applied sciences

Article

Fall Detection for Elderly from Partially Observed Depth-Map Video Sequences Based on View-Invariant Human Activity Representation

Rami Alazrai *, Mohammad Momani and Mohammad I. Daoud

School of Electrical Engineering and Information Technology, German Jordanian University, Amman 11180, Jordan; m.momani@gju.edu.jo (M.M.); mohammad.aldaoud@gju.edu.jo (M.I.D.)
* Correspondence: rami.azrai@gju.edu.jo; Tel.: +962-798-213-151

Academic Editors: Plamen Angelov and José Antonio Iglesias Martínez
Received: 1 February 2017; Accepted: 22 March 2017; Published: 24 March 2017

Abstract: This paper presents a new approach for fall detection from partially-observed depth-map video sequences. The proposed approach utilizes the 3D skeletal joint positions obtained from the Microsoft Kinect sensor to build a view-invariant descriptor for human activity representation, called the motion-pose geometric descriptor (MPGD). Furthermore, we have developed a histogram-based representation (HBR) based on the MPGD to construct a length-independent representation of the observed video subsequences. Using the constructed HBR, we formulate the fall detection problem as a posterior-maximization problem in which the posteriori probability for each observed video subsequence is estimated using a multi-class SVM (support vector machine) classifier. Then, we combine the computed posteriori probabilities from all of the observed subsequences to obtain an overall class posteriori probability of the entire partially-observed depth-map video sequence. To evaluate the performance of the proposed approach, we have utilized the Kinect sensor to record a dataset of depth-map video sequences that simulates four fall-related activities of elderly people, including: walking, sitting, falling form standing and falling from sitting. Then, using the collected dataset, we have developed three evaluation scenarios based on the number of unobserved video subsequences in the testing videos, including: fully-observed video sequence scenario, single unobserved video subsequence of random lengths scenarios and two unobserved video subsequences of random lengths scenarios. Experimental results show that the proposed approach achieved an average recognition accuracy of 93.6%, 77.6% and 65.1%, in recognizing the activities during the first, second and third evaluation scenario, respectively. These results demonstrate the feasibility of the proposed approach to detect falls from partially-observed videos.

Keywords: fall detection; partially-observed videos; fall prediction; view-invariant geometric descriptor; support vector machines; human representation; Kinect sensor

1. Introduction

The growing population of elderly people is becoming a pressing issue worldwide, especially in developed countries. In Europe, the official projections indicate that the elderly population is expected to grow rapidly by 58 million between 2004 and 2050 [1]. In the United States, 20% of the population is expected to be elderly by 2030, while in 1994, the percentage of elderly people was only one in eight Americans [2,3]. In fact, falls are among the major threats to the health of elderly people, particularly those who live by themselves. According to [4], the percentage of elderly people who fall each year is more than 33%. Falls can lead to both physiological and psychological problems for elderly people [5]. Moreover, in the case of falls that do not lead to immediate injuries, around one half of the non-injured elderly fallers require assistance to get up, and hence, delayed assistance can result in long immobile periods, which might affect the faller's health [5,6]. Therefore, accurate and efficient fall detection systems are crucial to improve the safety of elderly people.

Recently, several fall detection systems have been proposed to detect fall incidents. Among these systems, wearable and non-wearable sensor-based systems are the most commonly used [7]. Wearable sensor-based systems employ sensors that are attached to the subject to continuously monitor the subject's activities. For example, several fall detection systems utilize wearable devices that are equipped with gyroscope and accelerometer sensors in order to identify fall incidents [8–10]. One limitation of wearable sensor-based fall detection systems is the requirement of wearing sensing devices all of the time. Such a requirement might be impractical and inconvenient, especially that wearable sensing devices need to be recharged on a regular basis. The second group of commonly-used fall detection systems is based on using non-wearable sensing devices, such as vision and motion tracking systems, to identify human activities [11–13]. RGB video cameras (2D cameras) and motion-capturing systems are among the most widely-used non-wearable sensing devices for fall detection [14,15]. Fall detection systems that are based on analyzing RGB videos recorded by 2D cameras are highly affected by several factors, such as occlusions, illumination, complex background and camera view-angle. Such factors can reduce the accuracy of fall detection. Utilizing motion-capturing systems for fall detection can be a remedy for the aforementioned factors. However, the high cost of the motion-capturing systems and the need to mount markers on the subjects to track their motions might limit the practical application of motion-capturing systems for fall detection.

Recently, Microsoft (Microsoft Corporation, Redmond, WA, USA) has introduced a low-cost RGB-D sensor, called the Kinect sensor, which comprises both an RGB camera and a depth sensor. The depth sensor of Kinect enables the recording of depth-map video sequences that preserve the privacy of the subjects. This privacy feature makes the Kinect sensor a non-intrusive sensor compared with the 2D cameras that reveal the identity of the subjects. Furthermore, the capability of the Kinect sensor to detect and track the 3D positions of skeletal joints has attracted several researchers to utilize the Kinect sensor to track human activities and detect falls by analyzing depth-map videos.

The literature reveals that the majority of fall detection systems that employ depth-map video sequences are focused on detecting falls by analyzing the frames of a complete video sequence that covers the entire fall incident [16–25], as illustrated in Figure 1a. Nonetheless, in real-life scenarios, the fall incident might be partially observed due to the presence of an occlusion that blocks the view of the camera or occludes the subject of interest. Moreover, power disconnections may result in missing some video segments, which in turns can lead to partially-observed videos. Fall detection based on partially-observed videos is considered challenging due to the fact that the unobserved subsequences of video frames might be of different lengths and can occur at any time during video recording. To address this challenge, researchers have recently attempted to predict falls using incomplete depth-map video sequences [26,27]. Specifically, fall prediction aims to predict the falling event by analyzing an incomplete depth-map video sequence in which a subset of frames that covers only the beginning of the fall is observed, as depicted in Figure 1b. In fact, fall prediction can be viewed as a special case of the general scenario of fall detection based on partially-observed videos, in which the fall is identified by analyzing a video sequence that includes unobserved subsequences of video frames having different durations and occurring at any time. Figure 1c illustrates the general scenario of fall detection based on partially-observed videos.

Recently, several approaches have been proposed to detect falls under occluded conditions based on analyzing the depth images acquired using the Kinect sensor. For example, Gasparrini et al. [23] proposed a Kinect-based approach for detecting falls in indoor environments. The approach analyzes the depth images acquired from the Kinect sensor using anthropometric relationships and features to identify depth blobs that represent humans. Then, a fall is detected if the depth blob associated with a human is near the floor. Stone and Skubic [28] proposed a two-stage fall detection approach to identify three fall-related positions, including: standing, sitting and lying down. The first stage of their approach analyzes the depth images acquired from a Kinect sensor to identify the vertical characteristics of each person and identify on-ground events based on the computed vertical state information of the subject over time. Then, the second stage employs an ensemble of decision trees classifier to identify falls in the on-ground events. Rougier et al. [24] proposed a Kinect-based system to detect falls under occlusion conditions. Their system employs two features for fall detection, including the subject's centroid to measure the distance from the floor, as well as the body velocity. A fall is detected when the velocity is larger than a certain threshold, while the distance from the ground to the subject's centroid is smaller than a specific threshold value.

Unlike the aforementioned approaches, in which fall-related activities were recognized while the subject is partially occluded based on depth images, Cao et al. [29] proposed an approach for recognizing human activities from partially-observed videos based on sparse coding analysis. The reported experimental results in [29] show limited recognition accuracy, which might not be suitable for real-world applications. Moreover, the approach employed in [29] considered only RGB videos that include a single unobserved subsequence of video frames. As discussed previously, the use of depth sensors, such as the Microsoft Kinect sensor, for fall detection provides several advantages over the RGB cameras, including subject privacy preservation and the capability to acquire the 3D positions of skeletal joints at interactive rates. To the best of our knowledge, the general scenario of fall detection from partially-observed depth-map video sequences based on utilizing the 3D skeletal joint positions has not been investigated in the literature.

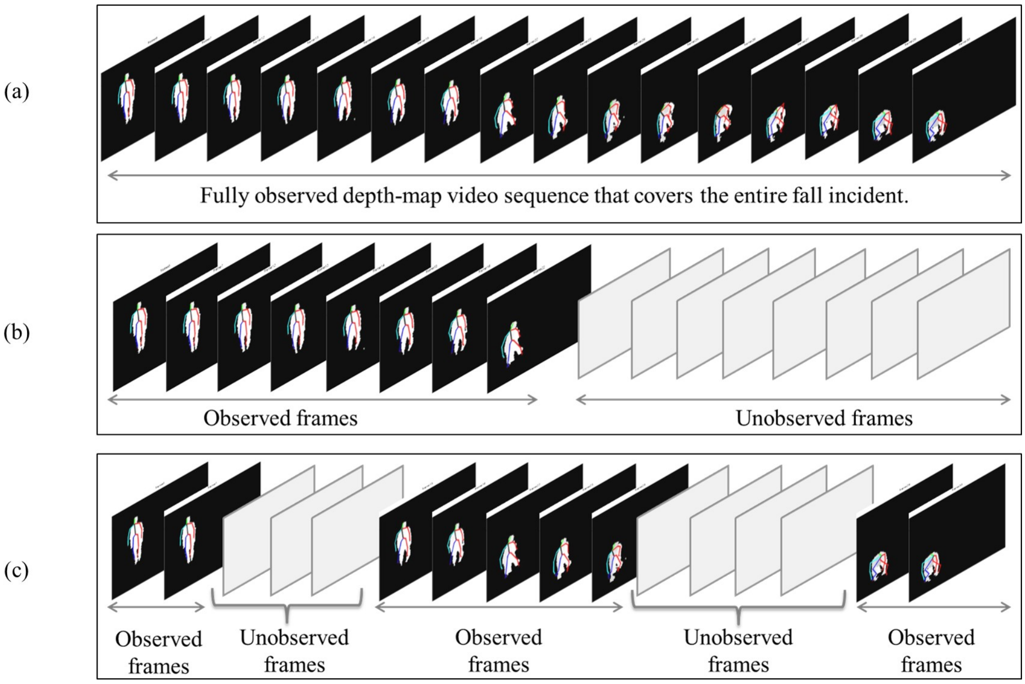

Figure 1. Illustration of the different fall detection schemes. (**a**) Fall detection using a complete depth-map video sequence in which the frames cover the entire fall incident [30]; (**b**) Fall prediction using incomplete depth-map video sequence in which a subset of frames that cover the beginning of the fall incident is observed, while the frames that cover the rest of the fall incident are not observed [27]; (**c**) Fall detection using a partially-observed depth-map video sequence in which the observed frames cover discontinuous parts of the fall incident.

In this paper, we propose an approach for detecting falls from partially-observed depth-map video sequences based on the 3D skeletal joint positions provided by the Kinect sensor. The proposed approach utilizes the anatomical planes concept [31] to construct a human representation that can capture both the poses and motions of the human body parts during fall-related activities. In particular, we adopt and expand our earlier work [30,32] in which a motion pose geometric descriptor (MPGD) was proposed for analyzing human-human interactions and elderly fall detection using complete depth-map video sequences. The MPGD consists of two profiles, namely motion and pose profiles. These profiles enable effective capturing of the semantic context of human activities being performed at each video frame. In this work, we expand the motion and pose profiles in the MPGD by introducing a new set of geometrical relational-based features, to better describe fall-related activities. In order to detect falls in partially-observed depth-map video sequences, we segment fully-observed training video sequences into overlapping segments. Then, we construct a histogram-based representation (HBR) of the MPGDs for the video frames of each segment. The HBR describes the distribution of the MPGDs within each video segment and captures the spatiotemporal configurations encapsulated within the activities of an elderly person during each video segment. Using the computed HBRs of MPGDs, we train a support vector machine (SVM) classifier with a probabilistic output [33] to predict the class of the performed activity in a given video segment. For any new video with unobserved frames subsequences, we compute the HBR of the MPGDs that are associated with video frames in each observed video subsequence. Then, for each observed video subsequence, we utilize the learned SVM model to compute the class posteriori probability of the observed subsequence given its HBR. Finally, to predict the class of the performed activity in the given partially-observed video, we combine the obtained posteriori probabilities from all observed subsequences to obtain an overall posteriori probability estimation for the partially-observed video.

In order to evaluate the proposed fall detection approach, we have utilized the Microsoft Kinect sensor to record a dataset of depth-map video sequences that simulates fall-related activities of elderly people. The recorded activities include: walking, sitting, falling form standing and falling from sitting. Three evaluation procedures are developed to quantify the performance of the proposed approach at various configurations, including: fully-observed video sequence, partially-observed video sequence that includes one unobserved video subsequence of random length and partially-observed video sequence that includes two unobserved video subsequences of random lengths. The experimental results reported in this study demonstrate the feasibility of employing the proposed approach for detecting falls based on partially-observed videos.

The remainder of this paper is organized as follows. In Section 2, we describe the collected dataset of fall-related activities, the modified MPGD and HBR for human activities and the proposed fall detection approach based on partially-observed depth-map videos. Section 3 presents the experimental results and discussion. Finally, conclusions are presented in Section 4.

2. Materials and Methods

2.1. Dataset

Six healthy subjects (1 female and 5 males) volunteered to participate in the experiments. The mean ± standard deviation age of the subjects was 33 ± 8.7 years. The experimental procedure was explained in detail for each subject. A signed consent form was collected from each subject before participating in the experiments. Furthermore, the participants had the chance to withdraw from the study at anytime during the experimental procedure. The experimental procedure was reviewed and approved by the Research Ethics Committee at the German Jordanian University. In this study, each subject was asked to simulate four activities related to the fall event. These activities are: walking, sitting, falling from standing and falling from sitting. The subjects performed the activities several times, and each time, the activities were carried out with various speeds and styles to capture the inter- and intra-personal variations between different subjects. This experiment was performed in a laboratory environment in which a mattress was placed on the ground to protect the subjects during falling. In order to record depth-map video sequences of the aforementioned four activities, a Kinect Sensor for XBOX 360 (Microsoft, Redmond, WA, USA) with the Kinect SDK v1.0 beta 2 (Microsoft, Redmond, WA, USA) was utilized.The collected dataset consists of approximately 19,300 frames that represent 229 activity sequences. In fact, different activities have various time durations depending on the subject and the type of the activity. The average length of the recorded activity sequences is approximately 84 frames. The activity sequences were captured at a rate of 15 frames per second (fps). The resolutions of the acquired RGB images and depth maps are 640 × 480 and 320 × 240, respectively. Simultaneous to the RGB images and depth maps, the Kinect sensor enabled the acquisition of the three-dimensional (3D) coordinates of twenty skeletal joints at a rate of 15 fps. The acquired sequences of RGB images, depth maps and 3D coordinates of the skeletal joints that correspond to each activity were manually annotated into multiple temporal segments, such that each segment represents a sub-activity. For example, the falling from standing activity was divided into three temporal segments. The first temporal segment represents the standing pose sub-activity. The second temporal segment represents the falling from standing pose sub-activity. Finally, the third temporal segment represents the fall pose sub-activity. The sub-activities associated with each of the four recorded activities are provided in Table 1. Figure 2 provides sample images of the different sub-activities associated with the fall-related activities.

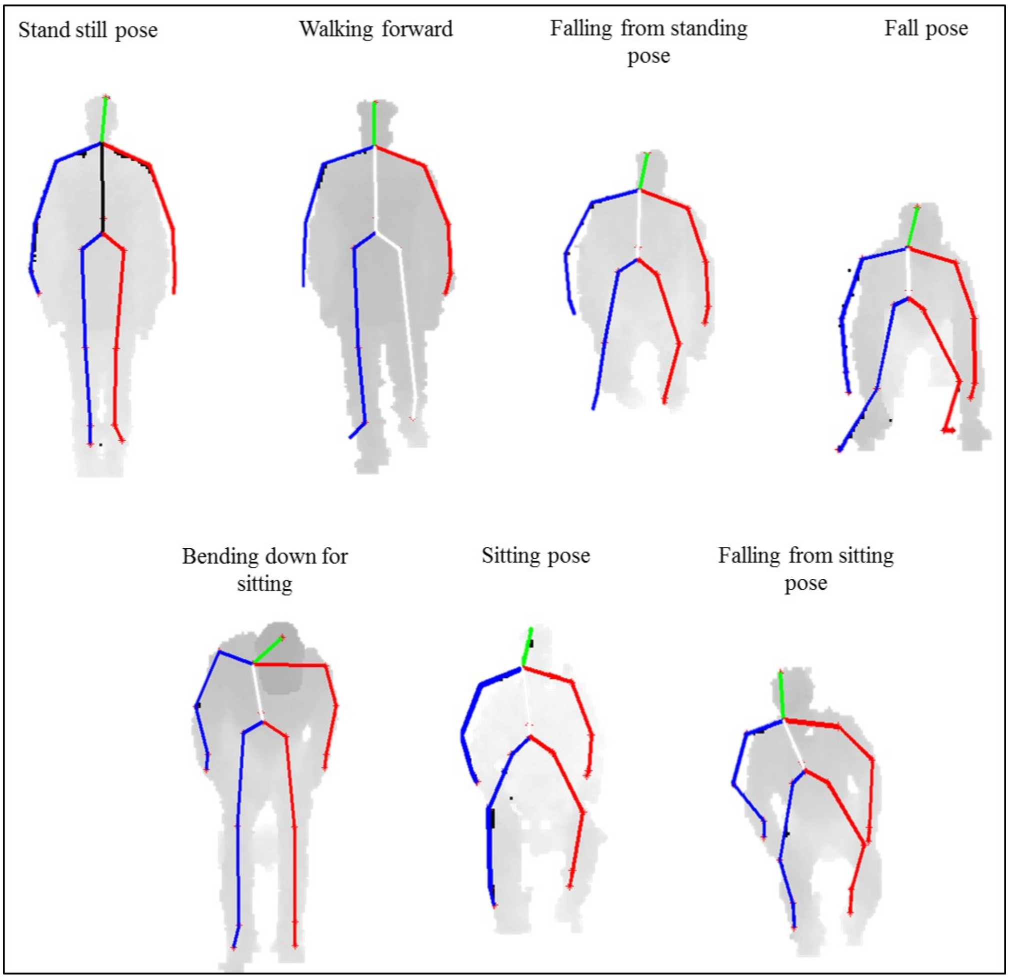

Figure 2. Sample images of different sub-activities associated with the four fall-related activities investigated in this study.

Table 1. The sub-activities encapsulated within each activity.

Activity	Sub-Activities
Walking	Walking forward (i.e., walking towards the Kinect sensor) and walking backward (i.e., walking away from the Kinect sensor).
Sitting	Stand still pose, bending down for sitting and sitting pose.
Falling from standing	Stand still pose, falling from standing pose and fall pose.
Falling from sitting	Sitting pose, falling from sitting pose and fall pose.

2.2. Human Activity Representation

Recently, Alazrai et al. [32] proposed a view-invariant motion-pose geometric descriptor (MPGD) that utilizes the anatomical planes concept [31] to model the interactions between two humans. The MPGD employs the Microsoft Kinect sensor to capture the activities of two interacting humans by analyzing the 3D locations of twenty skeletal joints of each human, including: hip center (hc), spine (sp), shoulder center (sc), head (hd), left hand (lhd), right hand (rhd), left wrist(lwt), right wrist (rwt), left elbow (lew), right elbow (rew), left shoulder (lsh), right shoulder (rsh), left hip (lhp), right hip (rhp), left knee (lke), right knee (rke), left ankle (lak), right ankle (rak), left foot (lft) and right foot (rft). Then, using the acquired skeletal joint positions, two profiles, namely motion and pose profiles, are constructed to describe the movements of the body parts of the two interacting humans. Finally, the MPGD is constructed by combining the motion and pose profiles. Using the two profiles of the MPGD, different spatiotemporal configurations that are associated with various activities can be captured at each video frame [32]. In this paper, we adopt and modify the MPGD to describe the four fall-related activities described in Section 2.1. Specifically, the MPGD is modified to represent the activities of a single human rather than two humans. In addition, the motion and pose profiles are expanded by introducing a set of geometrical relational-based features to better characterize the fall-related activities. In the next subsections, we provide a detailed description of the modified motion and pose profiles of the MPGD.

2.2.1. Motion Profile

The movements of different body parts during various human activities can be described with respect to three anatomical planes, namely the transverse plane (TP), coronal plane (CP) and sagittal plane (SP), intersecting at a specific point on the human body [31]. Inspired by the concept of the anatomical planes, we propose a view-invariant motion profile that can capture the movements of different body parts during fall-related activities. Specifically, to construct a view-invariant motion profile that is independent of the location of the Kinect sensor, we build a body-attached coordinate system that is centered at a specific joint position. Then, the 3D positions of the skeletal joints, which are acquired using the Kinect sensor, are recomputed with respect to the origin of the body-attached coordinate system. Using the recomputed skeletal joint positions, we construct the three anatomical planes, such that the three planes are intersecting at the origin of the body-attached coordinate system. Finally, the movement of a specific body part can be described in terms of the position of the displacement vector, which extends between the initial and final positions of the movement with respect to the three anatomical planes. For example, the movement of the left hand in the upward direction can be represented in terms of the position of the displacement vector constructed between the initial and final positions of the hand movement with respect to the three anatomical planes. The construction procedure of the motion profile for each frame in the depth-map video sequence is as follows:

1. In order to analyze the depth-map video sequence, we utilize a sliding window of size W frames and overlap of size O frames between any two consecutive positions of the sliding window. Moreover, for a given position of the sliding window, the frames in the window are numbered

sequentially between 1 and W. For each window position, we build a motion profile for the human activities incorporated within the frames of the window as follows:

1.1 We create a body-attached coordinate system that is centered at the hip center (*hc*) joint. Then, we recalculate the positions of all of the other skeletal joints with respect to the *hc* joint. Figure 3 illustrates the constructed body-attached coordinate system.

1.2 Using the recalculated joint positions, we define the three anatomical planes, i.e., the TP, CP and SP, such that the three planes intersect at the *hc* joint (see Figure 3). Each plane is defined using three non-collinear skeletal joint positions as described below:

$$TP \equiv < hc, l\tilde{h}p, r\tilde{h}p > . \tag{1}$$

$$CP \equiv < hc, lsh, rsh > . \tag{2}$$

$$SP \equiv < hc, sc, sp > . \tag{3}$$

In Equation (1), $l\tilde{h}p$ and $r\tilde{h}p$ are the 3D positions of the left and right hip joints after applying a translation transformation along the Y-axis of the body-attached coordinate system to align the two joints with the *hc* joint.

1.3 We compute the displacement vectors for a subset of the skeletal joints, denoted as s_{dv}, that are related to the fall-event. In this study, the subset s_{dv} is composed of the following joints: sp, hd, rhd, lhd, rft and lft. The displacement vector of each skeletal joint $X \in s_{dv}$ is computed with respect to the first frame in the sliding window as follows:

$$\mathbf{DV}_{\mathbf{X}(k)} = [(\mathbf{X}(k) - \mathbf{X}(1)], \tag{4}$$

where $\mathbf{DV}_{\mathbf{X}(k)}$ is the displacement vector of the joint X in the k-th frame within the current sliding window. $\mathbf{X}(1)$ and $\mathbf{X}(k)$ are the 3D coordinates of the joint X in the first and k-th frames, respectively, within the current sliding window, where $k \in \{2, 3, \cdots, W\}$. Then, we identify the direction of motion of each joint in the set s_{dv} with respect to the three anatomical planes by calculating the signed distance between the displacement vector of that joint and each one of the three anatomical planes. The signed distance ($SgnDist$) of the displacement vector $\mathbf{DV}_{\mathbf{X}(k)}$ with respect to the anatomical plane $\mathbf{Y}$, where $\mathbf{Y} \in \{TP, CP, SP\}$, can be calculated as follows:

$$SgnDist(\mathbf{DV}_{\mathbf{X}(k)}, \mathbf{Y}) = \left(\frac{(\mathbf{Y}(2) - \mathbf{Y}(1)) \times (\mathbf{Y}(3) - \mathbf{Y}(1))}{\parallel (\mathbf{Y}(2) - \mathbf{Y}(1)) \times (\mathbf{Y}(3) - \mathbf{Y}(1)) \parallel} \right) \cdot \mathbf{DV}_{\mathbf{X}(k)}, \tag{5}$$

where $\mathbf{Y}(i)$ is the i-th joint that was used to construct the plane $\mathbf{Y}$ and $i \in \{1, 2, 3\}$. The operators · and × indicate the vector dot-product and cross-product operations, respectively. Depending on the sign (positive or negative) of the $SgnDist$ computed for a specific displacement vector with respect to each one of the three anatomical planes, we can determine if the displacement vector is located above or below the TP, in front or behind the CP and to the left or right of the SP.

1.4 The motion profile of each video frame in the current window position is defined as a vector, which consists of the calculated displacement vectors along with their associated signed distances for the skeletal joints in the set s_{dv}.

2. Then, we move the sliding window to the next position and repeat the procedure in the first step.

In the next subsection, we describe the construction procedure of the pose profile for each frame in the depth-map video sequence.

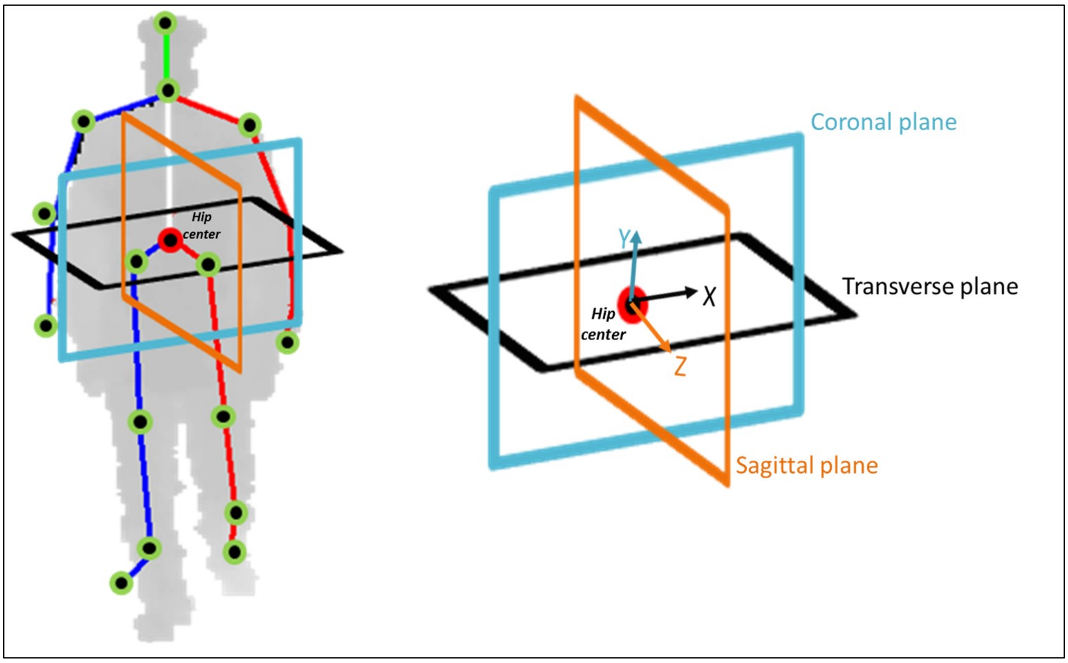

Figure 3. A schematic diagram that describes the construction of the body-attached coordinate system at the hip center joint along with the three anatomical planes.

2.2.2. Pose Profile

Fall-related activities involve different sub-activities that occur over time, such as the sitting pose, fall pose, stand still pose and other sub-activities, as described in Section 2.1. Having the ability to distinguish between these different sub-activities is crucial to enhance the fall detection process. However, different sub-activities may have similar body-postures due to inter- and intra-personal variations, which make the process of identifying these sub-activities challenging. In order to recognize different fall-related poses, we propose to construct a pose profile that can describe the different human postures that are incorporated within the different fall-related sub-activities using two types of geometrical relational-based features, namely the distance-based features (P_{df}) and angle-based features (P_{af}).

In order to obtain the distance-based features, we compute the Euclidean distances between the 3D locations of every pair of the skeletal joints as follows:

$$P_{df}(i,j) = \parallel S_i - S_j \parallel \ , \forall i \neq j, \tag{6}$$

where (i,j) represents any pair of the twenty skeletal joints employed in this study for detecting the fall-related activities as described in Section 2.2. S_i and S_j are the 3D positions of the i-th and j-th skeletal joints, respectively. Then, for a video frame at index t, the distance-based features $P_{df}(t)$ are defined as the vector combining all of the distances $P_{df}(i,j)$ for all $i \neq j$.

The angle-based features is represented as the set of time-varying angles between the different body parts associated with the fall-related sub-activities. Specifically, for each video frame, we utilize the skeletal joint positions to compute the angles listed in Table 2. Then, for a video frame at index t, the angle-based features $P_{af}(t)$ are defined as a vector, which consists of the values of the nine angles listed in Table 2.

Table 2. The angle-based features employed in the pose profile.

Angle	Description	Mathematical Formulation
θ_{Lshank}	The angle between the left shank and a translated transverse plane (TP_1) that passes through the left ankle joint position, where the left shank is defined as a line in the space that passes through the left ankle and left knee joint positions.	$\theta_{Lshank} = \arcsin \frac{\lvert\vec{n}_{TP_1} \cdot \vec{u}_{Lshank}\rvert}{\parallel \vec{n}_{TP_1} \parallel \parallel \vec{u}_{Lshank} \parallel}$, (7) where $\vec{n}_{TP_1}$ is the normal vector of the translated transverse plane TP_1, and $\vec{u}_{Lshank}$ is the direction vector of the left shank line.
θ_{Lthigh}	The angle between the left thigh and a translated transverse plane (TP_2) that passes through the left knee joint position, where the left thigh is defined as a line in the space that passes through the left hip and left knee joint positions.	$\theta_{Lthigh} = \arcsin \frac{\lvert\vec{n}_{TP_2} \cdot \vec{u}_{Lthigh}\rvert}{\parallel \vec{n}_{TP_2} \parallel \parallel \vec{u}_{Lthigh} \parallel}$, (8) where $\vec{n}_{TP_2}$ is the normal vector of the translated transverse plane TP_2, and $\vec{u}_{Lthigh}$ is the direction vector of the left thigh line.
θ_{Lknee}	The angle between the left thigh and left shank.	$\theta_{Lknee} = \theta_{Lthigh} - \theta_{Lshank}$ (9)
θ_{Rshank}	The angle between the right shank and a translated transverse plane (TP_3) that passes through the right ankle joint position, where the right shank is defined as the line in the space that passes through the right ankle and right knee joint positions.	$\theta_{Rshank} = \arcsin \frac{\lvert\vec{n}_{TP_3} \cdot \vec{u}_{Rshank}\rvert}{\parallel \vec{n}_{TP_3} \parallel \parallel \vec{u}_{Rshank} \parallel}$, (10) where $\vec{n}_{TP_3}$ is the normal vector of the translated transverse plane TP_3 and $\vec{u}_{Rshank}$ is the direction vector of the right shank line.
θ_{Rthigh}	The angle between the right thigh and a translated transverse plane (TP_4) that passes through the right knee joint position, where the right thigh is defined as the line in the space that passes through the right hip and right knee joint positions.	$\theta_{Rthigh} = \arcsin \frac{\lvert\vec{n}_{TP_4} \cdot \vec{u}_{Rthigh}\rvert}{\parallel \vec{n}_{TP_4} \parallel \parallel \vec{u}_{Rthigh} \parallel}$, (11) where $\vec{n}_{TP_4}$ is the normal vector of the translated transverse plane TP_4, and $\vec{u}_{Rthigh}$ is the direction vector of the right thigh line.
θ_{Rknee}	The angle between the right thigh and right shank.	$\theta_{Rknee} = \theta_{Rthigh} - \theta_{Rshank}$ (12)
θ_{trunck}	The angle between the trunk and the transverse plane (TP), where the truck is defined as the line in the space that passes through the the hip center and shoulder center joint positions.	$\theta_{trunk} = \arcsin \frac{\lvert\vec{n}_{TP} \cdot \vec{u}_{trunk}\rvert}{\parallel \vec{n}_{TP} \parallel \parallel \vec{u}_{trunk} \parallel}$, (13) where $\vec{n}_{TP}$ is the normal vector of the transverse plane TP and $\vec{u}_{trunk}$ is the direction vector of the trunk line.
θ_{Lhip}	The angle between the trunk and left thigh.	$\theta_{Lhip} = \theta_{Lthigh} - \theta_{trunk}$ (14)
θ_{Rhip}	The angle between the trunk and the right thigh.	$\theta_{Rhip} = \theta_{Rthigh} - \theta_{trunk}$ (15)

After computing the motion and pose profiles, the modified MPGD of the frame at index t in a depth-map video sequence is constructed by combining both the motion and pose profiles to form a descriptor vector as follows:

$$\mathbf{MPGD}(t) = [\mathbf{MP}(t), \mathbf{P}_{df}(t), \mathbf{P}_{af}(t)], \tag{16}$$

where $\mathbf{MP}(t)$ represents the motion profile of the video frame at index t. In the next subsection, we describe the proposed classification framework for detecting falls from partially-observed depth-map video sequences.

2.3. Fall Detection from Partially-Observed Depth-Map Video Sequences

A partially-observed depth-map video sequence (V) that consists of s observed video subsequences can be represented as the union of the observed video subsequences. Specifically, V can be defined as follows:

$$V \equiv \bigcup_{r=1}^{s} V_r[t_r(1) : t_r(l)], \tag{17}$$

where $V_r[t_r(1) : t_r(l)]$ is the r-th observed subsequence of frames and $t_r(1)$ and $t_r(l)$ are the indices of the first and last frames in the r-th observed video subsequence, respectively. In Figure 4, we illustrate the representation of partially-observed video sequences described in Equation (17). In particular, Figure 4 provides an example of a partially-observed depth-map video sequence that consists of three observed video subsequences. Such a partially-observed video can be represented as the union of the three observed video subsequences, namely $V_1[t_1(1) : t_1(l)]$, $V_2[t_2(1) : t_2(l)]$ and $V_3[t_3(1) : t_3(l)]$.

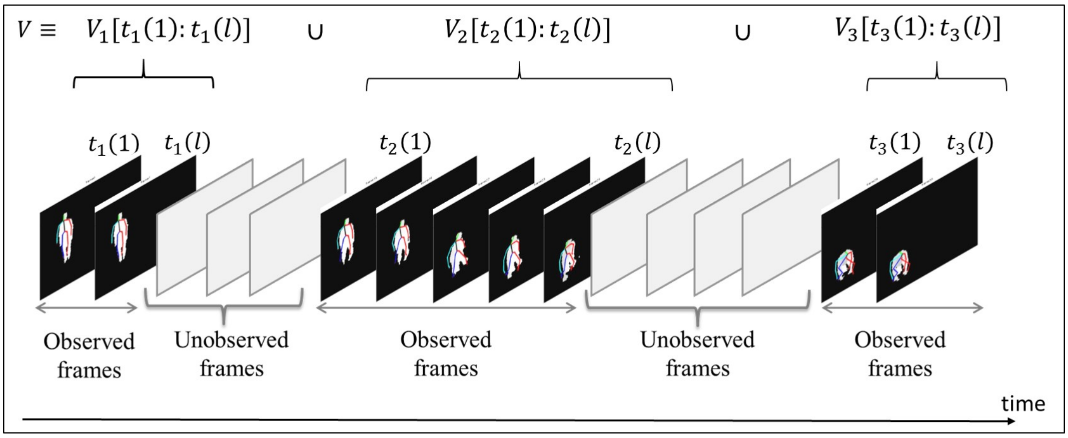

Figure 4. A partially-observed depth-map video sequence composed of three observed video subsequence.

In this work, we aim at classifying a partially-observed video sequence V into one of the four fall-related activities, namely walking (ω_1), sitting (ω_2), falling from standing (ω_3) and falling from sitting (ω_4). Let us denote the set of fall-related activities as $\Omega = \{\omega_c\}, c \in \{1, 2, 3, 4\}$. Then, the class posteriori probability that a partially-observed video V belongs to the activity class ω_c given the observed subsequences can be defined as follows:

$$P\Big(\omega_c | \bigcup_{r=1}^{s} V_r[t_r(1) : t_r(l)]\Big) \propto \sum_{r=1}^{s} \alpha_r P\Big(\omega_c | V_r[t_r(1) : t_r(l)]\Big), \tag{18}$$

where $P(\omega_c | V_r[t_r(1) : t_r(l)])$ is the class posteriori probability that the r-th observed video subsequence $V_r[t_r(1) : t_r(l)]$ belongs to the class ω_c. α_r represents the ratio between the length of the subsequence $V_r[t_r(1) : t_r(l)]$ and the sum of the lengths of the observed subsequences. The class of the partially-observed video V is the activity with the index c^* that maximizes the posteriori probability in Equation (18) and can be written as follows:

$$c^* = \arg\max_{c} P\Big(\omega_c | \bigcup_{r=1}^{s} V_r[t_r(1) : t_r(l)]\Big). \tag{19}$$

Partially-observed depth-map video sequences have different lengths. At the same time, the unobserved video subsequences can occur at any time and have various durations. Hence, in order to estimate the posteriori probability described in Equation (18), we need to have a length-independent representation of the observed video subsequences. In this paper, we propose to represent each observed video subsequence using a histogram of MPGDs that consists of n bins. In particular, the proposed histogram-based representation (HBR) employs the k-means clustering algorithm to build a codebook that consists of n codewords using the MPGDs extracted from the fully-observed training video sequences. The number of codewords n is selected to match the number of sub-activities in our dataset, which is equal to eight. Then, for any new video subsequence, we construct a histogram of eight bins, where each bin is associated with a specific codeword. The value of each bin represents the number of MPGDs in the given video subsequence that belong to a specific codeword after performing the k-means clustering algorithm. Therefore, the HBR can represent any video subsequence in our dataset using a vector $\mathbf{H} \in \mathbb{R}^8$.

In order to estimate the class posteriori probability $P\Big(\omega_c | \bigcup_{r=1}^{s} V_r[t_r(1) : t_r(l)]\Big)$, we start by constructing the HBR for each observed video subsequence. Specifically, the HBR for the r-th observed video subsequence $V_r[t_r(1) : t_r(l)]$ is denoted as $\mathbf{H}_{V_r[t_r(1):t_r(l)]}$. Then, the class posteriori probability in Equation (18) can be formulated as:

$$P\Big(\omega_c | \mathbf{H}_{V_1[t_1(1):t_1(l)]}, \cdots, \mathbf{H}_{V_r[t_r(1):t_r(l)]}, \cdots, \mathbf{H}_{V_s[t_s(1):t_s(l)]}\Big) \propto \sum_{r=1}^{s} \alpha_r P(\omega_c | \mathbf{H}_{V_r[t_r(1):t_r(l)]}), \tag{20}$$

where $P(\omega_c | \mathbf{H}_{V_r[t_r(1):t_r(l)]})$ is the class posteriori probability given the HBR of the r-th observed subsequence $V_r[t_r(1) : t_r(l)]$. In this work, we propose to utilize a multi-class support vector machine (SVM) classifier with a Gaussian radial basis function (RBF) kernel [33,34] to estimate the class posteriori probabilities of the observed video subsequences described in Equation (20). In order to train the multi-class SVM classifier, we utilize a one-against-one scheme using fully-observed video sequences. In particular, we divide each training sequence into ξ overlapped video segments, then using the extracted video segments, we construct a set of training pairs for each training video as follows:

$$(V_{trn}, \omega_c) = \{(\mathbf{H}_{V_{trn}[t_1(1):t_1(l)]}, \omega_c), \cdots, (\mathbf{H}_{V_{trn}[t_j(1):t_j(l)]}, \omega_c), \cdots, (\mathbf{H}_{V_{trn}[t_\xi(1):t_\xi(l)]}, \omega_c)\}, \tag{21}$$

where V_{trn} is a fully-observed training video of class ω_c. $\mathbf{H}_{V_{trn}[t_j(1):t_j(l)]}$ is the histogram-based representation of the j-th segment of the training video V_{trn}, and $t_j(1)$ and $t_j(l)$ are the indices of the first and last frames in the j-th video segment, respectively. After constructing the training pairs from each training video sequence, we employ a leave one video sequence out cross-validation procedure (LOVSO-CV) to train the multi-class SVM classifier and a grid-based search to tune the RBF kernel parameter $\sigma > 0$ and the regularization parameter $C > 0$. Using the trained multi-class SVM model and Equation (20), the class of a partially-observed video sequence V can be determined by rewriting Equation (19) as follows:

$$c^* = \arg\max_{c} P\Big(\omega_c | \mathbf{H}_{V_1[t_1(1):t_1(l)]}, \cdots, \mathbf{H}_{V_r[t_r(1):t_r(l)]}, \cdots, \mathbf{H}_{V_s[t_s(1):t_s(l)]}\Big). \tag{22}$$

In particular, to predict the class of a partially-observed testing video sequence, we compute the HBR of each observed video subsequence in the testing video. Then, we utilize the trained multi-class SVM model to approximate the class posteriori probability given the HBRs of the observed subsequences as described in Equation (20). Finally, the class of the testing video is determined using Equation (22), such that the testing video will be assigned to the activity class that has the maximum posterior probability given the HBRs of the observed video subsequences.

3. Experimental Results and Discussion

In order to evaluate the performance of the proposed approach, we utilize the collected dataset, described in Section 2.1, to develop three evaluation scenarios based on the number of unobserved video subsequences in the testing videos, including: fully-observed video sequences, single unobserved video subsequence with random length and two unobserved video subsequences with random lengths. Moreover, we utilize the LOVSO-CV procedure to evaluate the performance of each scenario. In particular, the classifiers in each scenario are trained using all of the video sequences except one video sequence that is used for testing. This evaluation scheme is repeated for all possible combinations, and the overall result is computed by averaging the results obtained from each repetition. For all evaluation scenarios, the window size W and the overlap size O of the motion profile are selected experimentally and are equal to three frames and one frame, respectively. In order to train the SVM classifiers in the second and third evaluation scenarios, we divided each video sequence into overlapping segments. The size of each segment is equal to 20% of the total number of frames in the video sequence, and the overlap between any two consecutive segments is equal to 50%. Moreover, as the lengths of the unobserved video sequences are random, we evaluate the performance of the proposed approach for the second and third evaluation scenarios by repeating the LOVSO-CV procedure ten times, such that in each repetition, we generate unobserved video subsequences with different random lengths in the testing sequences. Then, the average values of the recognition accuracy, precision, recall and F1-measure are computed as performance evaluation metrics over the ten LOVSO-CV train-test repetitions. These metrics are defined as follows:

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}, \tag{23}$$

$$Precision = \frac{TP}{(TP + FP)}, \tag{24}$$

$$Recall = \frac{TP}{(TP + FN)}, \tag{25}$$

$$F_1 - measure = \frac{2TP}{(2TP + FP + FN)}, \tag{26}$$

where TP is the number of true positive cases, TN is the number of true negative cases, FP is the number of false positive cases and FN is the number of false negative cases. Next, we discuss the results of our proposed approach for each evaluation scenario.

3.1. Evaluation on the Fully-Observed Video Sequences Scenario

In this section, we evaluate the performance of the proposed approach in recognizing the four fall-related activities from fully-observed video sequences. Specifically, in this evaluation scenario, the video sequences are fully observed as described in Figure 1a. For the purpose of this evaluation scenario, we have trained a multi-class SVM classifier using the HBRs obtained from unsegmented training video sequences. In particular, for each training video V_{trn} of length N frames, we construct the HBR of the video frames with indices between one and N. Using the obtained HBR from each training video sequence, we train the multi-class SVM classifier to identify the four fall-related activities in our dataset.

In order to classify a testing video sequence V_{tst} of length T, we construct the HBR of the video frames with indices between one and T in the video V_{tst}, namely $\mathbf{H}_{V_{tst}[1:T]}$. Then, using the trained multi-class SVM classifier, the class of V_{tst} can be determined by rewriting Equation (22) as follows:

$$c^* = \arg\max_{c} P(\omega_c|\mathbf{H}_{V_{tst}[1:T]}). \tag{27}$$

Table 3 shows the results of recognizing the four fall-related activities expressed using the precision, recall and F1-measure as an evaluation metric. The average recognition accuracy of fully-observed video sequences is 93.6%. The results demonstrate the capability of the proposed approach to detect falls from fully-observed video sequences.

Table 3. The recognition results of the four fall-related activities from fully-observed video sequences.

Activity	Precision	Recall	F1-Measure
Walking	95.0%	94.4%	94.7%
Sitting	92.1%	90.1%	91.1%
Falling from sitting	97.0%	96.5%	96.7%
Falling from standing	96.1%	95.3%	95.7%
Overall average	95.1%	92.8%	94.6%

In comparison, the Kinect-based approach proposed by Marzahl et al. [35], which was evaluated using 55 fall depth-map videos, achieved an average fall classification accuracy of 93%. Similarly, the Kinect-based system introduced by Planinc and Kampel [36] achieved fall detection accuracies between 86.1% and 89.3% based on a dataset that includes 40 falls and 32 non-falls depth-map videos. In fact, the results reported in our study for the fully-observed depth-map video sequences are comparable to the results reported in the previous approaches. It is worth noting that the main focus of our work is to detect falls form partially-observed depth-map video sequences, which has not been investigated in previous studies. The following subsections provide the performance evaluation results obtained by our approach for the partially-observed depth-map video sequences.

3.2. Evaluation on the Single Unobserved Video Subsequence with Random Length Scenarios

In this section, we evaluate the performance of the proposed approach in recognizing the four fall-related activities from partially-observed video sequences. In particular, we have constructed partially-observed video sequences by generating temporal gaps with random lengths at the beginning, end and middle of the testing video sequences. Next, we discuss the evaluation results of our proposed approach for each temporal gap configuration.

3.2.1. Evaluation Results When the Temporal Gap Is at the Beginning of the Video Sequences

In this evaluation scenario, we aim at evaluating the performance of the proposed approach in recognizing the four fall-related activities when the unobserved video subsequence is happening at the beginning of the video sequence. In particular, we have investigated the scenario in which the temporal gap is mainly affecting the video frames belonging to the first sub-activity and a subset of the second sub-activity in each of the four fall-related activities. Hence, our assumption is that the majority of the observed video frames belong to the last sub-activity. To achieve this goal, we have set the temporal gap interval to $[1 : \lambda]$, where λ is a random integer that is larger than one and less than 60% of the total length of the video sequence. Table 4 shows the results of recognizing the four fall-related activities expressed in terms of the precision, recall and F1-measure as an evaluation metric. In addition, Table 4 provides the average length of the temporal gaps introduced in the testing video sequences for each activity over the 10 repetitions of the LOVSO-CV procedure. The proposed approach achieved an average recognition accuracy of 81.3% in identifying the four fall-related activities from partially-observed video sequences.

Table 4. The recognition results of the four fall-related activities from partially-observed videos when the missing subsequence of frames is at the beginning of the testing video sequences.

Activity	Precision	Recall	F1-Measure	Average Gap's Length (Frames)
Walking	90.8%	81.5%	85.9%	32
Sitting	84.0%	85.0%	84.5%	37
Falling from sitting	76.2%	79.0%	77.6%	34
Falling from standing	77.0%	78.4%	77.7%	38
Overall average	82.0%	81.0%	81.4%	35

The results in Table 4 indicate that the average recognition accuracy has been reduced to 81.3% compared to 93.6% that was obtained in the fully-observed scenario. This can be attributed to the fact that the ratio of the average length of the introduced temporal gaps across the four activities in this scenario, which is equal to 35 frames, to the average sequence length in our dataset, which is equal to 84 frames, is approximately 42%. The relatively high lengths of the introduced temporal gaps can generate unobserved subsequences that represent multiple sub-activities, which in turn reduces the recognition accuracy. For example, when the unobserved video subsequence is spanning the first two sub-activities in the falling from sitting and falling from standing activities, the remaining sub-activity represents the falling pose, which is common to both activities. This can reduce the ability of the proposed approach to distinguish between different activities that involve fall-related events.

3.2.2. Evaluation Results When the Temporal Gap Is at the End of the Video Sequences

In this evaluation scenario, we aim at evaluating the performance of the proposed approach in recognizing the four fall-related activities when the unobserved video subsequence is happening at the end of the video sequence. This is similar to the prediction scenario illustrated in Figure 1b. In particular, we have investigated the scenario in which the temporal gap is mainly affecting the video frames belonging to the last sub-activity and a subset of the second sub-activity in each of the four fall-related activities. Hence, our assumption is that the majority of the observed video frames belong to the first sub-activity. To achieve this goal, we have set the temporal gap interval to $[\gamma : T]$, where γ is a random integer in the range $(\frac{4T}{10}, T)$. Table 5 provides the recognition results of the four fall-related activities expressed in terms of the precision, recall and F1-measure as an evaluation metric. In addition, Table 5 provides the average length of the temporal gaps introduced in the testing video sequences for each activity over the 10 repetitions of the LOVSO-CV procedure. The proposed approach achieved an average recognition accuracy of 73.3% in recognizing the four fall-related activities from partially-observed video sequences.

Table 5. The recognition results of the four fall-related activities from partially-observed videos when the missing subsequence of frames is at the end of the testing video sequences.

Activity	Precision	Recall	F1-Measure	Average Gap's Length (Frames)
Walking	79.7%	81.2%	80.4%	29
Sitting	69.3%	80.0%	74.3%	35
Falling from sitting	71.5%	74.0%	72.7%	32
Falling from standing	61.8%	68.0%	64.7%	36
Overall average	70.6%	75.8%	73.0%	33

Recognizing human activities from partially-observed video sequences with the unobserved video subsequence at the end of the video is considered challenging. The reason behind that is the absence of some key sub-activities that can distinguish different activities from each other. For example, when the sub-activity that represents the falling pose is unobserved, distinguishing between sitting and falling from sitting activities becomes challenging, especially when the duration of the falling from sitting pose sub-activity is short. Similarly, the absent of the fall pose increases the difficulty in distinguishing between the walking and falling from standing activities. This explain the reduction in the average recognition accuracy between this evaluation scenario and the previously described scenario in Section 3.2.1.

3.2.3. Evaluation Results When the Temporal Gap Is at the Middle of the Video Sequences

In this evaluation scenario, we aim at evaluating the performance of the proposed approach in recognizing the four fall-related activities when the unobserved video subsequence is happening at the middle of the video sequence. In particular, we have investigated the scenario in which the temporal gap is mainly affecting the video frames belonging to the second sub-activity, along with a subset of the frames that belong to the first and last sub-activities in the four fall-related activities. Hence, our assumption is that the majority of the observed video frames are belonging to the first and last sub-activities. To achieve this goal, we have set the temporal gap interval to $[\beta_1, \beta_2]$, where β_1 and β_2 are random integers that satisfy $\frac{2T}{10} < \beta_1 < \beta_2 < \frac{8T}{10}$. The scenario in this section can be viewed as a simplified version of the scenario described in Figure 1c, as it consists of a single unobserved subsequence of frames. Table 6 presents the recognition results of the four fall-related activities expressed in terms of the precision, recall and F1-measure as an evaluation metric. The last column in Table 4 provides the average length of the temporal gaps introduced in the testing video sequences for each activity over the 10 repetitions of the LOVSO-CV procedure. The proposed approach achieved an average recognition accuracy of 78.1% in recognizing the four fall-related activities from partially-observed video sequences.

Table 6. The recognition results of the four fall-related activities from partially-observed videos when the missing subsequence of frames is at the middle of the testing video sequences.

Activity	Precision	Recall	F1-Measure	Average Gap's Length (Frames)
Walking	86.0%	72.7%	78.8%	21
Sitting	76.2%	85.0%	80.4%	27
Falling from sitting	82.1%	84.0%	83.0%	24
Falling from standing	73.2%	82.0%	77.3%	22
Overall average	79.4%	80.9%	79.9%	24

The results in Table 6 show that the proposed approach achieved a better recognition accuracy in comparison with the results obtained when the unobserved video subsequences were at the end of the video sequences, as described in Section 3.2.2. This can be attributed to the fact that, in this scenario, we observe two video subsequences from each testing video. These observed video subsequences comprise the starting and ending sub-activities of each activity. Furthermore, the observed video subsequences might contain subsets of the video frames that belong to the intermediate sub-activities in each activity. This in turn can enhance the recognition accuracy as depicted in Table 6.

3.3. Evaluation of the Two Unobserved Video Subsequences with Random Lengths Scenarios

In this section, we evaluate the performance of the proposed approach in recognizing the four fall-related activities from partially-observed video sequences with two unobserved frame subsequences. In particular, we consider two configurations for the locations of the two unobserved frames subsequences. In the first configuration, we construct partially-observed video sequences by generating two temporal gaps with random lengths at the beginning and the end of the testing video sequences. In the second configuration, we generate two temporal gaps with random lengths between the beginning and the end of the testing video sequences. Next we discuss the evaluation results of our proposed approach for the aforementioned two configurations.

3.3.1. Evaluation Results When the Two Temporal Gaps Are at the Beginning and the End of the Video Sequences

In this evaluation scenario, we aim at evaluating the performance of the proposed approach in recognizing the four fall-related activities when the unobserved video subsequences are at the beginning and the end of the video sequence. To achieve that, we created two temporal gaps. The first temporal gap spans a subset of video frames that belong to the first and second sub-activities; while the second gap spans video frames that belong to the second and last sub-activities. Hence, the majority of the remaining frames belong to the second sub-activity. To achieve this goal, we have set the intervals of the two temporal gaps to $[1, \beta_1]$ and $[\beta_2, T]$, where β_1 and β_2 are two random integers in the ranges of $1 < \beta_1 < \frac{4T}{10}$ and $\frac{6T}{10} < \beta_2 < T$, respectively. Table 7 presents the recognition results of the four fall-related activities expressed in terms of the precision, recall and F1-measure as an evaluation metric. The last two columns in Table 7 provide the average lengths of the two temporal gaps introduced in the testing video sequences of each activity over the 10 repetitions of the LOVSO-CV procedure. The proposed approach achieved an average recognition accuracy of 58.8% in identifying the four fall-related activities from partially-observed video sequences.

Table 7. Recognition results of the fall-related activities from partially-observed videos with missing video subsequences at the beginning (first temporal gap) and the end (second temporal gap) of the video sequences.

Activity	Precision	Recall	F1-Measure	Average Length of the First Temporal Gap (Frames)	Average Length of the Second Temporal Gap (Frames)
Walking	63.3%	52.7%	57.5%	18	16
Sitting	64.4%	62.0%	68.7%	22	27
Falling from sitting	57.0%	66.0%	61.1%	20	18
Falling from standing	54.2%	64.0%	58.7%	17	19
Overall average	59.7%	61.2%	61.5%	19	20

Table 7 shows that the recognition results have been reduced drastically compared with the results of the previous evaluation scenarios. This reduction in the recognition accuracy is due to the large amount of unobserved video frames, which are mainly frames from the first and the last sub-activities of each activity. These sub-activities have a key role in distinguishing between different fall-related activities, such as sitting and falling from sitting activities.

3.3.2. Evaluation Results When the Two Temporal Gaps Are between the Beginning and the End of the Video Sequences

In this evaluation scenario, we aim at evaluating the performance of the proposed approach in recognizing the four fall-related activities; the unobserved video subsequences are at random locations between the beginning and the end of the video sequence, which is similar to the scenario described in Figure 1c. To achieve that, we created two temporal gaps. The first gap spans a subset of the video frames that belong to the first and second sub-activities. While the second gap spans a subset of the video frames that belong to the second and last sub-activities. Hence, the remaining frames are sparsely distributed between the first, second and last sub-activities. To implement the temporal gaps in this evaluation scenario, we have set the intervals of the two temporal gaps to $[\beta_1, \beta_2]$ and $[\gamma_1, \gamma_2]$, where β_1, β_2, γ_1 and γ_2 are random integers that satisfy the two conditions: $\frac{2T}{10} < \beta_1 < \beta_2 < \frac{4T}{10}$ and $\frac{6T}{10} < \gamma_1 < \gamma_2 < \frac{8T}{10}$. Table 8 provides the recognition results of the four fall-related activities expressed in terms of the precision, recall and F1-measure as an evaluation metrics. The last two columns in Table 8 provide the average lengths of the two temporal gaps introduced in the testing video sequences of each activity over the 10 repetitions of the LOVSO-CV procedure. The proposed approach achieved an average recognition accuracy of 71.4% in identifying the four fall-related activities from partially-observed video sequences.

Table 8. Recognition results of the four fall-related activities from partially-observed videos with two unobserved video subsequences between the beginning and the end of each testing video sequence.

Activity	Precision	Recall	F1-Measure	Average Length of the First Temporal Gap (Frames)	Average Length of the Second Temporal Gap (Frames)
Walking	78.3%	73.6%	75.9%	10	9
Sitting	73.1%	72.0%	72.5%	11	12
Falling from sitting	71.6%	76.3%	73.9%	13	10
Falling from standing	72.2%	78.1%	75.0%	12	11
Overall average	73.8%	75.0%	74.3%	11	10

The results in Table 8 show that, in this evaluation scenario, the proposed approach achieved a better recognition accuracy compared with the results obtained when the two temporal gaps were at the beginning and the end of the testing video sequences, as described in Section 3.3.1. This can be justified by observing that each testing video consists of three observed video subsequences after creating the temporal gaps. The first observed video subsequence is at the beginning of the testing video and consists of video frames from the first sub-activity. The second observed video subsequence is at the middle of the testing video and might contain frames from more than one sub-activity depending on the lengths of the temporal gaps. The third observed video subsequence is at the end of the testing video and consists of video frames from the last sub-activity. This implies that the observed video subsequences are comprising video frames from the different sub-activities of an activity in a given testing video, which can enhance the recognition accuracy.

4. Conclusions

In this paper, we proposed an approach for fall detection from partially-observed depth-map video sequences. The proposed approach utilizes the Microsoft Kinect sensor to build a view-invariant descriptor for human activities, namely the motion-pose geometric descriptor (MPGD). To detect falls in the partially-observed depth-map video sequence, we segmented fully-observed training video sequences into overlapping video segments. Then, we constructed an HBR of the MPGDs extracted from the video frames within each segment. Using the computed HBRs, we trained an SVM classifier with a probabilistic output to predict the class of the performed activity in a given partially-observed video. To classify a new video with unobserved frames subsequences, we utilized the trained SVM models to compute the class posteriori probability of each observed subsequence. Then, we combined the computed posteriori probabilities from all observed subsequences to obtain an overall class posteriori probability for the partially-observed video. In order to evaluate the performance of the proposed approach, we utilized the Kinect sensor to record a dataset of depth-map video sequences that simulates four fall-related activities of elderly people, namely walking, sitting, falling form standing and falling from sitting. Furthermore, using the collected dataset, we have developed three evaluation scenarios based on the number of unobserved video subsequence in the testing videos. Experimental results show the potential of the proposed approach to detect falls from partially-observed videos efficiently.

In the future, we intend to extend our approach to utilize multiple Kinect sensors to overcome the distance limitation of the Kinect sensor, the subject occlusion problem and the requirement of having the subject in the frontal or near-frontal view with respect to the Kinect sensor. Such an extension can also enhance the accuracy of localizing the 3D skeletal joint positions. Furthermore, we plan to extend our dataset to evaluate the performance of the proposed approach in recognizing a larger number of human activities in partially-observed depth-map video sequences. Moreover, we plan to evaluate the proposed approach using more complex datasets that comprise activities of more than one person. In addition, for each examined human activity, a higher number of evaluation trails will be employed that include various configurations, such as different lengths and locations of the induced temporal gaps.

Author Contributions: Rami Alazrai and Mohammad Momani conceived of and designed the experiments. Rami Alazrai, Mohammad I. Daoud and Mohammad Momani performed the experiments. Rami Alazrai and Mohammad Momani analyzed the data. Rami Alazrai and Mohammad I. Daoud contributed reagents/materials/analysis tools. Rami Alazrai and Mohammad I. Daoud wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Costello, D.; Carone, G. Can europe afford to grow old. *Int. Monet. Fund Financ. Dev. Mag.* **2006**, *43*, 28.
2. United States Census Bureau, Population Profile of the United States. Available online: www.census.gov (accessed on 18 December 2016).

3. Centers for Disease Control and Prevention, Web Based Injury Statistics Query and Reporting. Available online: http://www.cdc.gov/injury/wisqars/index.html (accessed on 21 December 2016).
4. Murphy, S.L. *Final Data for 1998 National Vital Statistics Reports*; Technical Report; National Center for Health Statistics: Hyattsville, MD, USA, 2000.
5. Hsieh, J.W.; Hsu, Y.T.; Liao, H.Y.M.; Chen, C.C. Video-based human movement analysis and its application to surveillance systems. *IEEE Trans. Multimedia* **2008**, *10*, 372–384.
6. Vellas, B.J.; Wayne, S.J.; Romero, L.J.; Baumgartner, R.N.; Garry, P.J. Fear of falling and restriction of mobility in elderly fallers. *Age Ageing* **1997**, *26*, 189–193.
7. Feng, W.; Liu, R.; Zhu, M. Fall detection for elderly person care in a vision-based home surveillance environment using a monocular camera. *Signal Image Video Process.* **2014**, *8*, 1129–1138.
8. Narayanan, M.; Lord, S.; Budge, M.; Celler, B.; Lovell, N. Falls Management: Detection and Prevention, using a Waist-mounted Triaxial Accelerometer. In Proceedings of the 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Lyon, France, 23–26 August 2007.
9. Boyle, J.; Karunanithi, M. Simulated fall detection via accelerometers. In Proceedings of the International Conference of the IEEE Engineering in Medicine and Biology Society, Vancouver, BC, Canada, 20–25 August 2008.
10. Wang, C.C.; Chiang, C.Y.; Lin, P.Y.; Chou, Y.C.; Kuo, I.T.; Huang, C.N.; Chan, C.T. Development of a Fall Detecting System for the Elderly Residents. In Proceedings of the 2nd International Conference on Bioinformatics and Biomedical Engineering, Shanghai, China, 16–18 May 2008.
11. Rougier, C.; Meunier, J.; St-Arnaud, A.; Rousseau, J. Robust Video Surveillance for Fall Detection Based on Human Shape Deformation. *IEEE Trans. Circuits Syst. Video Technol.* **2011**, *21*, 611–622.
12. Auvinet, E.; Multon, F.; Saint-Arnaud, A.; Rousseau, J.; Meunier, J. Fall Detection With Multiple Cameras: An Occlusion-Resistant Method Based on 3-D Silhouette Vertical Distribution. *IEEE Trans. Inf. Technol. Biomed.* **2011**, *15*, 290–300.
13. Rougier, C.; Meunier, J.; St-Arnaud, A.; Rousseau, J. Monocular 3D Head Tracking to Detect Falls of Elderly People. In Proceedings of the 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, New York, NY, USA, 31 August–3 September 2006.
14. Mubashir, M.; Shao, L.; Seed, L. A survey on fall detection: Principles and approaches. *Neurocomputing* **2013**, *100*, 144–152.
15. Yu, M.; Yu, Y.; Rhuma, A.; Naqvi, S.M.R.; Wang, L.; Chambers, J.A. An online one class support vector machine-based person-specific fall detection system for monitoring an elderly individual in a room environment. *IEEE J. Biomed. Health Inform.* **2013**, *17*, 1002–1014.
16. Dai, J.; Bai, X.; Yang, Z.; Shen, Z.; Xuan, D. PerFallD: A pervasive fall detection system using mobile phones. In Proceedings of the 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), Mannheim, Germany, 29 March–2 April 2010; pp. 292–297.
17. Enayati, M.; Banerjee, T.; Popescu, M.; Skubic, M.; Rantz, M. A novel web-based depth video rewind approach toward fall preventive interventions in hospitals. In Proceedings of the 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Chicago, IL, USA, 26–30 August 2014; pp. 4511–4514.
18. Dubois, A.; Charpillet, F. A gait analysis method based on a depth camera for fall prevention. In Proceedings of the 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Chicago, IL, USA, 26–30 August 2014; pp. 4515–4518.
19. Li, Y.; Berkowitz, L.; Noskin, G.; Mehrotra, S. Detection of patient's bed statuses in 3D using a Microsoft Kinect. In Proceedings of the 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Chicago, IL, USA, 26–30 August 2014; pp. 5900–5903.
20. Zhang, C.; Tian, Y.; Capezuti, E. Privacy preserving automatic fall detection for elderly using RGBD cameras. In *International Conference on Computers for Handicapped Persons*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 625–633.
21. Huang, S.H.; Pan, Y.C. Learning-based Human Fall Detection using RGB-D cameras. In Proceedings of the International Conference on Machine Vision Applications, Kyoto, Japan, 20–23 May 2013.
22. Garrido, J.E.; Penichet, V.M.; Lozano, M.D.; Valls, J.A.F. Automatic Detection of Falls and Fainting. *J. Univ. Comput. Sci.* **2013**, *19*, 1105–1122.

23. Gasparrini, S.; Cippitelli, E.; Spinsante, S.; Gambi, E. A Depth-Based Fall Detection System Using a Kinect® Sensor. *Sensors* **2014**, *14*, 2756–2775.
24. Rougier, C.; Auvinet, E.; Rousseau, J.; Mignotte, M.; Meunier, J. Fall detection from depth map video sequences. In *International Conference on Smart Homes and Health Telematics*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 121–128.
25. Flores-Barranco, M.M.; Ibarra-Mazano, M.A.; Cheng, I. Accidental Fall Detection Based on Skeleton Joint Correlation and Activity Boundary. In *International Symposium on Visual Computing*; Springer: Berlin/Heidelberg, Germany, 2015, pp. 489–498.
26. Tong, L.; Song, Q.; Ge, Y.; Liu, M. HMM-Based Human Fall Detection and Prediction Method Using Tri-Axial Accelerometer. *IEEE Sens. J.* **2013**, *13*, 1849–1856.
27. Alazrai, R.; Mowafi, Y.; Hamad, E. A fall prediction methodology for elderly based on a depth camera. In Proceedings of the 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Milano, Italy, 25–29 August 2015; pp. 4990–4993.
28. Stone, E.E.; Skubic, M. Fall detection in homes of older adults using the Microsoft Kinect. *IEEE J. Biomed. Health Inform.* **2015**, *19*, 290–301.
29. Cao, Y.; Barrett, D.; Barbu, A.; Narayanaswamy, S.; Yu, H.; Michaux, A.; Lin, Y.; Dickinson, S.; Mark Siskind, J.; Wang, S. Recognize human activities from partially observed videos. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2658–2665.
30. Alazrai, R.; Zmily, A.; Mowafi, Y. Fall detection for elderly using anatomical-plane-based representation. In Proceedings of the 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Chicago, IL, USA, 26–30 August 2014; pp. 5916–5919.
31. Snell, R.S. *Clinical Anatomy by Regions*, 9th ed.; Lippincott Williams & Wilkins, Walters Kluwer: Philadelphia, PA, USA, 2011.
32. Alazrai, R.; Mowafi, Y.; Lee, C.G. Anatomical-plane-based representation for human-human interactions analysis. *Pattern Recognit.* **2015**, *48*, 2346–2363.
33. Wu, T.F.; Lin, C.J.; Weng, R.C. Probability Estimates for Multi-class Classification by Pairwise Coupling. *J. Mach. Learn. Res.* **2004**, *5*, 975–1005.
34. Chang, C.C.; Lin, C.J. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.* **2011**, *2*, 1–27.
35. Marzahl, C.; Penndorf, P.; Bruder, I.; Staemmler, M., Unobtrusive Fall Detection Using 3D Images of a Gaming Console: Concept and First Results. In *Ambient Assisted Living: 5. AAL-Kongress 2012 Berlin, Germany, January 24–25, 2012*; Wichert, R., Eberhardt, B., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 135–146.
36. Planinc, R.; Kampel, M. Introducing the use of depth data for fall detection. *Pers. Ubiquitous Comput.* **2013**, *17*, 1063–1072.