Watchful-Eye: A 3D Skeleton-Based System for Fall Detection of Physically-Disabled Cane Users

Mona Saleh Alzahrani[1,2](✉), Salma Kammoun Jarraya[2], Manar Salamah Ali[2], and Hanêne Ben-Abdallah[2]

[1] College of Information and Computer Science, Al Jouf University, Sakaka, Saudi Arabia
mszahrani@ju.edu.sa

[2] Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia
{smohamadl,mali,hbenabdallah}@kau.edu.sa

Abstract. In this paper, we present *Watchful-Eye*, a 3D skeleton-based system to monitor a physically disabled person using a cane as a mobility aid. Watchful-Eye detects fall occurrences using skeleton tracking with a Microsoft Kinect camera. Compared to existing systems, it has the merit of detecting various types of fall under multiple scenarios and postures, while using a small set of features extracted from Kinect captured video streams. To achieve this merit, we followed the typical machine learning process: First, we collected a rich fall detection dataset. Second, we experimentally determined the most relevant features that best-distinguish fall from non-fall frames, and the best performing classifier. As we report in this paper, the offline evaluation results show that Watchful-Eye reached an accuracy between 87.2% and 94.5% with 5.5% to 12.8% error rate depending on the used classifier. Furthermore, the online evaluation shows that it can detect falls with an accuracy between 89.47% and 100%.

Keywords: Physically disabled · Computer vision · Machine learning
Fall detection · Skeleton tracking · Features selection · Kinect

1 Introduction

Computer vision research is actively contributing in building smart applications by providing for image/video content "understanding". Among the various application domains, this paper is interested in the contribution of computer vision to the development of applications to provide care for physically disabled people. More specifically, this paper proposes a computer vision based system to detect falls of physically disable people using canes as a means of assistance. Such a system is vital given the fall consequences on these people and the fall occurrence rates.

P. Perego et al. (Eds.): MobiHealth 2017, LNICST 247, pp. 107–116, 2018.
https://doi.org/10.1007/978-3-319-98551-0_13

Indeed, according to statistics from the World Health Organization (WHO)[1] in 2016, one out of three 65-year-old people falls each year and, as age increases to 80, the fall occurs each year. Furthermore, falls constitute the second leading cause of accidental or injury deaths after injuries of road traffic. These statistics call for efficient and practical/comfortable means to monitor physically disabled people in order to detect falls and react urgently.

In fact, several researches have proposed systems and/or methods for fall detection using computer vision techniques [1–4]. Our investigation of the recent systems showed that the most of them use the Microsoft Kinect camera, and some of them [1, 5, 6] also use smart sensors mounted on the person in an effort to increase the fall detection rate. In addition, these systems do not cover all types nor scenarios of fall, and they may depend on the distance of the person from the camera. Furthermore, our investigation highlighted the need for a benchmark dataset to assist in the development of new fall detection methods and the comparison of existing ones. As such, this paper has a two-fold objective. First, it proposes a dataset that contains data covering a large spectrum of fall types and scenarios. Secondly, it proposes a new system called Watchful-Eye for fall detection of physically disable people using canes. Compared to existing systems, it has the merit of detecting various types of fall under multiple scenarios and postures, while using a small set of features extracted from Kinect captured video streams. To achieve this merit, we followed the typical machine learning process: First, we collected a rich fall detection dataset. Second, we experimentally determined seven most relevant features that best-distinguish fall from non-fall frames, and RandomForest as the best performing classifier. As we report in this paper, the offline evaluation results show that Watchful-Eye reached an accuracy between 87.2% and 94.5% with 5.5% to 12.8% error rate, depending on the used classifier. Furthermore, the online evaluation shows that it can detect falls with an accuracy between 89.47% and 100%.

The remainder of this paper is organized as follows: Sect. 2 overviews the Kinect-based literature studies. Section 3 presents the proposed Watchful-Eye system and details its building steps—dataset collection, features' extraction and selection, and system setting and development. Section 4 discusses the experimental evaluation results, and Sect. 5 summarizes the presented work and outlines its extensions.

2 Kinect-Based Fall Detection

Several recent studies [1–3], classified under vision-based approaches, use Kinect for developing Fall Detection (FD) systems. In 2013, Lee and Lee [2] present a system to detect falls and notify health care services. They use the Kinect depth camera as the input sensor and Microsoft Kinect SDK to collect skeleton data. Among the collected skeleton data, they chose to track only the hip joins which they process with two functions that check the position and velocity of the center of mass. They achieve a 90% accuracy FD rate. Nonetheless, they have a list of false positive postures such as

[1] http://www.who.int/mediacentre/factsheets/fs344/en/.

sitting on the floor with both legs folded behind, kneeling on the floor, squatting, bending down to wear shoes or tie shoelaces. In addition, they cannot detect when the user falls off a chair, which is relatively a very common scenario.

In 2014, in an effort to reduce the number of false alarms by collecting more information, Kwolek and Kepski [1] add to the Kinect a wearable smart device containing accelerometer and gyroscope sensors; this smart device is worn near the pelvis region of the monitored person. They use a triaxial accelerometer to indicate both a potential fall and whether the person is in motion. Their proposed system operates as follows: If the measured acceleration is higher than an assumed threshold value, the system extracts the person on the basis of the depth reference maps, calculates some depth features, and executes the SVM-based classifier to authenticate the fall alarm. This system acquires depth images using the OpenNI (Open Natural Interaction) library. It achieves 98.33% accuracy when using accelerometer and depth data, and 90% accuracy and 80% specificity when using depth only which is the worst result compared to other techniques in their research.

In 2015, Stone and Skubic [3] develop a two-stage FD system for detecting falls in the homes of older adults using the Microsoft Kinect. The first stage characterizes the vertical state of a 3D object for an individual frame, it then segments on ground events from the vertical state time series. The second stage utilizes a set of decision trees and a set of features extracted from an on-ground event to generate a confidence that a fall preceded it. As a preprocessing step, this system segments 3D foreground objects from each depth frame using dynamic background subtraction. When the falls are near the sensor and not significantly occluded, this system can achieve 98%, 70%, and 71% accurate detection of standing, sitting, and lying falls, respectively; however, when the falls are far to the sensor and significantly occluded, the system can achieve 79%, 58%, and 5% accurate detection of standing, sitting, and lying falls, respectively.

Overall, existing fall detection systems using Kinect [1–3] differ in their performance: Some do not cover many fall types and/or scenarios; others have high false alarm rates when operating on particular postures; yet others have low accuracy when the faller is far from the Kinect. In addition, those trying to improve the fall detection rate use wearable sensors, which may hinder daily activities and/or make the person uncomfortable. Furthermore, the proposed systems' performance evidently depends on the features used. However, the feature differences (in nature and number) and the lack of a benchmark dataset hinders a systematic evaluation of the performance of existing systems.

As such, the aim of this paper is to propose a fall detection method that: suits physically disabled people using canes, relies solely on Kinect, and can determine various types of falls in different postures with a high accuracy and a low false alarm. Such a method highly depends on the selection of the appropriate features. Towards this end, this paper's second contribution is the elaboration of a dataset that can be used as a benchmark to both identify the features and compare existing/future methods.

3 Overview of the Skeleton-Based Fall Detection

The development of Watchful-Eye proposes three main contributions to the domain of fall detection of physically disabled people:

1. Proposition of a new fall dataset that covers all fall types and scenarios. The dataset is available in all the image-based streams provided by the Kinect camera.
2. Identification of the features best describing fall scenarios of cane users; besides accounting for all fall types and scenarios, the identified features overcome the challenges incurred by the distance between the Kinect and the faller.
3. Proposition of new skeleton-based method that: detects the different fall types and scenarios, imposes minimum restrictions on the people with physical disabilities (i.e. pose or calibration), overcomes the natural scene conditions (e.g., lighting), requires no prior knowledge about the rooms, is suitable for physical disabled people using canes, and minimizes the false alarm rate.

The above contributions are detailed in the next subsections, and the experimentally evaluated performance of the developed system is discussed in Sect. 4.

3.1 The Fall Detection Dataset

The current FD datasets [1–3] are not suitable for this study for the following reasons: either they targeted healthy people only, they did not provide skeleton data streams, did not cover most of the scenarios, or they are not made accessible. These reasons prompted us to record a new dataset that: is made especially for the physically disabled people using canes as mobility aid; provides all the Kinect image-based streams (RGB color, depth, skeleton, infrared and body index); covers almost all the fall scenarios suggested by Noury et al. [7]; is accessible by contacting the authors and will be available soon at web.

To include all fall types and scenarios in [1–3, 7], we prepared a large dataset that contains 392 videos. These videos include 208 fall videos that cover backward, forward, lateral fall to the right and to the left. In addition, they include 184 non-fall videos composed of 115 videos of pseudo fall situations and 69 videos of ADL (Activities of Daily Living). This dataset was recorded using Kinect v2 for two male subjects in a frame rate of 30 fps. In each testing room, Kinect was set at 1 m high from the floor.

A dataset with real fall cases would be much more valuable but it is actually impracticable to test the fall situations with physically disabled people. So, the subjects simulated the cane user's walking pattern introduced by Melis et al. in [8]. Sample images of the dataset are shown in Fig. 1.

3.2 Features Extraction and Selection

In order to prepare the learning data for the detection system, we first extracted, from all videos' frames, the 3D positions of the 25 joints obtained from Kinect v2, which represent the skeleton. Second, for each frame, we preprocessed the joints' positions in three different ways: (i) *Original* positions (W) without any preprocessing;

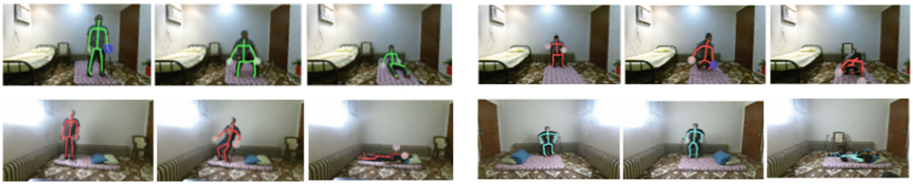

Fig. 1. Dataset samples.

(ii) *Translated* positions (T) by translating the mid spine joint to Kinect origin along with the other joints depending on it; and (iii) *Normalized* positions (N) using the torso-centered method [9]. These preprocessing ways overcome the difference of the faller size, distance and position. After each preprocessing method, we calculated three feature sets as follow:

1. Distances (D) of the joints from the Kinect [10]: 75 features (25 from the original skeleton, 25 from the translated skeleton, and 25 from the normalized skeleton).
2. Velocities (V) of the joints in the direction normal to the floor plane [4]: 75 features (25 from the original skeleton, 25 from the translated skeleton, and 25 from the normalized skeleton).
3. Angles (A) [11]: 45 features (15 from the original skeleton, 15 from the translated skeleton, and 15 from the normalized skeleton).

Because of the large number of features (195 features), we conducted three selection trials to eliminate irrelevant features, using two filter methods (Relief-F and Information Gain) and wrapper methods using two classifiers (C4.5 and IBk). From the union of all the features resulting from these methods, we took the most relevant features that gave us the best results.

In each of the three trials, we changed the way of classifying the fall frames. In Trial 1, the fall frame was *any* frame belonging to a fall video. In Trial 2, the fall frames were divided into two classes as indicated in [1]: (i) *Temporary-pose* frames when the faller starts falling until s/he reaches the floor; and (ii) *Fall* frames when the faller hits the floor, and stays on it. Finally, in Trial 3, the fall frame was the temporary pose and the fall frames from Trial 2. Tables 1, 2 and 3 summarize these three trials. Figures 2, 3 and 4, respectively, show the most relevant features obtained from the three trials.

Furthermore, to identify the most appropriate/performing features, we evaluated the results of each trial using a C4.5 classifier to measure its performance. As seen in Table 4, the best results were obtained from the seven relevant features in Trial 3 with a 91.53% accuracy. These features belong to the upper body part, which makes sense because this part is the main part used to support the cane before a fall happens.

3.3 Proposed Skeleton-Based Fall Detection Method and Its Setting

The conceptual architecture of the Watchful-Eye system receives the skeleton stream captured through the Kinect and transferred through USB to a laptop running the FD method. This latter first extracts the skeleton and normalizes a copy of it. Afterward, from the original data, it extracts features: right shoulder distance, right hand and right

Table 1. The conducted Trial 1 of the feature selection experiment.

Training data	4960 frames; 2480 for each class	
Classes	• 1 (Fall): if the frame from fall video classifies as fall • 0 (Non-fall): if the frame from non-fall video classifies as non-fall	See Fig. 2
Best features	**5 features calculated from 7 joints:** 1. Original angle of (left hip, base spine, right hip) 2. Original head velocity 3. Translated left ankle distance 4. Translated angle of (shoulder spine, mid spine, base spine) 5. Normalized mid spine distance	
Accuracy	77.29%	

Table 2. The conducted Trial 2 of the feature selection experiment.

Training data	4959 frames; 1653 for each class	
Classes	• 2 (Temporary pose): from fall video, the frame when the person starts fallen classify as a temporary pose • 1 (Fall): from fall video, the frame after the person are fallen and laying in the floor classify as fall • 0 (Non-fall): if the frame from non-fall video classifies as non-fall	See Fig. 3
Best features	**4 features calculated from 3 joints:** 1. Original left shoulder distance 2. Original head velocity 3. Normalized left shoulder distance 4. Normalized left thumb distance	
Accuracy	74.8%	

Table 3. The conducted Trial 3 of the feature selection experiment.

Training data	4960 frames; 2480 for each class	
Classes	• 1 (Fall): from fall video, the frame when the person starts fallen until he fallen and laying in the floor classify as fall • 0 (Non-fall): if the frame from non-fall video classifies as non-fall	See Fig. 4
Best features	**7 features calculated from 6 joints:** 1. Original right shoulder distance 2. Original right hand velocity 3. Original right thumb velocity 4. Normalized left hand distance 5. Normalized left shoulder distance 6. Normalized left thumb distance 7. Normalized right thumb distance	
Accuracy	91.53%	

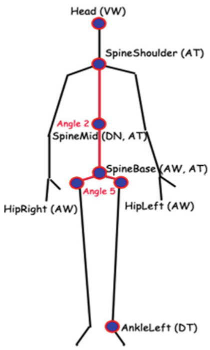

Fig. 2. Mapping of the best features representing the fall from Trial 1.

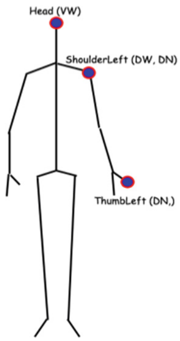

Fig. 3. Mapping of the best features representing the fall from Trial 2.

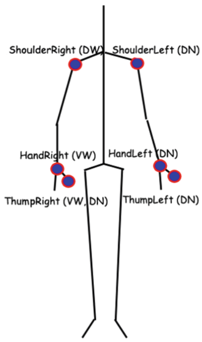

Fig. 4. Mapping of the best features representing the fall from Trial 3.

thumb velocities. From the normalized data, it extracts the rest features: left hand, left shoulder, left thumb and right thumb distances (features retained from Trial 3) and it feeds them to the appropriate classifier. Finally, based on the frames' classification results, it decides whether there is a fall, in which case it triggers an alarm. To

Table 4. The online evaluation results.

Fall type	Video number	Weaker side	F	TP	FN	SE
Backward fall	1	Left	79	75	4	94.9%
	2	Any	63	57	6	90.5%
Forward fall	3	Right	49	49	0	100%
	4	Any	31	31	0	100%
Lateral fall to the right	5	Left	76	68	8	89.47%
	6	Any	49	49	0	100%
Lateral fall to the left	7	Right	77	72	5	93.5%
	8	Any	52	49	3	94.23%

determine the appropriate classifier, we conducted a set of experiments whose results we discuss in the next section.

During the development of Watchful-Eye, we supposed that the system is to operate under the following settings/hypotheses:

1. It is set to monitor a room containing the disabled person in real-time;
2. It uses Kinect v2 for Windows with the free tool of Kinect SDK to detect and track the body/skeleton of the disabled person;
3. The Kinect sensor is placed 1 m high from the floor, as we did in the dataset. Its field of view should be able to cover both the room and the monitored person. In addition, the depth range of Kinect, which could reach 4.5 m [4] is also considered in the detection process;
4. The tracked joints are the six extracted from Trial 3 of the feature selection experiment.

4 Experimental Results

In this section, we analyze the detection performance of Watchful-Eye through an offline and online experimental evaluations. The two subjects from the recorded dataset were engaged in both experiments.

Experiment 1: Offline Evaluation. This first experiment aims to identify the appropriate classifier. Towards this end, we prepared a sample of 4960 frames from the recorded data with 2480 fall frames and 2480 non-fall frames. The frames used during the offline experiment correspond to the whole range of the captured 392 videos. We used 70% of this data (3472 frames) for training and 30% (1488 frames) for testing.

To determine the best classifier to build the classification model, we tested the data using different classification algorithms: C4.5, Logistic Model Trees (LMT), RandomForest, RandomTree, REPTree, and Instance-Based k (IBk) used with their default parameters as suggested by [12]. Based on the obtained results, we concluded that RandomForest is the best classifier (**Accuracy = 94.5%**, **Sensitivity = 92.8%**, **Specificity = 96%** and **AUC** (Area Under the ROC Curve) = **0.9858**) to use in Watchful-Eye and the online experimental evaluation.

Experiment 2: Online Evaluation. For the online evaluation, we developed the Watchful-Eye program that we operated with real-time videos of the two subjects as captured directly from Kinect. In this experiment, we used eight different fall videos (V) representing the following types of falls:

1. Backward fall from standing ending lying.
2. Backward fall from sitting on chair with no back ending lying.
3. Forward fall from standing ending on the knees.
4. Forward fall from sitting with forwarding arm protection.
5. Lateral fall to the right from standing ending lying flat.
6. Lateral fall to the right from lying on bed.
7. Lateral fall to the left from standing ending lying flat.
8. Lateral fall to the left from lying on bed.

Table 4 shows the experimental results where F and SE are the Total Video Frames, and the Sensitivity, respectively. In addition, "Weaker side" represents the weaker side of the physically disabled person. If s/he has left weaker side, then s/he was holding the cane by his other stronger side (right) as explained by Melis et al. in [8]. The videos with (*Any*) weaker side, that means the subject did not need to hold a cane because her/his postures (sitting or lying), and this situation could happen to the physically disabled person with *any* (left or right) weaker side.

From these experimental results, we notice that the offline evaluation results show that Watchful-Eye reached an accuracy between 87.2% and 94.5% with 5.5% to 12.8% error rate depending on the used classifier. Furthermore, the online evaluation shows that it can detect falls with an accuracy between 89.47% and 100%.

5 Conclusion

In this work, we presented Watchful-Eye, a skeleton-based monitoring system to monitor a physically disabled person using a cane and to detect fall occurrences. To develop such system, we constructed a dataset that can serve as a benchmark for evaluating and/or developing fall detection methods. This dataset has the merit of using the latest Kinect version, containing rich collected data, and covering a large spectrum of fall types and scenarios. In addition, we experimentally identified seven relevant features and appropriate classifier (RandomForest) classify frames into fall or non-fall. Finally, we experimentally showed that thus-developed system offers accuracy between 87.2% and 94.5% with 5.5% to 12.8% in offline evaluation, while in online evaluation shows that it can detect falls with an accuracy between 89.47% and 100%. In our future works, we will focus on improving Watchful-Eye by training it to detect falls from frame sequences in order to increase its accuracy and reduce its false alarm rates.

References

1. Kwolek, B., Kepski, M.: Human fall detection on embedded platform using depth maps and wireless accelerometer. Comput. Methods Programs Biomed. **117**, 489–501 (2014)
2. Lee, C.K., Lee, V.Y.: Fall detection system based on Kinect sensor using novel detection and posture recognition algorithm. In: Biswas, J., Kobayashi, H., Wong, L., Abdulrazak, B., Mokhtari, M. (eds.) ICOST 2013. LNCS, vol. 7910, pp. 238–244. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39470-6_30
3. Stone, E.E., Skubic, M.: Fall detection in homes of older adults using the Microsoft Kinect. IEEE J. Biomed. Health Inf. **19**, 290–301 (2015)
4. Kawatsu, C., Li, J., Chung, C.: Development of a fall detection system with Microsoft Kinect. In: Kim, J.H., Matson, E., Myung, H., Xu, P. (eds.) Robot Intelligence Technology and Applications 2012. AISC, vol. 208, pp. 623–630. Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37374-9_59
5. Kozina, S., Gjoreski, H., Gams, M., Luštrek, M.: Efficient activity recognition and fall detection using accelerometers. In: B, Juan A., Álvarez-García, J.A., Fujinami, K., Barsocchi, P., Riedel, T. (eds.) EvAAL 2013. CCIS, vol. 386, pp. 13–23. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41043-7_2
6. Abdali-Mohammadi, F., Rashidpour, M., Fathi, A.: Fall detection using adaptive neuro-fuzzy inference system. Int. J. Multimed. Ubiquit. Eng. **11**, 91–106 (2016)
7. Noury, N., Fleury, A., Rumeau, P., Bourke, A., Laighin, G., Rialle, V., et al.: Fall detection-principles and methods. In: 2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 1663–1666 (2007)
8. Melis, E., Torres-Moreno, R., Barbeau, H., Lemaire, E.: Analysis of assisted-gait characteristics in persons with incomplete spinal cord injury. Spinal Cord **37**, 430–439 (1999)
9. Rhemyst and Rymix. Kinect SDK Dynamic Time Warping (DTW) Gesture Recognition, 30 July 2011–2 January 2017. http://kinectdtw.codeplex.com/
10. Pterneas, V.: Measuring Distances using Kinect – The Right Way, 29 October 2016. http://pterneas.com/2016/08/11/measuring-distances-kinect/
11. Le, T.-L., Nguyen, M.-Q.: Human posture recognition using human skeleton provided by Kinect. In: 2013 International Conference on Computing, Management and Telecommunications (ComManTel), pp. 340–345 (2013)
12. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. ACM SIGKDD Explor. Newsl. **11**, 10–18 (2009)