach is reported to have a high rate of false alarm.

2.1.2 Ambience devices based approaches

Ambience device based approaches are to install sensors in a fixed space to collect information of a person who is in their effective range. Pressure, vibration, acoustic sensors and stabilometers are commonly used in this approach. [10] introduces a vibration sensor based fall detection apparatus developed by Alwan *et al.* The vibration sensor are embedded on the floor to illustrate the location at any moment. The processor recognizes falls through analyzing the data from these location. [11] introduces a patented bed exit detection device, which utilizes bladders or other fluid-carrying devices cooperating with a pressure sensor so that the pressure sensor

registers a bladder pressure in response to the person's weight. An acoustic fall detection system (acoustic-FADE) is introduced in [12] by Y. Li *et al.* The acoustic system acoustic-FADE consists of a circular microphone array that captures sounds in a room. When a sound is detected, acoustic-FADE locates the sound using steered response power with phase transform technique, enhance the sound signal by the beamforming technique and then using mel-frequency cepstral coefficient features extracted from the enhanced signal to do fall and nonfall classification. H. Rimminen in [13] presents a fall detection method using a floor sensor based on electric near-field imaging. A. Sixsmith in [14] develops an intelligent fall detector based on a low-cost array of infrared detectors.

An advantage of this approach is that the devices are inexpensive, and unlike the wearable devices, they are not intrusive to the person being monitored. The disadvantage is the range limitation of usage and the inaccuracy resulting from susceptibility of the signals to environmental factors.

2.1.3 Vision based approaches

Vision based approaches generally use videos or images to analyze motion features of a human body, and distinguish features of fall activities from those of non-falls to realize the function of fall detection. This approach can be divided into 3 categories based on utilized principles relating to the characteristics of fall movements. The first category is *inactivity analysis*. An inactivity period on the ground is considered as a consequent result of a fall activity. Jansen and Deklerck in [15] identified the body area and the body's orientation according to 3D image recorded by a stereo camera. In certain context, a fall activity is detected once inactivity is identified by analyzing the orientation change of a human body. Second category is *shape change analysis*. Toreyin *et al* in [16] presents an hidden Markov model (HMM) based algorithm. The height to width ratio of the bounding box of a

body configuration under falling conditions and walking conditions are employed as movement features for HMMs. In [17], Southwest Research Institute developed an fall-prevention monitoring solution based on image sensors, with bed-exiting image processing and pattern recognition, which is designed to alert caregivers to impending bed exits. [18] developed an awakening behavior detection system on patient behavior, such as almost falling down from the bed and waking up on the bed. This system uses a web camera as an imagine sensor and the captured images are analyzed using the neural network (NN) algorithm for recognizing patient behavior into sleeping or waking up. The effectiveness of the image and NN based system are also discussed with experimental results in a laboratory environment. In [19], the authors analyzed the histogram of bed-exiting images, and proved that the fluctuation of brightness quantity decreases the detection capability. The bed-exiting detection performance was improved by reducing fluctuation of brightness using histogram equalization. The last category is *3D head motion analysis*. This category considers that velocity of vertical movement is larger than horizontal movement in a fall activity. Rougier in [20] uses monocular 3D head tracking to distinguish fall activities and walking activities.

2.2 Motion history images

Motion history images (MHIs) first introduced by Bobick and Davis [21] are widely used for analyzing dynamics of a series of moving human silhouettes. In an MHI, each pixel value illustrates the recency of motion at that location, for which brighter pixel corresponds to more recent motion. In a series of continuing works of the origin developers Bobick and Davis [21, 22, 23, 24, 25], the developers combined the motion energy images (a motion energy image is a binary image, which roughly locates the area of motions of a given action from a given view) and motion history images for representation and recognition of actions, further they improve the MHIs

into timed motion history images (tMHIs) and compute the motion history gradient to achieve motion segmentation and pose recognition. Figure 2.1 from [21] gives an example of motion energy and motion history images. And they extend their previous work to provide a method for calculating local motion orientation and implement their algorithm into the Intel computer vision library (CVLib) for real-time application. In another research work for fall detection by Rougier *et al* [26], the authors combine MHIs with human shape changes and a condition, that lack of motion generally appears after a fall event takes place to achieve the goal of fall detection.

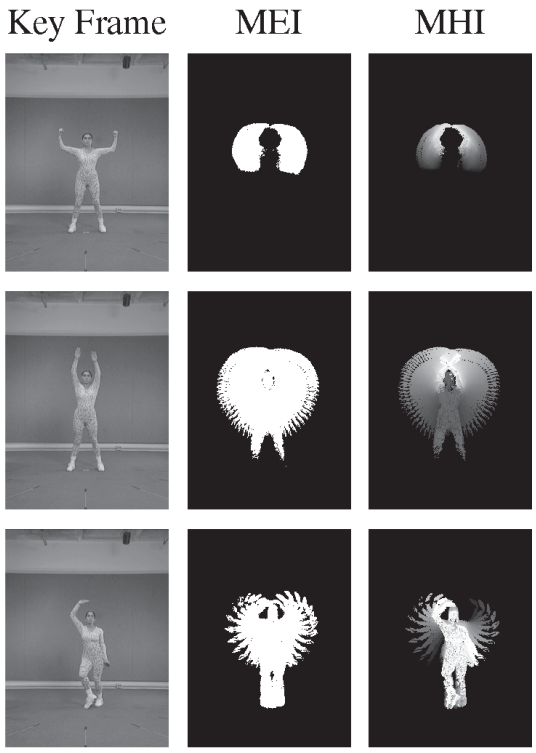

Figure 2.1: Example of motion energy images and motion history images.

Chapter 3

Skeletal view based fall detection

In this chapter, we exploit the *Kinect* sensor with its *Windows SDK* to detect fall activities. The recorded spatial coordinates of the human body joints fromt Kinect's 3D skeletal view are then processed to extract posture features. Then the principle component analysis and k-means clustering algorithms are applied for dimensionality reduction and feature translation. HMMs are well known for their application in temporal pattern recognition such as speech, handwriting, gesture recognition, thus they are chosen for this project to classify human motion which is a temporal sequence of postures. HMMs are trained by the labelled extracted features to model and discriminate four fall motion classes and three non-fall classes. The algorithm and data flow path of proposed elderly fall detection system is displayed in Fig. 3.1.

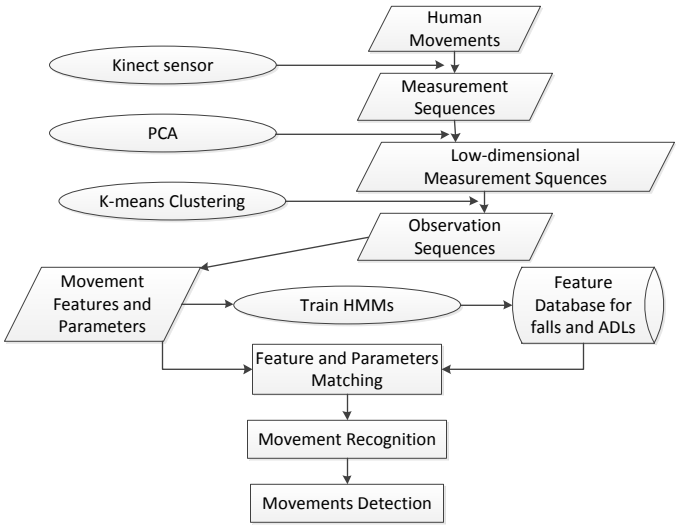

Figure 3.1: The algorithm and data flow of proposed fall detection system.

3.1 Motion classification and experiment design

The aim of this work is to detect fall activities of elderly people, therefore recorded videos or pictures of real elderly falls should be the best materials to train HMMs and to form the proposed motion analysis database. However, such real falls were rarely recorded. As a result, we collect data using laboratory simulated falls. Although involvement of elderly people to perform simulated falls would best serve the goal of this research work, anticipated risk for elderly people to fall is too high for IRB to approve. For this reason, only human subjects in the age range of 18-40 are intended to be recruited in the data collection experiments. By learning the descriptions of recorded real elderly falls in [27] , nine most commonly seen recurrent falls are summarized and are divided into 4 classes according to different starting positions and different movement directions. And 6 activities of daily living (ADLs) are also classified into 3 classes based on starting positions. Scenarios of 9 laboratory simulated falls, 6 simulated ADLs and their classifications are listed in

1. **Start from a standing position, fall occurs in horizontal direction**	
(1)	Standing then fall to the right
(2)	Standing then fall to the left
2. **Start from a standing position, fall occurs in vertical direction**	
(1)	Standing then fall forward
(2)	Standing then slip to fall backward
(3)	Standing then sit on empty
3. **Start from a sitting position, fall occurs in horizontal direction**	
(1)	Sitting in an armchair then try to stand up but suddenly lose strength and fall to the right
(2)	Sitting in an armchair then try to stand up but suddenly lose strength and fall to the left
4. **Start from a sitting position, fall occurs in vertical direction**	
(1)	Sitting on a stool shuffling in seat then fall off the stool
(2)	Sitting in an armchair then try to stand up then fall forward

Table 3.1: Scenarios of nine simulated falls.

Table 3.1 and Table 3.2, respectively.

5. **Start from a standing position, movements take place in the horizontal direction**	
(1)	Stand then walk forward and backward
(2)	Stand then bend to pick up something on the floor
6. **Start from a sitting position**, **movements take place in the vertical direction**	
(1)	Sit on a chair with various poses
(2)	Sit on floor with various poses
7. **Starts or ends at a lying position**	
(1)	Lying on bed then get up and sit on the edge of the bed then stand up and walk
(2)	Stand by a bed then sit down on the edge of the bed then lie down

Table 3.2: Scenarios of six ADLs.

Note that unlike previous studies, which simply divide falls and non-falls, or operates with only fall movements and walking movements, subdivision within fall activities and ADLs are investigated in this paper. We design our experiments in this way due to the following considerations. 1) the property of the data collection device, which utilizes depth measurements to describe a body's posture, such that vertical movements can contribute a huge difference comparing with horizontal movements. 2) It is the best for a certain HMM to be trained by a group of similar sequences in this application; therefore, criterion of movement direction is applied. This also explains the criterion of starting position. 3) Another consideration of not combining all falls and all ADLs is that, unlike application of HMM in speech recognition, which at first detects presence or absence of speech, fall activities and ADLs in fact have a lot in common. The vital decision moment might pretty short, and the interfusion of all falls and all non-falls might lose these vital features to contrast them.

In this data collection experiment, currently, only one volunteer was recruited to perform the designed activities. However volunteers of different ages and genders will be recruited to ensure the diversity of training data. When fall activities were performed, the floor was padded by an 8-inches professional gymnastic mat. Each of the 15 activities was performed 10 times by the volunteer, and approximately 5 seconds per activity. However, this final scenario was replenished and optimized after the first experiment, in which fall activity 4.(1) and ADL 1.(2) were not included. Thus only 13 activities are performed, and in total 130 clips were recorded, generating 130 measurement sequences.

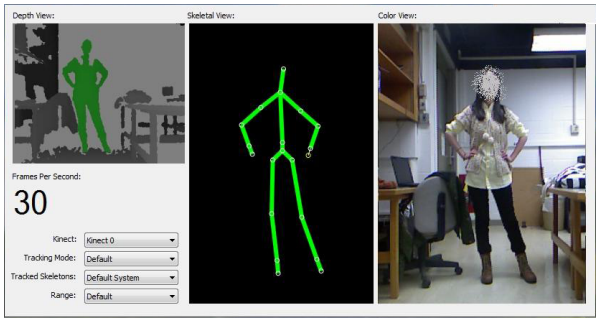

Figure 3.2: Sample picture of *Skeletal Viewer*.

3.2 Methodology

3.2.1 *Skeletal Viewer*

The *Kinect* sensor offers a simple and convenient way to capture and record features of human body motion. *Skeletal viewer* in *Kinect for Windows SDK* allows us to extract a human body skeleton in each frame, which consists of 20 joints. A sample picture is shown below in Fig. 3.2 and Fig. 3.3 interprets name of every joint.

Skeleton parameters are utilized for human posture sequence modeling and fall detection. *Kinect for Windows SDK* outputs 3 coordinates for each joint, 20 joints for each frame, and about 30 frames per second. a position vector of one joint is denoted as

$$\mathbf{p}_{k,l} = [x_{k,l}\, y_{k,l}\, z_{k,l}]^T,$$

where k is the frame index, l is the joint index, $l = 1, 2, \cdots, 20$, and T denotes matrix transpose.

Each frame is used as an elementary processing unit, which is denoted as a column-vector employing the 60 variables for one frame. The 60-dimensional experimental data is denoted as

$$\mathbf{p}_k = [\mathbf{p}_{k,1}^T \mathbf{p}_{k,2}^T \cdots \mathbf{p}_{k,l}^T]^T.$$

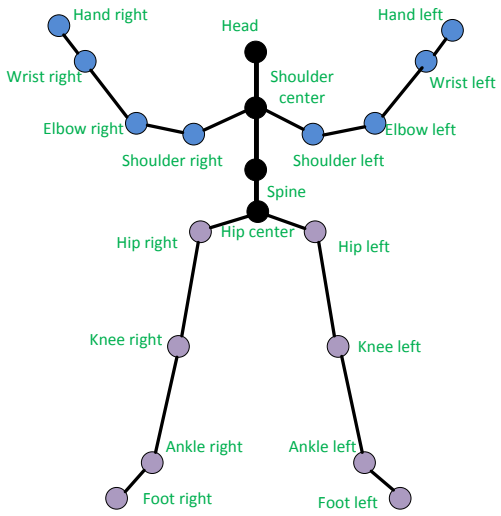

Figure 3.3: Interpretation of 20 joints of a skeleton.

And one clip of a human activity consisting of K frames is defined as a measurement matrix

$$P = [\mathbf{p}_1 \mathbf{p}_2 \cdots \mathbf{p}_K].$$

Since the output of *Kinect* sensor consists of 3-D coordinates for human body joints, by selecting the same origin and x, y, z axes the skeleton review based data collection method is invariant to different view angles.

3.2.2 Data collection

Data collection experiments are conducted to record in total 130 clips of activities from 7 motion classes. Definition of motion classification and details of experiment are narrated later in this paper in Section 3.1. The sets of raw measurement matrices are denoted as

$$\mathcal{P}^i = \{P^{i,1}, P^{i,2}, \cdots, P^{i,j}, \cdots, P^{i,J(i)}\},$$

where $i = 1, 2, \cdots, 7$, is the index of motion class, j is the index of the video clip, and $J(i)$ is the number of recorded activities in the ith motion class, $J(i) = 10$ if $i = 4$, $J(i) = 20$ otherwise. Difference in value of J is not intentionally set by the author, but the true number of activities recorded in the data collection experiment.

3.2.3 Data processing

Up to this point, we have obtained 130 raw measurement matrices $P^{i,j}$. We first apply principle component analysis to reduce data dimension. Then, we employ the k-means clustering to prepare training data for HMMs. We train 7 HMMs corresponding to 7 motion classes. Finally, we evaluate the ability of these HMMs to discriminate activities among the seven motion classes, and also their capability to distinguish fall and non-fall activities.

Principle component analysis

Principal component analysis (PCA) runs an orthogonal transformation to convert $P^{i,j}$ to a new coordinate system that consists of linearly uncorrelated variables such that the first variable has the largest possible variance of $P^{i,j}$, and each succeeding variable in turn has the highest variance. The new variables are called principle components. By choosing the first n ($n \leq 60$) principle components, we can reduce data dimension and preserve most of the information in $P^{i,j}$. PCA is accomplished by algorithms narrated below. Singular value decomposition of original measurement matrix ${P^{i,j}}_{(60\times K)}$ is

$${P^{i,j}}_{(60\times K)} = {W^{i,j}}_{(60\times 60)}{\Sigma^{i,j}}_{(60\times K)}{V^{i,j}}_{(K\times K)},$$

where

$W^{i,j}$ is 60×60 matrix of eigenvectors of the covariance matrix $P^{i,j}P^{i,j^T}$,

$\Sigma^{i,j}$ is $60 \times K$ rectangular diagonal matrix with non-negative real numbers on the diagonal,

$V^{i,j}$ is $K \times K$ matrix of eigenvectors of $P^{i,j^T}P^{i,j}$.

The reduced-dimensionality presentation ${R^{i,j}}_{(n\times K)}$ of ${P^{i,j}}_{(60\times K)}$ is

$$ {R^{i,j}}_{(n\times K)} = {I^{i,j}}_{(n\times 60)}{\Sigma^{i,j}}_{(60\times K)}{V^{i,j}}_{(K\times K)}, $$

where

n is reduced dimensionality after PCA,

$I^{i,j}$ is $n \times 60$ rectangular identity matrix.

Let w_m denote the weight of eigenvalue of the mth principle component, which is defined as ratio of the mth eigenvalue to the sum of all the eigenvalues, and $m = 1, 2, \cdots, 60$. Fig. 3.4 represents the relationship of w_m versus m. Because the first few principle components account for a majority of the variance of measurement matrix $P^{i,j}$, weights of from the $16th$ to the last principle components are too small and are neglected, therefore they are not displayed in Fig. 3.4.

Let r_n denote the percentage of the variance the first nth principle components account, i.e. how much information of original measurement matrix can be preserved by leaving n principle components is demonstrated in Table 3.3.

n	3	5	10	15	30	60
r_n	90.67%	95.08%	98.46%	99.35%	99.92%	100.00%

Table 3.3: Relationship of r_n and n.

In this work, to trade off between lowering dimensionality and preserved information, we choose $n = 10$. And the measurement matrices after PCA become feature matrices

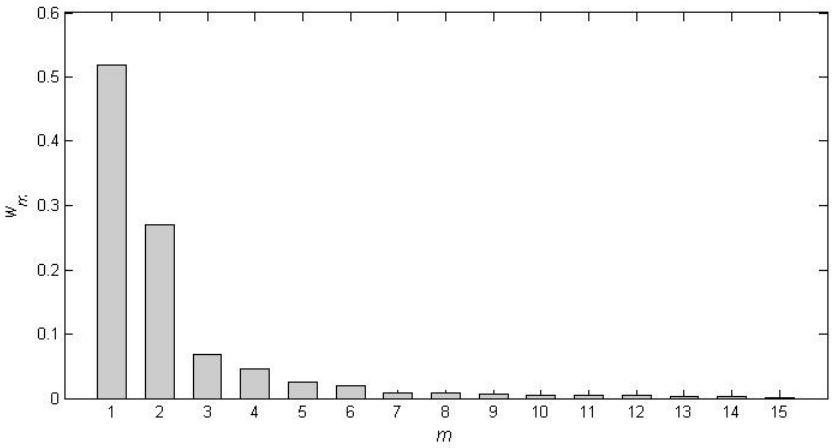

Figure 3.4: Relationship of w_m and m.

$$R_{(10\times K)} = [\mathbf{r}_1\mathbf{r}_2\cdots\mathbf{r}_K],$$

where $\mathbf{r}_k$ is the measurement vector at time k after PCA dimensionality reduction. And the sets of feature matrices are generated as

$$\mathcal{R}^i = \{R^{i,1}, R^{i,2}, \cdots, R^{i,j}, \cdots, R^{i,J(i))}\}.$$

K-means clustering

All the measurement vectors after PCA dimensionality reduction ${\mathbf{r}_k}^{i,j}$ from all seven motion classes are partitioned into c clusters $\mathcal{S} = \{\mathcal{S}_1, \mathcal{S}_2, \cdots, \mathcal{S}_c\}$ by the k-means clustering algorithm. The k-means clustering distributes vectors into c clusters aiming at achieving minimum within-cluster sum of squares

$$\arg\min_{\mathcal{S}} \sum_{q=1}^{c} \sum_{{\mathbf{r}_k}^{i,j}\in\mathcal{S}_q} \parallel {\mathbf{r}_k}^{i,j} - \boldsymbol{\mu}_q \parallel^2,$$

where

$\boldsymbol{\mu}_q$ is the mean of points in set $\mathcal{S}_q$,

$\| \mathbf{r}_k{}^{i,j} - \boldsymbol{\mu}_q \|^2$ is square of distance from $\mathbf{r}_k{}^{i,j}$ to the mean $\boldsymbol{\mu}_q$.

The data collected at every frame is indexed by the number of cluster $1, 2, \cdots, c$, to which it belongs. And the sets of feature matrices are converted to sets of observation sequences as following

$$\mathcal{R}^i = \{R^{i,1}, R^{i,2}, \cdots, R^{i,j}, \cdots, R^{i,J(i)}\},$$
$$\Downarrow$$
$$\mathcal{O}^i = \{\beta^{i,1}, \beta^{i,2}, \cdots, \beta^{i,j}, \cdots, \beta^{i,J(i)}\}.$$

where $\beta^{i,j}$ is the jth observation sequence in the ith motion class.

In this paper, the number of clusters c is selected to be $c = 70$, aiming at discriminating particulars of a movement.

Training of hidden Markov models (HMMs)

If an event can be separated into a set of observation sequences which are associated with a number of "invisible" states, the progress of this event can be treated as transitions among states. Similar events generally have similar states transitions. HMMs can model the state transitions of similar events, and describe them using a set of parameters:

$$\Lambda = \Big\{\pi, A, B\Big\},$$

where

π is the $s \times 1$ initial state distribution vector,

A is the $s \times s$ hidden state-transition matrix,

B is the $s \times c$ state-dependent observation probability density matrix,

s is the number of hidden states,

c is the number of observation symbols.

Given a set of training observation sequences $\mathcal{O}$, the training process of an HMM Λ is to optimize the model parameters so that they can best describe how the given sequence occurs, which is

$$\Lambda = \arg\max_{\pi,A,B} P(\mathcal{O}|\Lambda).$$

An iteration algorithm with expectation-maximum (EM) named Baum-Welch algorithm is applied to solve this problem.

In this paper, seven HMMs

$$\{\Lambda_1, \Lambda_2, \cdots, \Lambda_7\}$$

are created to describe their corresponding motion classes.

Classification using hidden Markov models

Evaluation of the capability of the 7 created HMMs to discriminate different activities from 7 motion classes is conducted by choosing testing sequences, and calculating the probability of such testing sequences given $\Lambda_1, \Lambda_2, \cdots, \Lambda_7$. A testing sequence ξ is classified into the $\hat{i}th$ motion class, only when the probability given $\Lambda_{\hat{i}}$ is the largest among those given the other 6 models, which is

$$\hat{i} = \arg\max_{i} P(\xi|\Lambda_i).$$

In our experiment, 10 training observation sequences from each motion classes are randomly chosen as testing data

$$\mathcal{T}^i = \{\xi^{i,1}, \xi^{i,2}, \cdots, \xi^{i,10}\},$$

where

$\xi^{i,j}$ is the jth testing sequence chosen from the ith motion class,

$\mathcal{T}^i$ is the set of testing sequences chosen from the ith motion class.

HMM based classification results are displayed in Section 3.3 using the confusion matrix of activity recognition of seven hidden Markov models.

3.3 Experiment results

All 130 measurement sequences are divided into 7 motion classes and used as training data for HMMs, and 10 measurement sequences from each motion classes are randomly chosen to act as testing data. The entire data processing procedure is conducted in MATLAB, and takes approximately 6 seconds, and it takes approximately 0.065 seconds to classify one single activity.

Table. 3.4 demonstrates the results of classification using trained HMMs. Rows $C_1, \cdots, C_7$ represent the motion classes from which testing sequences are chosen, and columns $\Lambda_1, \cdots, \Lambda_7$ represent the HMMs of corresponding motion classes. Intersection of row a and column b displays percentage of the event, that activity from motion class a is classified into motion class b. Classification of a single activity ξ is achieved in the following way, the likelihood probability of a certain activity given all seven trained HMM models are calculated, and the motion class corresponding to the model achieving the highest likelihood is recognized to be the motion class, to which this activity belongs. And this can be summarized as

$$\hat{i} = \arg\max_i P(\xi|\Lambda_i).$$

Ideally, the classification rate should be 1.0 on the diagonal where $a = b$, which means all the testing activities from C_i are correctly classified into C_i. Classification rate of a certain motion class is defined as the ratio of number of the correctly classified activities to the total number of activities for test from this motion class,

	Λ_1	Λ_2	Λ_3	Λ_4	Λ_5	Λ_6	Λ_7
C_1	10	0	0	0	0	0	0
C_2	0	8	0	2	0	0	0
C_3	0	0	8	1	0	1	0
C_4	0	0	1	8	0	1	0
C_5	0	0	0	2	8	0	0
C_6	0	1	0	1	1	7	0
C_7	1	0	0	0	1	1	7

Table 3.4: Confusion matrix of activity recognition of seven HMMs.

which can be read from the diagonal of confusion matrix.

Precision and recall and the F1 score of each HMM model Λ_i are presented in Table 3.5. The average F1 score reaches 0.80, and the lowest is 0.67.

	Λ_1	Λ_2	Λ_3	Λ_4	Λ_5	Λ_6	Λ_7
Precision	0.91	0.89	0.89	0.57	0.80	0.70	1.00
Recall	1.00	0.80	0.80	0.80	0.80	0.70	0.70
F1 score	0.95	0.84	0.84	0.67	0.80	0.70	0.82

Table 3.5: Precision, recall and F1 score of each HMM model.

3.4 Discussion

In this work, testing activities from all seven motion classes are generally correctly classified, the lowest recognition rate is 0.7, and average recognition rate is 0.8. Observing the results, major misclassification takes place among similar motion classes, which shares either the same starting position (C_3 and C_4, or C_4 and C_6) or the same movement direction (C_2 and C_4, or C_4 and C_6).

The division of falls corresponding to $\Lambda_1, \cdots, \Lambda_4$ and ADLs corresponding to $\Lambda_5, \cdots, \Lambda_7$ indicates that regardless of subdivision within falls or ADLs, 2 out of 40

falls are misunderstood as ADLs, and 4 out of 30 ADLs are misidentified as falls. Therefore, recognition rate of fall activities is 95%, and 87% for ADLs. It can be observed that falls are generally correctly recognized and distinguished from ADLs, but ADLs are lightly confused with falls

These results are promising under the condition that neither the size of our training data nor the size of testing data is big enough. Generally, it requires hundreds or thousands training sequences to train a "reliable" HMM, and size of training data should be at least five times larger than the size of testing data.

And regarding the selection of number of preserved principal component in Section 3.2.3, since the 60 original variables in $\mathbf{p}_k$ in our experiment data set are the spatial coordinates of 20 human body joints of one person, they are highly correlated. And this explains why only 15 out of 60 principal components can account 99% of the variance. Thus, we can foresee, that if massive data from different subjects are utilized, more principal components will need to be preserved compared to the current situation. And a threshold value on r_n need to be set for selecting proper value of n.

Chapter 4

Bed-exiting detection based on segmented motion history image sequences

In this chapter, we exploit segmented motion history image (MHI) sequences to extract spatiotemporal features of a moving human body. Eight Hu image moments are calculated to translate the spatiotemporal features of each frame into vectors to describe video frames. The k-means clustering and HMM modelling are utilized for vector quantization and classification between bed-exiting activities and rolling-on-bed activities. In addition, except activity classification, likelihood probability curves are generated along the time line of all MHIs, endeavoring to predict a bed-exiting activity.

The outline algorithm and data flow path of the research work described above is demonstrated in Fig. 4.1.

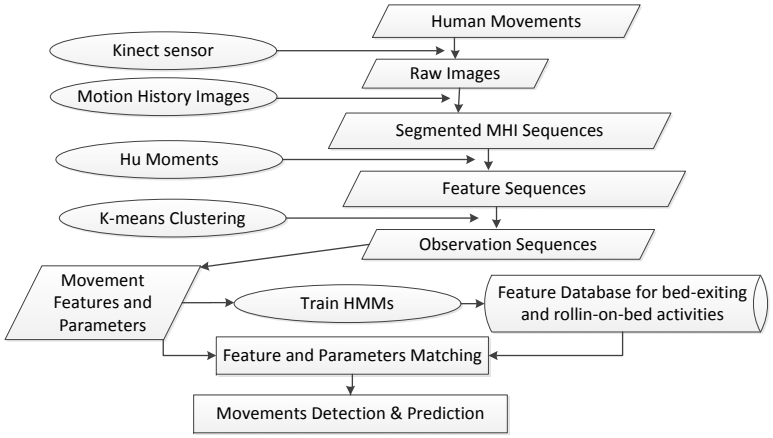

Figure 4.1: The algorithm and data flow of the proposed bed-exiting detection system.

4.1 Data collection

4.1.1 Preliminary data collection section

Preliminary data collection experiment is directed for algorithm testing, and only one ordinary digital camera is used to record all activities, thus only RGB videos are collected. Fifteen human subjects over 50 years old were invited to participate in a series of data collection experiments. These subjects are instructed by the experimenter to perform various movements in a randomized order. Movements performed by all the volunteers are listed in Table 4.1, and some of these movements are performed with a bed sheet on, and the other rest are with a bed sheet off. 172 video clips/activities in total were recorded, among which 132 are bed-exiting activities and 40 are rolling-on-bed activities. And 60 of 132 bed-exiting and 25 of 40 rolling-on-bed were performed with a bed sheet. The statistic is given in Table 4.2.

	Movement details
1.	Get up then exit a bed from the right/left side
2.	Lying with face down and exit a bed from the right/left side
3.	Lying on the right/left side and roll to the left/right side
4.	Lying with face up and roll to the left/right side
5.	Lying on the left/right side and roll to face up

Table 4.1: Scenario for the preliminary bed-exiting data collection section.

	Bed-exiting	Rolling-on-bed
Covered	60	25
Uncovered	72	15

Table 4.2: Statistic of video clips of preliminary data collection.

4.1.2 Experimental data collection section

A second data collection section was conducted in the human dynamic laboratory in Ritchie center at University of Denver. The experiment area was set up referring to the dimensions of a real chamber in a nursing home. Forty human subjects from both genders over 50 years old have participated in this section. Following the instructions of the experiment director, all subjects performed 32 movements with half of the movements being done with the covers on and the other half with the covers off, and after every 8 movements, bed orientation was changed. The order of movements were randomized to negate any effect of ordering. Figure 4.2 shows the basic 4 conditions of the data collection section.

The 8 basic movements begin from one of three starting positions, either lying face up or lying face left/right. And Table 4.3 offers the scenario of movements included in this section.

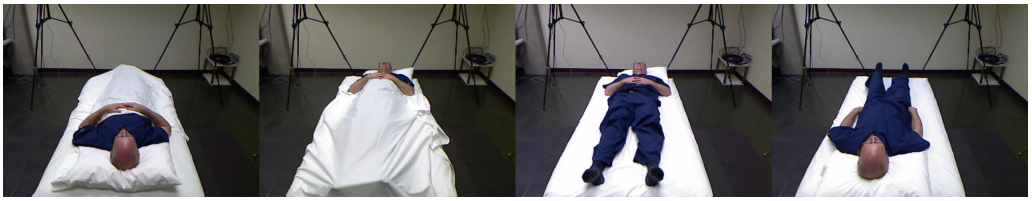

Figure 4.2: Four basic data collection conditions.

	Movement details
1.	Face up - roll to left
2.	Face up - roll to right
3.	Face up - exit right
4.	Face up - exit left
5.	Face right - roll to face up
6.	Face left - roll to face up
7.	Face right - exit right
8.	Face left - exit left

Table 4.3: Scenarios of 8 basic movements for bed-exiting detection data collection.

4.2 Methodology

Raw videos of human actions are recorded by a Kinect sensor using *NuiCapture* software from Cadavid Concepts, Inc. RGB videos and Depth videos are recorded simultaneously. Let $I_k(x, y)$ and $D_k(x, y)$ denote a raw RGB image and the corresponding depth image at frame k, respectively, k denotes the frame index. Both $I_k(x, y)$ and $D_k(x, y)$ are 640×480 matrices. The pixel value of $I_k(x, y)$ is the image intensity, and the pixel value of $D_k(x, y)$ is the distance from the camera. Thus one RGB video clip of a human activity consisting of K frames is denoted as

$$\mathcal{I} = \{I_1(x, y), I_2(x, y), \cdots, I_k(x, y), \cdots, I_K(x, y)\},$$

and the corresponding depth video clip is denoted as

$$\mathcal{D} = \{D_1(x,y), D_2(x,y), \cdots, D_k(x,y), \cdots, D_K(x,y)\}.$$

4.2.1 Segmented motion history image sequences

A general MHI for an RGB image sequence color-MHI (cMHI) at the kth frame $M_k^c(x,y)$ is produced according to:

$$M_k^c(x,y) = \begin{cases} \tau & \text{if } |I_k(x,y) - I_{k-1}(x,y)| > \delta I_{th} \\ max(0, M_{k-1}^c(x,y) - 1) & \text{otherwise} \end{cases}$$

where τ is the longest time window to be considered, δI_{th} is the threshold value for selecting wanted motions. However, unlike previous research works, which generate only one MHI for an entire action/video, we divide one action video into several pieces with the same length α, and generate one MHI every α frames. In this way, segmented cMHIs $M_k^c(x,y)$ in our approach are produced as

$$M_k^c(x,y) = \begin{cases} \pi & \text{if } |I_k(x,y) - I_{k-1}(x,y)| > \delta I_{th} \\ max(0, M_{\tilde{k}-1}^c(x,y) - p_c) & \text{otherwise} \end{cases}$$

$$\tilde{k} = \begin{cases} 1 & \text{if } k = \alpha n + 1, n = 0, 1, 2, \cdots, \lfloor \frac{K}{\alpha} \rfloor \\ k & \text{otherwise} \end{cases}$$

where

π is the augmentation coefficient for pixels of the most recent motions,

δI_{th} is the threshold value for selecting wanted motions,

p_c is the fading coefficient for pixels of previous selected motion,

K is the total number of frames in a video clip,

α is the number of frames in a cMHI generation cycle.

Thus the most useful cMHI is the last one in every cycle, which contains most motion information in that cycle. And a segmented cMHI sequence representing an entire video clip can be built as

$$\mathcal{M}^c = \{\bar{M}^c_1, \bar{M}^c_2, \cdots, \bar{M}^c_{\hat{k}}, \cdots, \bar{M}^c_{\lceil \frac{K}{\alpha} \rceil}\},$$

$$\bar{M}^c_{\hat{k}} = \begin{cases} M^c_{\hat{k}\alpha} & if\ \hat{k} < \lceil \frac{K}{\alpha} \rceil \\ M^c_K & if\ \hat{k} = \lceil \frac{K}{\alpha} \rceil \end{cases}$$

where $\hat{k}$ is the frame index of effective MHI frames.

Parameter δI_{th} is set to filter unwanted changes between two successive images. Larger δI_{th} can eliminate more noise in an MHI, which might be caused by changes in illumination or slight camera trembling. But a defect of a large δI_{th} is that it might also eliminate subtle human posture changes of interest.

p_c is the parameter that distinguishes the recency of motions in a cycle on an MHI. The value of p_c does not affect the classification capability of the proposed algorithm, but it is restricted by the value of augmentation coefficient π and cycle α, in the way

$$p_c \cdot \alpha \leq \pi,$$

or motions during the early time in a cycle cannot be recognized.

α is a time-related parameter. Since the proposed bed-exiting algorithm processes an activity classification every time when an effective MHI is generated, which is explained later in Section 4.4, the cycle of MHI generation, α, decides the density of video segment classification, and therefore decides the time precision of a bed-exiting early alarm. Smaller α can increase the time precision of the early alarm. However, the smaller the α is, the less motion information is included in an effective

MHI, therefore it will be more difficult for the training of HMMs. Considering the original application of MHIs, which is static posture classification, larger α generally can ensure higher recognition rate. In addition, since α is time-related, moving speed and camera frame rate can affect the selection of α. Different classification results for two different choices of α will be displayed in Section 4.3.

Related previous works using MHIs focus only on the 2D RGB images, that only cMHIs are produced. But in this way, only motions in the lateral directions are recognized, and changes of range are totally lost. To represent a motion more precisely, depth information should be taken into account. Inspired by Ni *et al* [28], MHIs derived from depth images dMHIs are exploited, and they are divided into two types: forward-dMHI (fdMHI), which indicates movement departing the camera and backward-dMHI (bdMHI), which indicates movement approaching the camera. Segmented fdMHIs in our approach are generated according to:

$$M_k^{fd}(x,y) = \begin{cases} \pi & \text{if } D_k(x,y) - D_{k-1}(x,y) > \delta D_{th} \\ max(0, M_{\tilde{k}-1}^{fd}(x,y) - p_d) & \text{otherwise} \end{cases}$$

$$\tilde{k} = \begin{cases} 1 & \text{if } k = \alpha n + 1, n = 0, 1, 2, \cdots, \lfloor \frac{K}{\alpha} \rfloor \\ k & \text{otherwise} \end{cases}$$

Here δD_{th} denotes the threshold value for depth images for selecting wanted motions. Segmented bdMHIs are induced in a similar way to that of segmented fdMHIs, with the thresholding condition substituted by $D_k(x,y) - D_{k-1}(x,y) < -\delta D_{th}$. Thus representation of a video clip by using segmented fdMHI and bdMHI sequences are:

$$\mathcal{M}^{fd} = \{\bar{M}_1^{fd}, \bar{M}_2^{fd}, \cdots, \bar{M}_{\hat{k}}^{fd}, \cdots, \bar{M}_{\lceil \frac{K}{\alpha} \rceil}^{fd}\},$$

and

$$\mathcal{M}^{bd} = \{\bar{M}_1^{bd}, \bar{M}_2^{bd}, \cdots, \bar{M}_{\hat{k}}^{bd}, \cdots, \bar{M}_{\lceil \frac{K}{\alpha} \rceil}^{bd}\},$$

where

$$\bar{M}_{\hat{k}}^{d} = \begin{cases} M_{\hat{k}\alpha}^{d} & \text{if } \hat{k} < \lceil \frac{K}{\alpha} \rceil \\ M_{K}^{d} & \text{if } \hat{k} = \lceil \frac{K}{\alpha} \rceil \end{cases}$$

Figure 4.3 shows an example of action representation by a series of cMHIs with $\alpha = 10,\ K = 65$. This example video is about a person exiting the bed. The first line of Figure 4.3 exhibits some key raw frames in this RGB video, the second line displays the segmented cMHI sequence. Figure 4.4 displays the corresponding key depth images on the first line, fdMHIs and bdMHIs on the second and last line for this sample video.

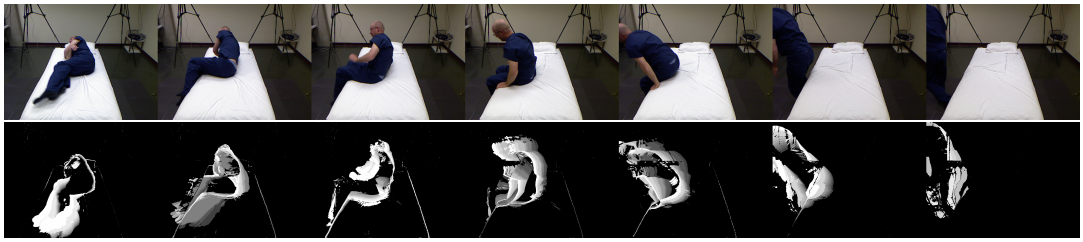

Figure 4.3: An example of representation of an action by segmented MHI sequences.

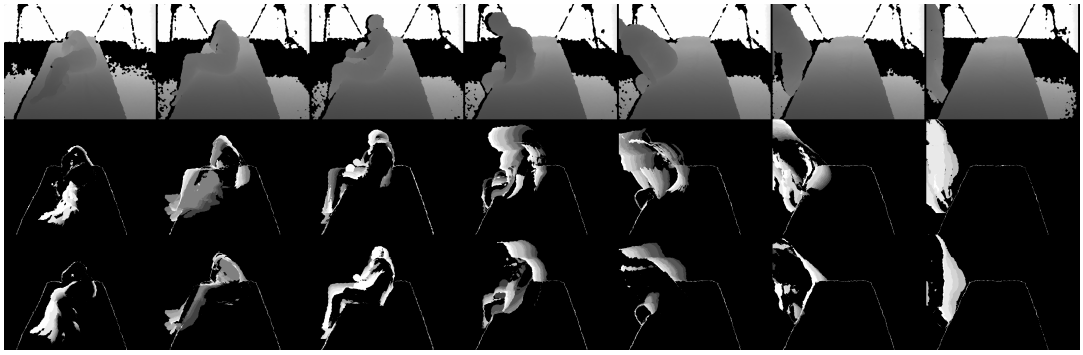

Figure 4.4: An example of representation of an action by segmented dMHI sequences.

4.2.2 Hu moments

In order to extract features of segmented MHIs, eight statistic descriptors, the Hu moments [29] $\mathbf{h}_{\hat{k}}$, which are invariant to scale, translation and rotation, are calculated for every MHI frame $M_{\hat{k}}(x, y)$, where $\hat{k}$ is frame index of MHIs. Except that the zeroth order Hu image moment refers to the area of an image, Hu moments of other orders have no specific physical meaning, for they are merely mathematically formulated to be invariant under translation, scale and rotation. Calculation of a general Hu moment vector

$$\mathbf{h} = [h_1 \ h_2 \ \cdots \ h_8]^T$$

is elaborated as below.

The $(u + v)$th order raw images moments G_{uv} for an intensity images $M(x, y)$ are computed by

$$G_{uv} = \sum_x \sum_y x^u y^v M(x, y).$$

Central moments are computed as

$$\mu_{uv} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x - \bar{x})^u (y - \bar{y})^v M(x, y) dx dy,$$

where

$$\bar{x} = \frac{G_{10}}{G_{00}} \ , \ \bar{y} = \frac{G_{01}}{G_{00}}.$$

Thus central moment of order up to 3 are:

$$\mu_{00} = G_{00},$$

$$\mu_{10} = 0,$$

$$\mu_{01} = 0,$$

$$\mu_{11} = G_{11} - \bar{x} G_{01},$$

$$\mu_{20} = G_{20} - \bar{x}G_{10},$$

$$\mu_{02} = G_{02} - \bar{y}M_{01},$$

$$\mu_{21} = G_{21} - 2\bar{x}G_{11} - \bar{y}G_{20} + 2\bar{x}^2G_{01},$$

$$\mu_{12} = G_{12} - 2\bar{y}G_{11} - \bar{x}G_{02} + 2\bar{y}^2G_{10},$$

$$\mu_{30} = G_{30} - 3\bar{x}G_{20} + 2\bar{x}^2G_{10},$$

$$\mu_{03} = G_{03} - 3\bar{y}G_{02} + 2\bar{y}^2G_{01}.$$

Normalized central moments are defined as:

$$\eta_{uv} = \frac{\mu_{uv}}{(\mu_{00})^{(1+\frac{u+v}{2})}}$$

where $u + v \geq 2$. Thus, eight Hu moments are computed as

$$\begin{aligned}
h_1 &= \mu_{20} + \mu_{02} \\
h_2 &= (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2 \\
h_3 &= (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2 \\
h_4 &= (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2 \\
h_5 &= (\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] \\
&\quad +(3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \\
h_6 &= (\mu_{20} - \mu_{02})[(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^{2]}] + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03}) \\
h_7 &= (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12})[(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] \\
&\quad -(\mu_{30} - 3\mu_{12})(\mu_{21} + \mu_{03})[3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] \\
h_8 &= \mu_{11}[(\mu_{30} + \mu_{12})^2 - (\mu_{03} + \mu_{21})^2] - (\mu_{20} - \mu_{02})(\mu_{30} + \mu_{12})(\mu_{03} + \mu_{21})
\end{aligned}$$

Data transformation from cMHI, dMHI to Hu moment vector can be expressed as

$$M_{\hat{k}}^{c} \Rightarrow \mathbf{h}_{\hat{k}}^{c} \ , \ M_{\hat{k}}^{fd} \Rightarrow \mathbf{h}_{\hat{k}}^{fd} \ , \ M_{\hat{k}}^{bd} \Rightarrow \mathbf{h}_{\hat{k}}^{bd}$$

For representing an ordinary RGB video:

$$H^{c} = [\mathbf{h}_{1}^{c} \ \mathbf{h}_{2}^{c} \ \cdots \ \mathbf{h}_{\hat{k}}^{c} \ \cdots \ \mathbf{h}_{\lceil \frac{K}{\alpha} \rceil}^{c}],$$

and for representing an RGB-Depth video, Hu moment vectors for cMHI, fdMHI and bdMHI are concatenated to form a feature vector of a single video frame

$$\mathbf{h}_{\hat{k}}^{3D} = [\mathbf{h}_{\hat{k}}^{c\ T} \ \ \mathbf{h}_{\hat{k}}^{fd\ T} \ \ \mathbf{h}_{\hat{k}}^{bd\ T}]^{T},$$

and an RGB-Depth video with K frames are defined as feature matrix

$$H^{3D} = [\mathbf{h}_{1}^{3D} \ \mathbf{h}_{2}^{3D} \ \cdots \ \mathbf{h}_{\hat{k}}^{3D} \ \cdots \ \mathbf{h}_{\lceil \frac{K}{\alpha} \rceil}^{3D}].$$

4.2.3 K-means clustering for vector quantization

Vector quantization is accomplished by k-means clustering as in Section 3.2.3. Details of the k-means clustering algorithm are explained in Section 3.2.3. Feature matrix of one single video is converted to observation sequence

$$H^{c} \Rightarrow \rho^{c} \ , \text{ and } H^{3D} \Rightarrow \rho^{3D}.$$

4.2.4 HMM based activity classification

A similar HMM training and testing algorithm as in Section. 3.2.3 is applied in this application, thus related concepts and processes are not duplicated here. All video clips are divided into two groups, bed-exiting activities and rolling-on-

bed activities. Further, each group is divided into two kinds, i.e. activity with or without a bed sheet. HMM named "Ex" indicate bed-exiting, and HMM named "Ro" indicates rolling-on-bed.

4.3 Experimental results of activity classification

Video clips collected in the preliminary data collection sections are used for evaluating the proposed activity classification approach, thus only cMHIs are extracted for algorithm testing, and the cycle of cMHI generation is set to be $\alpha = 20$. Single round cross validation and leave-one-subject-out approaches are applied to evaluate the capability of proposed algorithm to discriminate bed-exiting activities from rolling-on-bed activities.

Single round cross validation

For cross validation, covered activities and uncovered activities are mixed together in both models "Ex" and "Ro". Among each model, 75% of the given activities are randomly selected to be used as training sequences, and the rest 25% are testing sequences, which is illustrated in Table 4.4. The test result is displayed in Table 4.5.

Bed-exiting		Rolling-on-bed	
132		40	
Training	Testing	Training	Testing
99	33	30	10

Table 4.4: Selection of training and testing data

Confusion matrix shows a relatively good result, that all activities are correctly classified, thus the recognition rate is 100%.

Number of activities	Ex	Ro
Ex	33	0
Ro	0	10

Table 4.5: Confusion matrix of cross validation approach.

Leave-one-subject-out

In this approach, each time, performances of only one subject are selected as testing data for HMMs, and performances of the rest 14 subjects are chosen as training data. Here all activities are partitioned into "covered activities" and "uncovered activities" in advance. Thus we build up 4 HMMs in total: 1) "Cov-Ex" indicates covered bed-exiting, 2) "Cov-Ro" indicates covered rolling-on-bed, 3) "Un-Ex" indicates uncovered bed-exiting, and 4) "Un-Ro" indicates uncovered rolling-on-bed. Covered activities are trained and tested separately from the uncovered ones. This approach was applied to every subject, hence we obtain 15 pairs of confusion matrices (covered and uncovered pairs), thus all the video clips are tested/classified by HMM. Finally we add up all confusion matrices separately for the two experiment groups, and the result is shown in Table 4.6.

Activities	Cov-Ex	Cov-Ro
Cov-Ex	59	1
Cov-Ro	1	24

Activities	Un-Ex	Un-Ro
Un-Ex	64	8
Un-Ro	2	13

Table 4.6: Confusion matrix of leave-one-subject-out approach.

Recognition rate of bed-exiting activities with a bed sheet on is 98%, and 88.9% with a bed sheet off. Here, an interesting phenomenon is observed, that recognition rate with a bed sheet on is higher than that with a bed sheet off. Explanations of this phenomenon will be discussed in Section 4.5.

Precision and recall and the F1 score of each of the four models with $\alpha = 20$ are presented in Table 4.7. An average F1 score of 0.90 is reached with a lowest 0.72 and a highest 0.98.

	Cov-Ex	Cov-Ro	Un-Ex	Un-Ro
Precision	0.98	0.96	0.97	0.62
Recall	0.98	0.96	0.89	0.87
F1 score	0.98	0.96	0.93	0.72

Table 4.7: Precision, recall and F1 score of each model for bed-exiting with $\alpha = 20$.

Effect of choice of α on activity classification

As discussed earlier in Section 4.2.1, choice of α might affect the performance of classification. To observe this, results of activity classification with $\alpha = 5$ are demonstrated in Table 4.8 for comparison with results shown above with $\alpha = 20$.

Activities	Cov-Ex	Cov-Ro
Cov-Ex	56	4
Cov-Ro	2	23

Activities	Un-Ex	Un-Ro
Un-Ex	52	20
Un-Ro	0	15

Table 4.8: Confusion matrix of the leave-one-subject-out approach.

Recognition rate of covered bed-exiting activity is 93.3%, covered rolling-on-bed activity 92.0%, uncovered bed-exiting activity 72.2%, and uncovered rolling-on-bed activity 100%. In general, results with $\alpha = 5$ are similar to that with $\alpha = 20$, in which covered bed-exiting activities are better recognized than uncovered ones. But recognition rate of each model (except model Un-Ro) drops with smaller α, especially for the uncovered bed-exiting model. This result is consistent with our inference in Section 4.2.1.

Precision and recall and the F1 score of each of the four models with $\alpha = 5$ are presented in Table 4.9. The average F1 score is 0.82 with a lowest 0.6 and a highest 0.95. An obvious drop of F1 score compared with that with $\alpha = 20$ is observed. This also supports our inference about the effect of value of α.

	Cov-Ex	Cov-Ro	Un-Ex	Un-Ro
Precision	0.97	0.85	1.00	0.43
Recall	0.93	0.72	0.72	1.00
F1 score	0.95	0.88	0.84	0.60

Table 4.9: Precision, recall and F1 score of each model for bed-exiting with $\alpha = 5$.

4.4 Experimental result of bed-exiting prediction

The classification results demonstrated above reveal the possibility of proposed algorithm to achieve bed-exiting prediction.

4.4.1 Introduction of prediction curve

Referring to Section 4.2.4, activity classification is achieved by calculating likelihood probabilities of a particular observation sequence generated by given HMMs, and selecting the HMM with the highest likelihood probability. Instead of using the entire observation sequence, if we compute the likelihood probability of every subset of the sequence (subsets are defined as sequences from the beginning to every observation) to form a probability curve, and observe and compare the tendency of curves given certain HMMs, we can achieve prediction of a bed-exiting activity, further to issue an early alarm. An observation sequence representing a video clip with K frames is denoted as

$$\rho = [\rho_1 \ \rho_2 \ \cdots \ \rho_{\hat{k}} \ \cdots \ \rho_{\lceil \frac{K}{\alpha} \rceil}]^T,$$

and the HMMs for bed-exiting and for rolling-on-bed activities are denoted as Λ_{Ex} and Λ_{Ro} respectively. Likelihood probability curve of ρ given Λ_{Ex} and Λ_{Ro} can be generated by;

$$f_{Ex}(\hat{k}, \rho) = \log P(\rho_1, \rho_2, \cdots, \rho_{\hat{k}} | \Lambda_{Ex}),$$

and

$$f_{Ro}(\hat{k}, \rho) = \log P(\rho_1, \rho_2, \cdots, \rho_{\hat{k}} | \Lambda_{Ro}),$$

where

$$\hat{k} = 1, 2, \cdots, \lceil \frac{K}{\alpha} \rceil.$$

To identify which model a subset belongs to, we compute the "prediction curve" as:

$$\begin{aligned} g(\hat{k}, \rho) &= \log \frac{P(\rho_1, \rho_2, \cdots, \rho_{\hat{k}} | \Lambda_{Ex})}{P(\rho_1, \rho_2, \cdots, \rho_{\hat{k}} | \Lambda_{Ro})} \\ &= f_{Ex}(\hat{k}, \rho) - f_{Ro}(\hat{k}, \rho) \end{aligned}$$

If $g(\hat{k}, \rho) > 0$, it means subset $[\rho_1 \ \rho_2 \ \cdots \ \rho_{\hat{k}}]$ is classified as a bed-exiting activity, otherwise it is classified as a normal rolling-on-bed activity. Figure 4.5 gives an example of likelihood probability curve on the left and prediction curve on the right for an bed-exiting activity ρ_{Ex}. In the likelihood probability curve, red line represents $f_{Ex}(\hat{k}, \rho_{Ex})$, and green represents $f_{Ro}(\hat{k}, \rho_{Ex})$. For contrasting, Figure. 4.6 gives an example of these 2 types of curves for an rolling-on-bed activity ρ_{Ro}. Frame rate of raw video of these two samples is $\omega = 25$ FPS, and the cycle of MHI generation $\alpha = 5$. Time interval between 2 successive MHI frames Δt can be calculated by

$$\Delta t = \frac{\alpha}{\omega}$$

In Figure 4.5, the prediction curve strides over zero line from negative to positive, and keeps rising, which means that the possibility of the testing sequence being a bed-exiting activity becomes higher with time. In contrast, in Figure 4.6 the overall

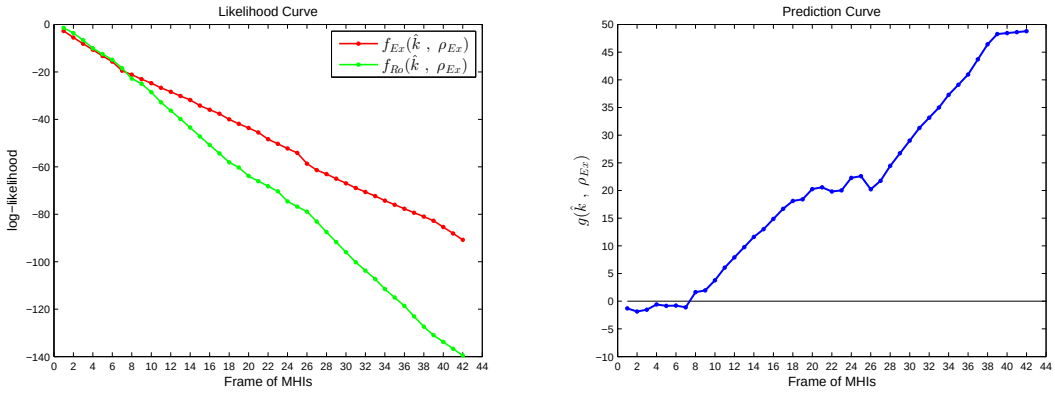

Figure 4.5: Likelihood probability curve and prediction curve of a bed-exiting activity.

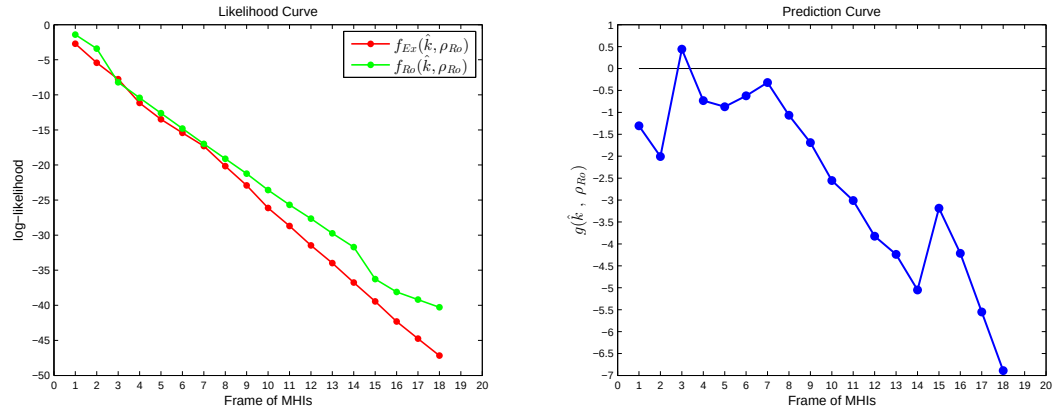

Figure 4.6: Likelihood probability curve and prediction curve of a rolling-on-bed activity.

trend of the prediction curve is going down beneath the zero line, but it can be noticed that there is a point on this curve at the beginning jumps over the zero line. This is because at the beginning of the testing process, behaviour of testing sequence hasn't been fully handled by the HMM, thus the first few points in a prediction curve cannot be used for prediction. Criterion to issue an early alarm should be a continuous presence of N positive points on a prediction curve. And how many positive points should be count (value of N) will depend on the timing

requirement of prediction and the required confidence level of activity classification. Since for the same α, larger N can increase the classification confidence level but will shorten the early alarm time.

4.4.2 Experimental results on early alarm time evaluation

In this section, we choose $N = 4$ to evaluate how early a prediction can be issued by our proposed algorithm. The early prediction duration is defined by its start point on a prediction curve and the end point in the video, and definition of start and end point is elaborated as below

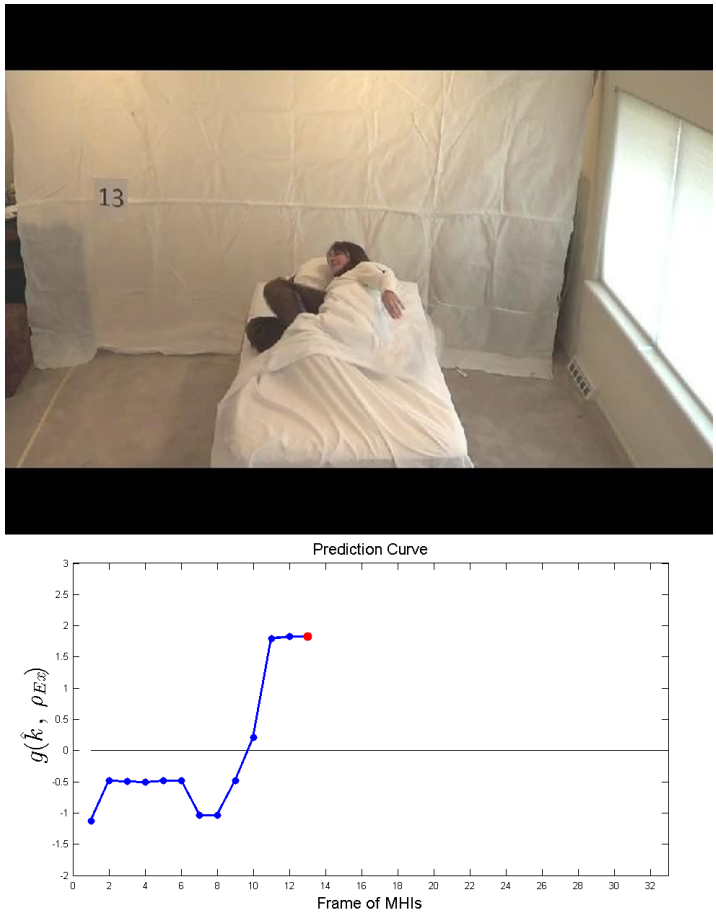

Figure 4.7: Example of the start point for a bed-exiting prediction.

Start point The Nth positive point of the first segment on a prediction curve, which consists of N continuous positive points. The start point corresponds to the MHI frame with frame index $\hat{K}_S$, and corresponds to the raw RGB image with frame index K_S.

End point The frame in raw video, in which the person just moves to the position to sit on the edge of a bed. And the end point corresponds to the raw RGB image with frame index K_E.

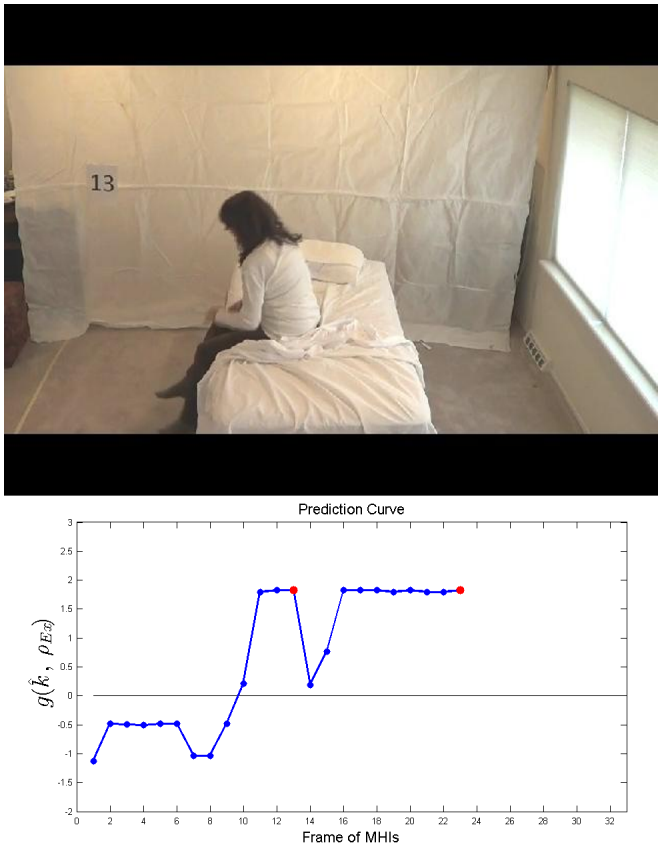

Figure 4.8: Example the end points for a bed-exiting prediction.

The early alarm time T_a is calculated by:

$$T_a = \frac{K_E - K_S}{\omega} = \frac{K_E - \hat{K}_S \cdot \alpha}{\omega}$$

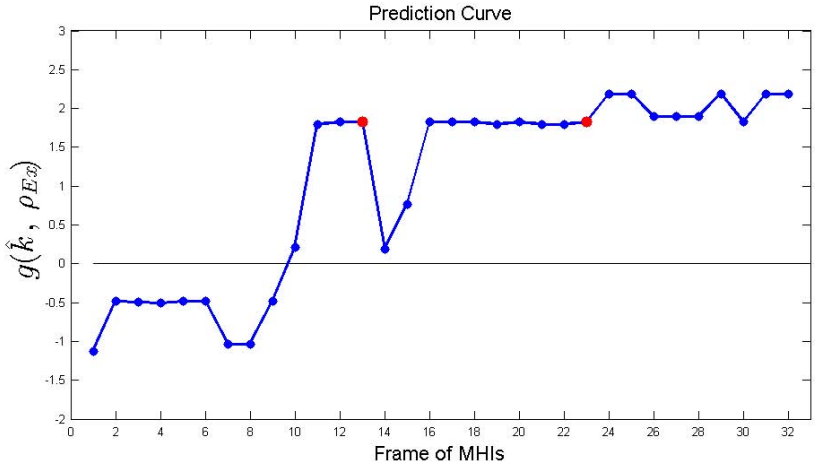

Figure 4.9: Full prediction curve of the sample video.

Figure 4.7 and 4.8 provide an example of the start and end points for a video with $\omega = 25$ FPS, $\alpha = 5$ and $N = 4$. And full prediction curve of this sample video is shown in Figure 4.9.

The red point on Figure 4.7 represents the start point, and indicates that an early alarm of bed-exiting can be issued at this point. And the second red point on Figure 4.8 represents the end point, which means the bed-exiting activity ends at this point. Full prediction curve of this sample video is shown in Figure 4.9. Early alarm time for video in Figure 4.9 is

$$T_a = \frac{115 - 13 \times 5}{25} = 2.00s.$$

More prediction curves for different video samples of different volunteers are demonstrated in Figure 4.10, with $T_a = 3.76s$, $3.28s$ and $1.56s$ respectively. The prediction curves for all bed-exiting activities in the preliminary data collection section are plotted, and 107 of them provide feasible prediction, and the average early alarm time is $\bar{T_a} = 2.48$ seconds.

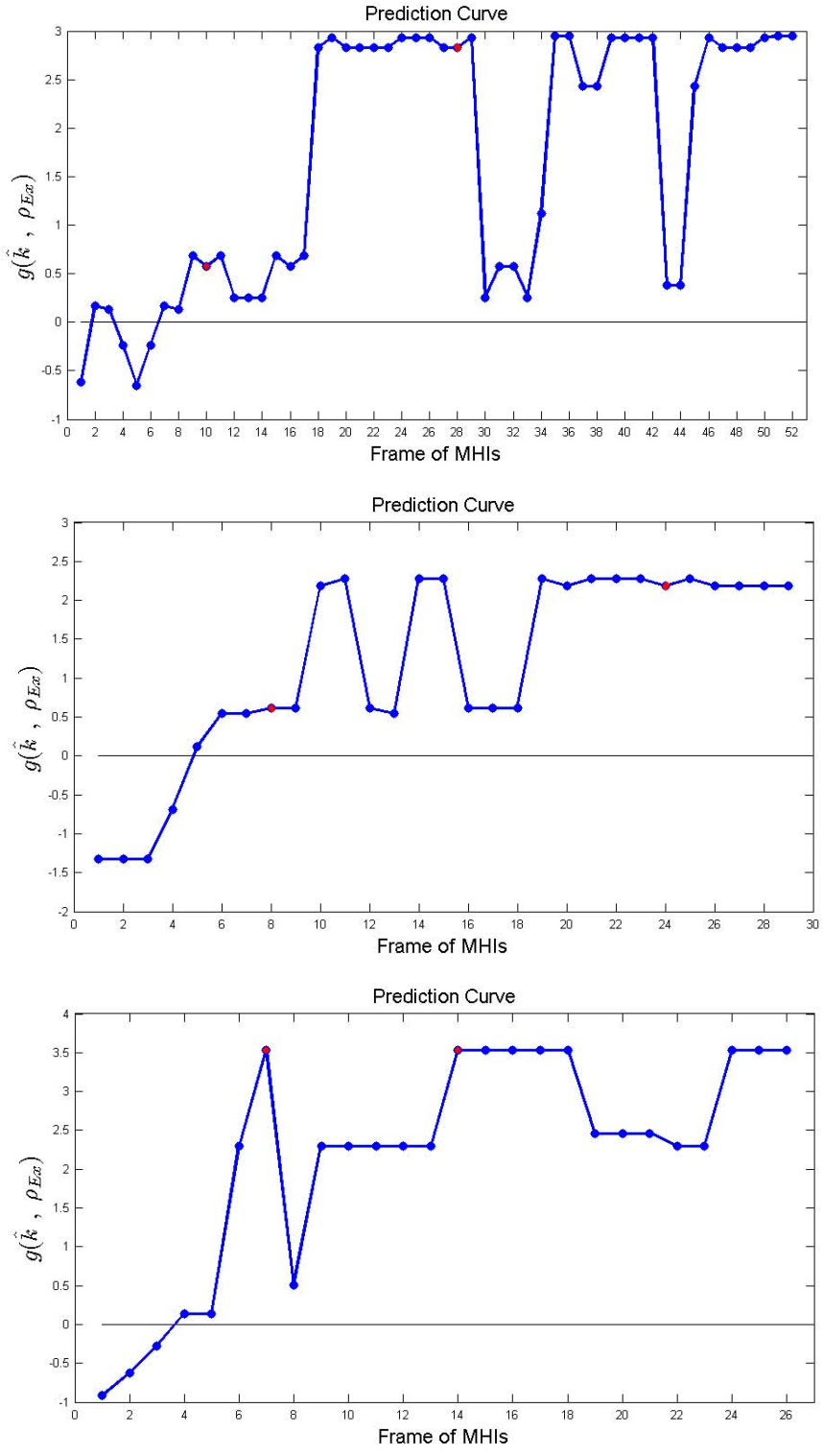

Figure 4.10: Sample bed-exiting prediction curves.

4.5 Discussion

Regarding the activity classification results with or without a bed sheet in Section 4.3, in our original presumption, bed-exiting activities without a bed sheet might be easier to detect, because more motion features of human trunk, legs and arms can be recognized, and this could be great help to distinguish a bed-exiting activity from a rolling-on-bed one. Experimental results explode the original presumption, and we come out with 3 possible explanations for this:

1. Bed-exiting activities with a bed sheet all, or at least most of them, share a common movement, that is removal of the sheet before exiting the bed. And the change from on-bed movements under a sheet before the removal to movements without a sheet after the removal can be easily detected by MHIs. Thus, movement sequences of removing a bed sheet might be a critical element to detect a bed-exiting activity.

2. Moving tendency of uncovered body parts, head and shoulder, is strong enough a motion feature to estimate a person's motion, and a bed sheet helps filtering out the unnecessary motion features of lower body.

3. Lengths of videos about activities with a bed sheet are generally longer than that without a sheet. And this provides longer training sequences for HMM, and longer decision time in the testing process.

Regarding the effect of different values of MHI generation cycle α, the results is consistent with our inference in Section 4.2.1, and illustrate that choice of α need to trade off between higher time precision in prediction and higher recognition rate of bed-exiting activities.

Chapter 5

Conclusion and future work

5.1 Conclusion

This thesis presents human motion analysis for fall detection and estimation based on two different approaches. All data collections are completed by the inexpensive Kinect sensor, which can record human skeletal images, RGB images and depth images.

The first approach relies on the "*skeleton viewer*" of Kinect sensor to obtain human motion features. This approach is applied in a general fall detection project. Seven HMMs corresponding to seven innovative motion classes are trained to discriminate different fall activities and non-fall activities. Simulated falls in a laboratory environment are used for algorithm testing. Experimental results verifies the feasibility of this approach for the general fall detection application, that an average recognition rate of 80% is achieved among different motion classes, and a recognition rate of 95% is achieved for fall activities.

The second approach is the segmented MHI sequence based approach, which is applied for bed-exiting detection. Two HMMs, bed-exiting and rolling-on-bed, are built using observation sequences derived from MHIs. We use both single round cross

validation and leave-one-subject-out to test the capability of this approach for discrimination between bed-exiting and rolling-on-bed activities, and both approaches generate promising results. Recognition rate of bed-exiting activity is 100% for the single round cross validation, and 98% with a bed sheet on, 88.8% with a bed sheet off for leave-one-subject-out. To observe the effect of the cycle duration parameter α, we compare the classification results with different values of α. Then we propse and investigate the prediction curve to explore capability of proposed algorithm for an early alarm of a bed-exiting activity, and evaluate how early an alarm can be issued. Average early alarm time with $\alpha = 5$ can achieve 2.48 seconds.

The proposed two motion estimation approaches are designed for real-time fall detection and bed-exiting detection applications. The data processing and testing mentioned in this thesis runs on a PC in MATLAB. For the fall detection application, coordinates of human body joints are exported once they are detected, which is not a concern for in real-time implementation. The establishment of feature database and training of HMM models are conducted off-line before they are applied into real-time application. The real-time implementation considerations need to address the issues of dimensionality reduction, vector quantization and HMM test for raw measurement data. For the bed-exiting application, generation of MHIs and computation of Hu moments require only arithmetic of addition, subtraction, comparison and multiplication, and these algorithms are easy to realize for real-time applications. Similar to the fall detection approach, feature database and HMM models will be prepared off-line.

5.2 Future work

5.2.1 RGB-Depth fusion for head-shoulder tracking

Based on the discussion on value of α in Section 4.3, we notice that pure motion features of head-shoulder movement could be a good studying material. For bed-exiting we propose a 3D head-shoulder tracking algorithm by combining RGB and depth images recorded by the Kinect sensor. This algorithm can plot 3D trajectory of head-shoulder for movement tendency estimation.

Feature extraction by histogram of oriented gradients descriptors

Head-shoulder feature extraction is achieved only on RGB images by histogram of oriented gradients descriptors which were first introduced by Dalal and Triggs in [30]. And a head-shoulder detection model is trained by a set of positive and negative training data. For initial algorithm testing, we use images of one bed-exiting movement from one subject. A rectangular region containing the head and shoulder is manually selected in every frame as positive training data, and the rest area of each frame are used as negative training data. Then the trained model is used to detect head-shoulder region on the training images themselves.

Head-shoulder tracking by data association

By actuating the head-shoulder detection, several regions might be detected in a frame due to false alarms. A very basic data association procedure is applied to associate the nearest detected region in consecutive frames and to filter out those false alarm regions.

RGB-D fusion

Once the head-shoulder regions in every frame are correctly detected and successfully tracked through the entire video, center point of a detected region on the RGB image, which corresponds the point of interest, is chosen as the point of interest $A(x_a, y_a)$. Azimuth angle $\epsilon(A)$ and altitude angle $\omega(A)$ of pixel $A(x_a, y_a)$ are roughly calculated with respect to the center point of the entire image $C(x_c, y_c) = C(320.5\ ,\ 240.5)$. Since viewing angle of a Kinect sensor is 43^o vertical by 57^o horizontal, image size is 640×480, $\epsilon(A)$ and $\omega(A)$ are calculated by

$$\epsilon(A) = (x_a - x_c)\ \frac{57^o}{640},$$
$$\omega(A) = (y_a - y_c)\ \frac{43^o}{480}.$$

By associating point of interest $A(x_a, y_a)$ on RGB image with the corresponding point $A^D(x_a^D, y_a^D)$ on the corresponding depth image, we can obtain distance from the point of interest in real space to the Kinect sensor $d(A)$ in millimeters. Coming to this step, we obtain the rough 3D information of the point of interest as $[\epsilon(A)\ \omega(A)\ d(A)]$, and then by connecting all points of interest in every frame we can achieve the moving trajectory of head-shoulder. Figure. 5.1 displays several image results of the proposed algorithm. The image on the left side demonstrates the head-shoulder region in the RGB images with value of $[\epsilon(A)\ \omega(A)]$ on top of the region, and image on the right side demonstrates the head-shoulder region on depth images with value of $d(A)$ on top of the region.

The basic object of using RGB-D images for location the head-shoulder area has been reached in current work, which lays the foundation of future improvement, refinement and large scale testing. Besides, once this methods is accomplished, it can be exploited not only in head-shoulder moving tendency estimation, but also in movement estimation of other human body parts.

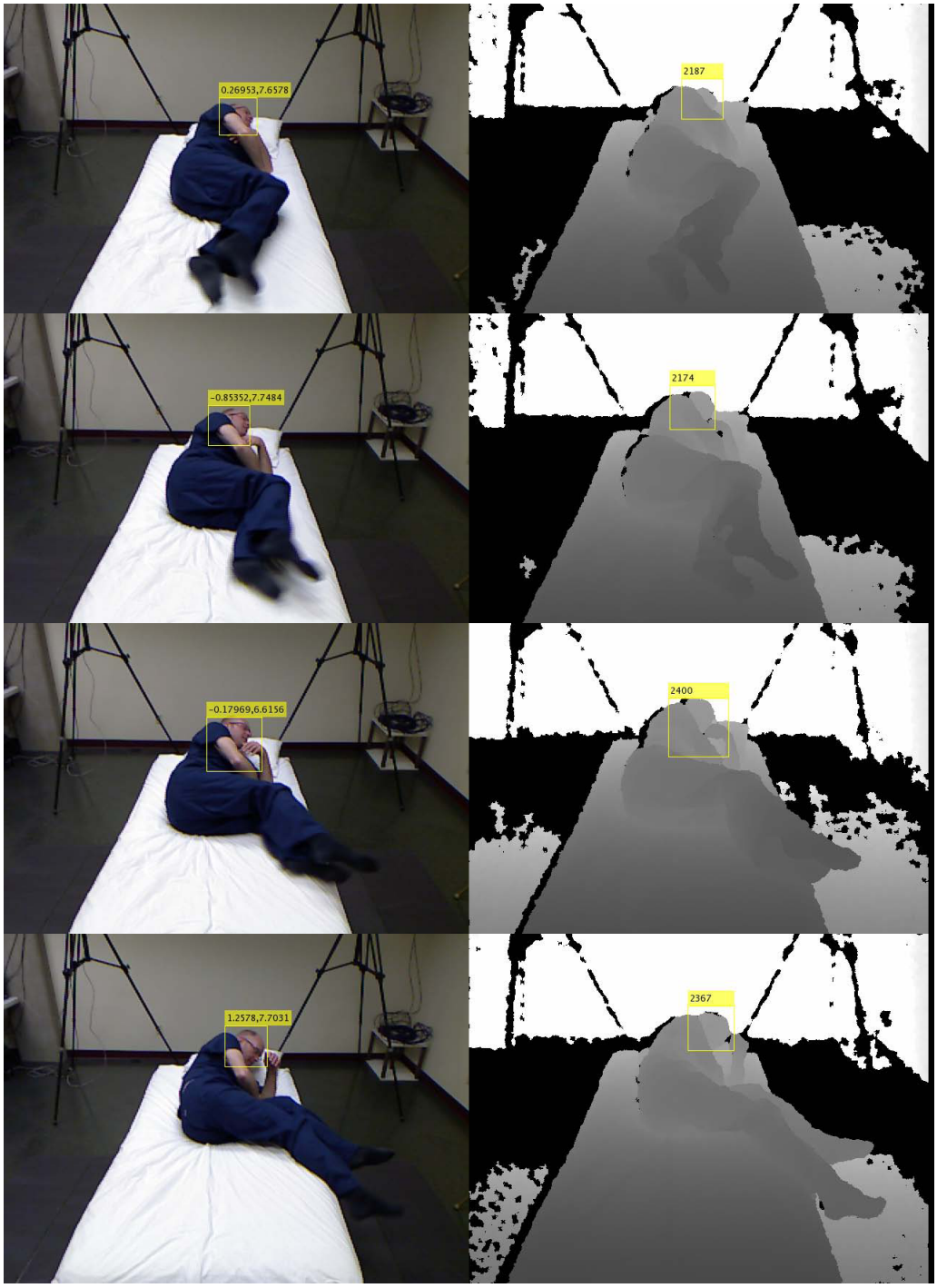

Figure 5.1: An example of head-shoulder tracking.

5.2.2 Other future work

1. Data fusion of cMHIs and dMHIs: As introduced in Chapter 4, dMHIs can provide information on changes in range, we hope to replenish this information to combine with cMHIs, aiming at enhancing classification and prediction capability.

2. Real-time implementation: As fall detection and bed-exiting prediction are both time-critical application, the realization of real-time implementation for proposed two approaches is another desired research topic for our future work.

Bibliography

[1] J. A. Stevens, P. S. Corso, E. A. Finkelstein, and T. R. Miller, "The costs of fatal and non-fatal falls among older adults," *Injury Prevention*, 2006.

[2] Centers for Disease Control and Prevention. National Center for Health Statistics., "National hospital discharge survey (NHDS)," Tech. Rep., www.cdc.gov/nchs/hdi.htm.

[3] P. Rajendran, A. Corcoran, B. Kinosian, and M. Alwan, *Eldercare Technology for Clinical Practitioners*, Humana Press, 2008.

[4] E. Capezuti, B. L. Brush, S. Lane, H. U. Rabinowitz, and M. Secice, "Bed-exit alarm effectiveness," *Archives of Gerontology and Geriatrics*, vol. 49, pp. 27–31, 2008.

[5] X. Yu, "Approaches and principles of fall detection for elderly and patient," in *10th e-health Networking, Applications and Services*, July 2008, pp. 42–47.

[6] M. A. Clifford, R. L. Borras, and L. Gomez, "System and method for human body fall detection," 2007.

[7] N. Noury, P. Barralon, G. Virone, P. Boissy, M. Hamel, and P. Rumeau, "A smart sensor based on rules and its evaluation in daily routines," in *25th Engineering in Medicine and Biology Society*, 2003, vol. 4, pp. 3286 – 3289.

[8] T. Tamura, T. Yoshimura, Masaki Sekine, M. Uchida, and O. Tanaka, "A wearable airbag to prevent fall injuries," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 6, pp. 910–914, 2009.

[9] H. Ghasemzadeh, V. Loseu, and R. Jafari, "Structural action recognition in body sensor networks: Distributed classification based on string matching," *IEEE Transactions on Information Technology in Biomedicine*, vol. 14, no. 2, pp. 425–435, 2010.

[10] M. Alwan, P. J. Rajendran, S. Kell, D. Mack, S. Dalal, M. Wolfe, and R. Felder, "A smart and passive floor-vibration based fall detector for elderly," in *2nd Information and Communication Technologies*, 2006, pp. 1003–1007.

[11] T. E. Scott, "Bed exit detection apparatus," 2000.

[12] Yun Li, K.C. Ho, and M. Popescu, "A microphone array system for automatic fall detection," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 5, pp. 1291–1301, 2012.

[13] H. Rimminen, J. Lindstrom, M. Linnavuo, and R. Sepponen, "Detection of falls among the elderly by a floor sensor using the electric near field," *IEEE Transactions onInformation Technology in Biomedicine*, vol. 14, no. 6, pp. 1475–1476, 2010.

[14] A. Sixsmith and N. Johnson, "A smart sensor to detect the falls of the elderly," *IEEE Computer Society Conference on Pervasive Computing*, vol. 3, no. 2, pp. 42–47, 2004.

[15] B. Jansen and R. Deklerck, "Context aware inactivity recognition for visual fall detection," in *Pervasive Health Conference and Workshops*, 2006, pp. 1–4.

[16] B.U. Toreyin, Y. Dedeoglu, and A.E. Cetin, "HMM based falling person detection using both audio and video," in *14th Signal Processing and Communications Applications*, April 2006, pp. 1 –4.

[17] M. P. Rigney, "Before the fall," *Technology Today*, pp. 12–13, Summer 2009.

[18] N. Yamanaka, H. Satoh, and F. Takeda, "Development of an awakening detection system with the NN and adaptation for fluctuation of brightness quantity in the captured image," *Internation Journal of Intelligent Systems Technologies and Applications*, vol. 9, pp. 286–299, 2010.

[19] H. Satoh, T. Ohkura, and F. Takeda, "Proposal for the awakening behavior detection system using images and adaptation for fluctuation of brightness quantity in the captured image," in *4th International Conference on Modeling, Simulation and Applied Optimization (ICMSAO)*, 2011, pp. 1–7.

[20] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau, "Monocular 3D head tracking to detect falls of elderly people," in *28th Engineering in Medicine and Biology Society*, 2006, pp. 6384 –6387.

[21] A.F. Bobick and J.W. Davis, "The recognition of human movement using temporal templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 3, pp. 257–267, 2001.

[22] G. R. Bradski and J. Davis, "Motion segmentation and pose recognition with motion history gradients," *5th IEEE workshop on Applications of Computer Vision*, 2000.

[23] J. W. Davis and A. F. Bobick, "The representation and recognition of human movement using temporal templates," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1997, pp. 928–934.

[24] J. W. Davis, "Hierarchical motion history images for recognizing human motion," in *IEEE Workshop on Detection and Recognition of Events in Video*, 2001, pp. 39–46.

[25] J. Davis and G. Bradski, "Real-time motion template gradients using Intel CVLib citation," in *IEEE ICCV Workshop on Frame-rate Vision*, 1999.

[26] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau, "Fall detection from human shape and motion history using video surveillance," *21st International Conference on Advanced Information Networking and Applications Workshops*, 2007.

[27] P. Holliday, C. Gryfe, and G. Griggs, "Accidental falls of elderly people," Centre for Studies in Aging, University of Toronto, 1989.

[28] B. Ni, G. Wang, and P. Moulin, "RGBD-HuDaAct: A color-depth video database for human daily activity recognition," in *IEEE International Conference on Computer Vision Workshops*, 2011, pp. 1147–1153.

[29] M. Hu, "Visual pattern recognition by moment invariants," *IRE Transactions on Information Theory*, vol. 8, no. 2, pp. 179–187, 1962.

[30] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005.