IEEE ICCE 2014

2014 IEEE Fifth International Conference on Communications and Electronics (ICCE)

Pullman Beach Resort Da Nang
July 30 - August 1, 2014

IEEE

WELCOME

Welcome to the 2014 IEEE Fifth International Conference on Communications and Electronics (IEEE ICCE 2014) integrated in a USB version.

ICCE is becoming a reputable biennial international conference series in the scientific community on the areas of Electronics and Communications recently. Following the past successful conferences, the fifth IEEE ICCE 2014 looks for significant contributions to various topics in communication engineering, networking, microwave engineering, signal processing and electronics engineering. The conference will also include tutorials, workshops, and technology panels given by world-class speakers.

At the conference, two hundred and twenty eight (228) papers from more than 30 countries have been submitted. Among these submissions, ninety six (96) regular full papers, which will be submitted for inclusion into IEEE Xplore and twenty nine (29) poster papers have been accepted for presentation and will be organized in 18 regular and 2 special sessions. The Opening Session will host 4 keynotes. Four tutorials offered by the conference widely cover the most interesting topics on electronics and communications engineering, whose issues related to Quality of Mobile Multimedia Experience, Immersive Visual Communication with Depth, Crowdsourcing, and Free Space Materials Characterization.

The technical program focuses on hot topics on the fields of Communications Networks and Systems, Signal Processing and Applications, Microwave Engineering, and Electronic Systems. Besides, the special sessions on Crowdsourcing and Crowdsourcing Applications, and on Information Hiding and Security in Communications: Recent Developments have been added to the technical program of this event.

Beside the printed proceeding, this edition in USB is designed for readers to locate the papers by session or authors. Papers are originated as electronic files and were converted to Adobe Acrobat PDF file format for a crossplatform access. Even though the viewing quality on your monitor may vary, all papers have been printed clearly.

Enjoy your reading through!

IEEE Catalog Number: CFP1416B-USB
ISBN: 978-1-4799-5050-8

TABLE OF CONTENTS

An analysis on human fall detection using skeleton from Microsoft Kinect

Thi-Thanh-Hai Tran, Thi-Lan Le
International Research Institute MICA
HUST - CNRS/UMI-2954 - GRENOBLE INP
Hanoi University of Science and Technology
{thanh-hai.tran, thi-lan.le}@mica.edu.vn

Jeremy Morel
ENSEIRB-MATMECA
Bordeaux Institute of Technology
Bordeaux, France

***Abstract*— In this paper, we present a novel fall detection system based on the Kinect sensor. The originalities of this system are two-fold. Firstly, based on the observation that using all joints to represent human posture is not pertinent and robust because in several human postures the Kinect is not able to track correctly all joints, we define and compute three features (distance, angle, velocity) on only several important joints. Secondly, in order to distinguish fall with other activities such as lying, we propose to use Support Vector Machine technique. In order to analyze the robustness of the proposed features and joints for fall detection, we have performed intensive experiments on 108 videos of 9 activities (4 falls, 2 falls like and 3 daily activities). The experimental results show that the proposed system is capable of detecting falls accurately and robustly.**

Keywords—Kinect sensor; Skeleton; Support Vector Machine

I. INTRODUCTION

Falls in elderly is a major public problem. Several studies have shown [1] that elderly people experience at least one fall every year. Moreover, falls are the main cause of accidental death in older adults aged 65 or more [2]. Therefore, fall detection has attracted a large attention of researchers as well as industrial. A huge number of material, equipment and fall detection methods have been proposed. Among these methods, vision based method has a great advantage because it does not require the elderly people to wear a specific equipment.

Recently, Kinect device has been released. The use of this sensor for health care application in general and fall detection in particular has two main advantages. Firstly, this sensor provides rich information (color, depth and skeleton) in comparison with conventional cameras and therefore gives a better representation of fall. Secondly, this device allows developing a solution for protecting privacy - major issue in health care application by using only depth and skeleton information.

In this paper, we present a novel fall detection system based on the Kinect device. The originalities of this system are two-fold. Firstly, based on the observation that using all joints is not pertinent and robust to represent human posture because in several human postures the Kinect is not able to track correctly all joints, we define and compute three features (distance, angle, velocity) on only several important joints. Secondly, in order to distinguish fall with the others activities such as lying, we propose to use SVM (Support Vector Machine). The experimental results show that the proposed system is capable of detecting falls accurately and robustly.

The remaining of this paper is organized as follows. The section II gives a brief survey on vision-based fall detection approaches in general and Kinect-based fall detection in particular. Then, our proposed method is described in detail in Section III. Section IV presents experimental results as well as discussions. Finally, we give some conclusions and future works in Section V.

II. RELATED WORKS

The fall of a person can be described as the rapid change from upright/sitting position to the reclining or almost lying position. The fall is not a controlled movement like lying for example. According to [3], the fall may be divided into four phases: prefall, critical, postfall and recovery.

Over the last years, there are a number of works that have been proposed for fall detection. These works can be classified according to whether they focus on direct detection of the critical phase/impact shock or postfall phase. In this paper, we do not try to do an exhaustive survey of all state of the art works for fall detection that is the scope of the others papers [3]. We present a brief summary of vision-based fall detection. As our work is based on Kinect device, we try to analyze the related works for fall detection using Kinect.

Since Kinect device provides color, depth image and skeleton information. The Kinect-based fall detection work can be classified according information source. The works belonging to the first category explores color and depth information while the second category attempts to use skeleton information to detect the fall.

Concerning fall detection using depth information, Rougier et al. [4] introduce a Kinect-based fall detection system. In this system, a fall will be recognized using two parameters: the distance between subject's centroid and the floor and the velocity of the center of mass. In this work, the

978-1-4799-5051-5/14/$31.00 ©2014 IEEE

floor is detected by a histogram analysis of a V-disparity image. In [5] (Fall Detection on Embedded Platform Using Kinect and Wireless Accelerometer), the authors combined Kinect and Accelometer to detect the fall. Concerning fall detection based on Kinect, the authors detect the foreground object (person) by using differencing technique on depth images. After that, the center of gravity of the person and the distance between the person's gravity center and the altitude of the Kinect are calculated. Finally, the fall can be detected by applying predefined rules. This work cannot distinguish the fall and Activities of Daily Living (ADL) such as brutally sitting on the floor. In Mastorakis et al. work [6], the authors exploit color and depth information to build 3D (height, width, depth) bounding box of the subject. Thus, two velocities (velocity in height and velocity in depth-width) are computed in order to define the fall.

The work presented in [7] belongs to the second category. In [7], the authors developed two algorithms to detect falls using the skeleton information provided by Kinect SDK. The first algorithm uses only joint position data while the second one calculated the velocity of joints. Concerning the first algorithm, firstly the maximum distance of all joints with the floor is determined. Then, if this distance is smaller than a predefined threshold, a fall is recognized. The second algorithm computes the average velocity of all joints in several frame and determines the fall by comparing this average velocity with a predefined threshold.

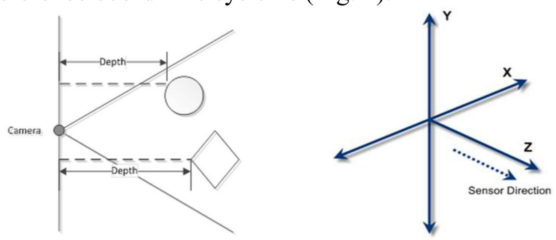
III. PROPOSED METHOD FOR HUMAN FALL DETECTION

A. Kinect overview

Before describing the proposed method for human fall detection, we make a brief overview of Kinect for a better understanding of Kinect sensor, Kinect SDK, coordinate systems used in our work.

A Kinect device has 3 sensors: a standard camera that gives RGB image; an IR camera detects points projected by a laser and automatically converts them into a depth map; a microphone array. The Kinect SDK is a free software package that provides lots of useful tools to exploit information provided by Kinect. For example, in our work, we use 3D location of skeleton joints and floor plane equation.

When working with Kinect, we need to understand some reference coordinate systems (Fig. 1).



(a) (b)

Fig. 1. (a) Depth space; (b) Skeleton space

1) Depth space

For each frame, the depth sensor captures a gray scale image representing the “depth” of things in the scene to the sensor. The value of each pixel is the Cartesian distance, in millimeters, from the camera plane to the nearest object at that particular (x, y) coordinate, as shown in Fig. 1a.

2) Skeleton space

For every frame, the depth image is processed by the Kinect runtime to convert into skeleton data. Skeleton data contains 3D positions of each skeleton joint is stored as (x, y, z) coordinates. Unlike depth space, skeleton space coordinates are expressed in meters. The x, y, and z-axes are the body axes of the depth sensor as shown in Fig. 1b.

B. Schema of the proposed system

The method for human activity classification that we propose is based on skeleton information of the human. Specifically, when the human performs a certain activity, the skeleton joints move. We characterize this movement by three features: the distance from each joint to the floor plane, the angle between the line linking a considered joint to the head joint and the normal of the floor plane, the velocity of each joint in the coordinate of the floor plane.

As each activity will be performed in a duration of time, normally, each activity will be represented by a sequence of three features set. The sequence length could change between activities and vary from one human to another. To simplify the activity representation, we will not take all values of angle, velocity, and distance in the sequence but we take only the most significant one of each feature type. In addition, only upper body joints are interesting to characterize fall activities. So in our paper, we propose to consider eight upper body joints as presented in the Tab. 1.

Each activity now is represented by a set of values (distance, velocity, angle) of several joints of human skeleton, that we call feature vector. Feature vectors will be inputs for training or testing Support Vector Machine (SVM) classifiers.

Fig. 2 shows different steps of the method. We will describe in more detail each step in the following sections.

- *Step 1:* Compute the floor plane of the room. This step is done one time at the beginning. Knowing the floor plane equation, we can compute the distance of any point (skeleton joint) to this plane.
- *Step 2:* Extract human skeleton. The Kinect SDK has proposed a tool to extract human skeleton. As in the section of Kinect overview, we will take 3D coordinates of each joint of human skeleton and convert into the coordinate of the plane floor.
- *Step 3:* Compute features of each joint (distance, velocity, angle) at every frame. Choose features that are most significant for activity representation
- *Step 4:* Classify activity using SVM model that we have learnt before using training dataset.

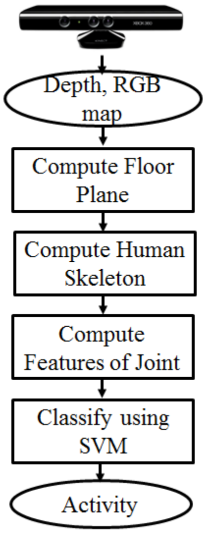


Fig. 2. Main steps of activity classification using skeleton information and SVM technique

C. Floor plane detection

As presented previously, the features used to recognize activities such as angle, velocity and position are all computed respect to the floor plane. Therefore first we need to determine floor plane. We have tried two methods for floor plane detection. One is based on V-disparity computation [4]. Another is based on plane equation already computed by Kinect SDK.

For V-disparity based method, we first compute the V-disparity image from depth map provided by Kinect. Then from V-disparity image, we apply the Hough transform to determine straight lines and choose one line having the depth appropriate to the depth of the floor plane in the scene.

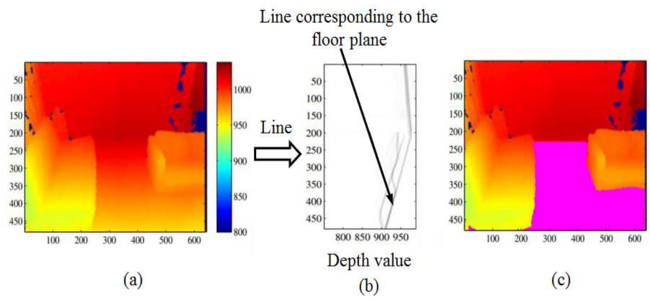


Fig. 3. (a) Depth map of the scene; (b) V-Disparity image and floor line determination; (c) Mapping floor in the depth map: magenta pixels belong to the floor plane (best viewed in color) [4]

Alternatively, Kinect SDK has provided a solution to capture floor plane. Specifically, it gives 4 values of the floor plane equation. However, when we tested with our scene, we found that Kinect SDK gives usually a floor plane with a lot of false positives. V-disparity method gives more accurate floor plane but it has some drawbacks: (i) it does not give stable result; (ii) it takes time due to the Hough transform. We will give analysis results in the experiment section using both methods of floor plane detection.

D. Feature extraction

Kinect SDK of Microsoft has provided 3D locations of 20 joints of human skeleton (Fig. 4) when the human is visible in the depth sensor's field of view.

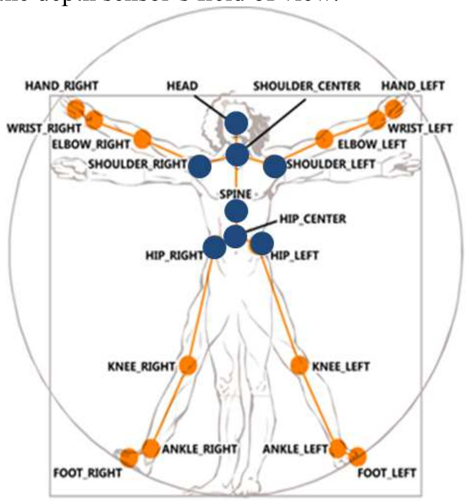


Fig. 4. 20 joints of human skeleton

However, all information is not pertinent in the framework of fall detection. In our proposed method, we consider 8 joints at most. These 8 joints (in blue color in Fig. 4) are combined to make 5 modes to experiment (Tab. 1).

TABLE I. DIFFERENT MODES FOR HUMAN FALL DETECTION

N0	Mode	Joints to be considered
1	1	Head
2	2	Head, Spine
3	4	Head, Shoulder Center, Spine, Hip Center
4	6	Head, Shoulder Center, Spine, Hip Center, Shoulder Right, Shoulder Left
5	8	Head, Shoulder Center, Spine, Hip Center, Shoulder Right, Shoulder Left, Hip Right, Hip Left

For each joint, we will compute 3 features: distance, angle and velocity (Fig. 5).

1) Distance

In our work, the distance of a joint is defined as the distance from this point to the floor plane. Given the floor plane equation, we can compute this distance easily.

$$d = \frac{Ax + By + Cz + D}{\sqrt{A^2 + B^2 + C^2}} \quad (1)$$

2) Angle

Angle of a joint is defined as the angle between the normal vector of the floor plane and the line linking the head joint and this joint.

3) Velocity

Velocity of a joint is defined as the difference of its positions at time *t* and time *t+1*.

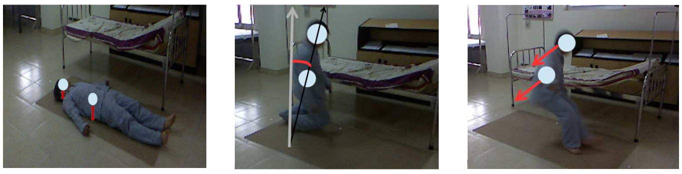


Fig. 5. Computation of distance, angle and velocity of each joint

E. Learning activities

To model activities and recognize them, we proposed to use Support Vector Machine technique.

As we observe, an activity does not happen at a frame, but in duration of time. In addition, the duration of an activity can vary from one person to another. Therefore, for each activity, we will keep only one value of each considered joint of human skeleton: maximal velocity, minimal percentage of distance w.r.t the maximal distance, maximal angle. The size of descriptor vector equals to the number of joints in each considered mode.

As we work with 5 modes, we will learn 5 SVM classifiers.

IV. EXPERIMENTAL RESULTS

A. Dataset

To evaluate our proposed method, we must prepare dataset for training and testing. As we would like to detect fall event from normal activities of the person, we will define scenarios in which such that events appear. Based on studying state of the art on fall dataset, we propose to make 9 activities:

- 4 fall activities: fall back, fall front, fall right, fall left
- 2 fall like activities: sit and lie down the ground
- 3 daily life activities: walk, pick an object, sit down the bed

6 subjects aging from 20 to 35 years old are asked to make 9 scenarios twice in a room of size 3mx3m equipped with a bed (Fig. 6). If we mark 5 points on the ground, the trajectories to perform 9 scenarios are defined as in the Tab.2. Totally we have 9 x 6 x 2 = 108 video clips. 54 clips for training and the remaining half for testing, but each test clips is tested 3 times.

Fig. 6. Exprimental environment viewed from Kinect sensor and markers to perform scenarios.

TABLE II. SCENARIOS FOR ACTIVTIY DATASET BUIDLING

N^0	Activity	Description
1	FallFront	S–>A–>Fall in the direction D
2	FallBack	S–>A–>Fall before arriving at D in your back in the direction of A
3	FallRight	S–>A–>Fall right after A in the direction C
4	FallLeft	S–>A–>Fall before arriving at C in your back in the direction of A
5	Sit	S–>A–>C–>Sit down suddenly on the floor
6	Lie	S–>A–>C–>Lie down in the direction of A
7	Walk	S–>A–>C–>A and stay at A position
8	Pick	S–>A Move to pick up an object, rise and stay at A
9	Sit	S–>A–>B–> Sit down and stay

B. Performance measurements

To evaluate the performance of our proposed method, we use two measurements: false positive rate (FP) and false negative rate (FN). Lower FP and FN are, better the system is.

C. Results

We will analyze the results of fall detection with the variation of types of features (distance, angle, and velocity), floor plane detection methods and skeletal modes (the number of used joints).

1) First experimentation: Analyze performance of three features with different modes and different floor plane detection methods

First, we prepare data for training and testing as presented above. Tab. 3 shows the obtained results with SVM technique. We notice that in all cases we do not obtain good values of false negative and false positive rates at the same time. In addition, as the angle value is quite unstable, the performance without using angle information is generally better. Beside, as the computation of floor plane using V-disparity method is not stable, the obtained result is worse than using Kinect SDK tools.

TABLE III. FIRST RESULTS OF ACTIVITY CLASSIFICATION (D: DISTANCE, V: VELOCITY, A: ANGLE)

Mode	***Rate (%)***	**Floor detection method**			
		Kinect		**V-Disparity**	
		{D, V, A}	***{D,V}***	***{D, V, A}***	***{D,V}***
1	FN	20	**10**	31	18
	FP	**21**	26	24	30
2	FN	18.06	**8.88**	31.11	17.78
	FP	**13**	24	24	28
4	FN	45	**26**	56.67	42.22
	FP	**0**	13.33	4.44	16.66
6	FN	45.56	**14.44**	51.11	16.67
	FP	**1.1**	4.4	5.6	7.8
8	FN	46.67	**9.44**	55.56	18.33
	FP	**1.11**	3.33	4.44	3.33

2) Second exprimentation: Modify the training dataset

To understand why SVM does not give satisfying results, we look into the distribution of training examples of each feature type (Fig. 7). We can observe that for angle feature, the separation of negative and positive samples is very difficult. In addition, distance and velocity features are more separated, mostly at mode 2 and 4.

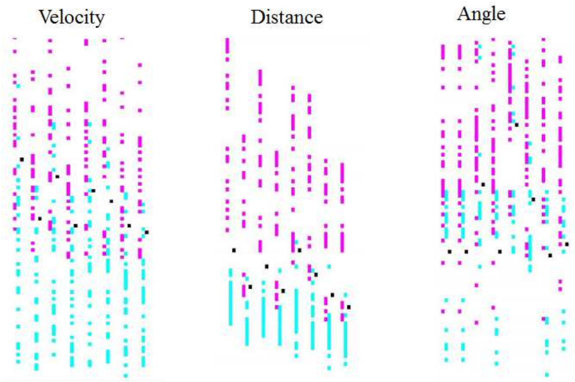


Fig. 7. Distribution of trainning examples for each feature type (left: velocity, middle: distance, right: angle) at each joint (from left to right in each column: Head, Shoulder Center, Spine, Hip Center, Shoulder Right, Shoulder Left, Hip Right, Hip Left) (best viewed in color). Magenta: negative samples (non-fall activities), Cyan: positive samples (fall activities). Black points represent the thresholds for classification.

Fig. 8. Removing some fall-like examples gives better separation on trainning set (with velocity feature).

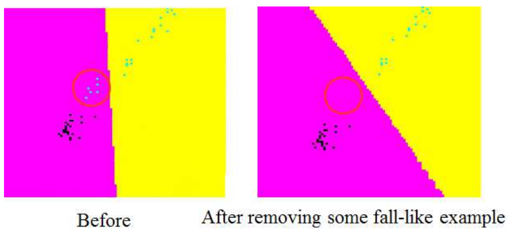


Fig. 9. Removing some fall-like example give better separation on trainning set (with distance feature). Points inside the cicle represent activity "sit suddenly on the floor".

Therefore, in the second experimentation, we will analyze two best modes (mode 2 and mode 4) using SVM technique for learning and SDK Kinect for floor plane detection. This time, we will look more detail into SVM representation at mode 2 for example. The separation line between positive set and negative set is not correct because of some fall-like samples (see Fig. 8, Fig. 9). We then remove them and re-learn SVM classifiers. The separation on the training dataset is now better. We will see if this gives better results on the test data.

TABLE IV. RESULTS AFTER RETRAINING SVM CLASSIFIERS

	Mode (using only Distance and Velocity feature)			
	2		4	
Rate (%)	FN	FP	FN	FP
	0.0	**18.0**	0.0	41.11

We can observe that with the new training there is no false negative, so it is very interesting result. But the false positive rate is still quite high (Tab. 4). Look into test results, we found that the activity "Lie down the floor" is often classified as fall activity.

3) Third exprimentation: Analyse variation of distance and velocity features

As the activity "Lie down the floor" is frequently classified as fall activity, we will analyze in more detail the characteristics of these two activities in order to improve the performance.

We found that there is a strong variation on the velocity and distance in both activities ("Lie down the floor" and "Fall") (Fig. 10). However, the transition time in case of "Lie down the floor" is longer. Therefore, we propose to take this observation to avoid the false positives.

Finally, we test the mode 2 and mode 4, using 2 types of features (distance and velocity) and with the time duration for fall is limited to a threshold (0.7s in our experiment) we obtain a very good result.

TABLE V. RESULTS WITH TIME DURATION OF ACTIVITY IS TAKEN INTO ACCOUNT

	Mode (using only Distance and Velocity feature)			
	2		4	
Rate (%)	FN	FP	FN	FP
	0.0	**3.3**	0.0	27.7

We found that this time no false negative is found and there are only 3.3% false positives at mode 2 (Tab. 5).

V. CONCLUSIONS

In this paper, we have presented a method for human fall detection based on skeleton information. An analysis on different modes (combination of skeleton joints), different feature types (angle, distance, velocity) has shown that using mode 2 (combination of 2 joints: head and spine) with two types of features that are distance and velocity give the best results. Angle feature is very sensitive so it is not recommended to be used. In addition, to avoid false positive, almost to distinguish fall-likes activity ("Lie down the floor") we should take transition time into account. The experimental results have been done with a dataset of 9 activities performed by different nationalities subjects. Compared to the works in [4], we worked with more challenge dataset. [4] uses only thresholding technique to detect fall so it is difficult to generize with another dataset.

In the future, we will test and compare our method with more data and deploy it in a system to help elderly people.

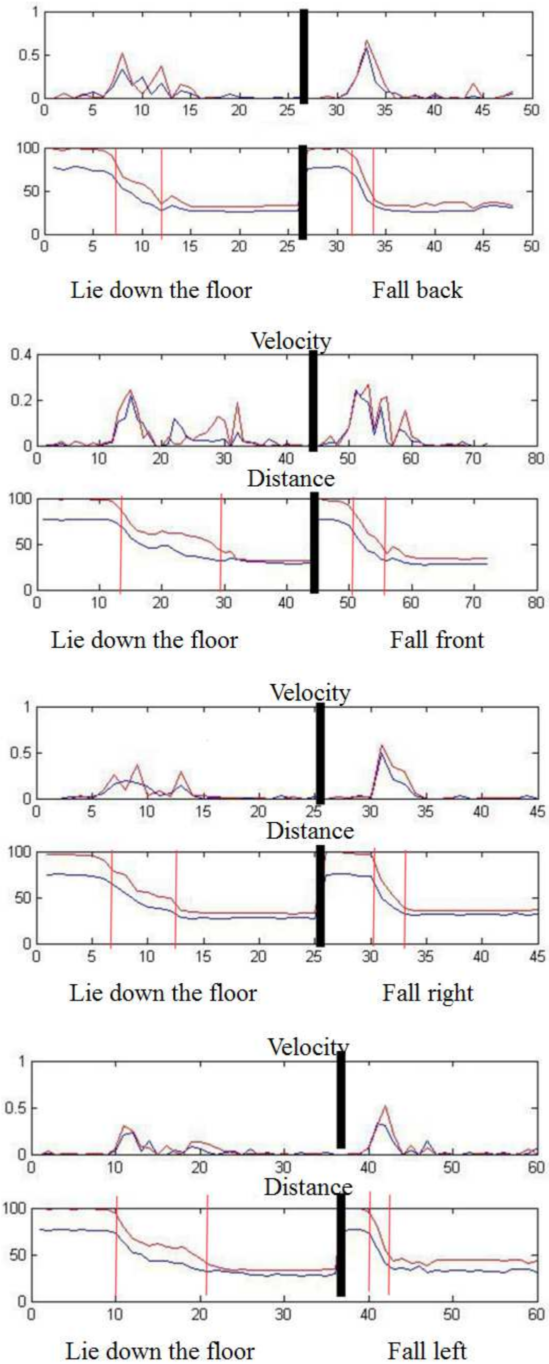


Fig. 10. Analyse the variation of velocity, distance of two joints (Head, Spine respectively in red and blue color) during time of two activities: “Lie down the floor” and “Fall”.

ACKNOWLEDGEMENTS

This work was supported by the Research Grant from Vietnam's National Foundation for Science and Technology Development (NAFOSTED), No. FWO.102.2013.08 and the National Project B2013.01.41.

REFERENCES

[1] Donald, I. and C. Bulpitt, The prognosis of falls in elderly people living at home. Age Ageing, 1999. 28(2).

[2] Deandrea, S., et al., Risk factors for falls in community-dwelling older people: a systematic review and meta-analysis. Epidemiology 2010. 21(5): p. 658–668.

[3] N, N., et al., A proposal for the classification and evaluation of fall detectors. Biomed Eng Res (IRBM) 2008. 29(6): p.:340–349.

[4] Rougier, C., et al., Fall Detection from Depth Map Video Sequences in ICOST 2011. p. 121-128.

[5] Kepski, M. and B. Kwolek, Fall Detection on Embedded Platform Using Kinect and Wireless Accelerometer, in ICCHP2012.

[6] Mastorakis, G. and D. Makris, Fall detection system using Kinect’s infrared sensor. Journal of Real-Time Image Processing, 2012.

[7] Kawatsu, C., J. Li, and C. Chung, Development of a Fall Detection System with Microsoft Kinect, in Robot Intelligence Technology and Applications 2012