Fall Detection Using Body-worn Accelerometer and Depth Maps Acquired by Active Camera

Michal Kepski[2] and Bogdan Kwolek[1]

[1] AGH University of Science and Technology, 30 Mickiewicza Av., 30-059 Krakow, Poland
http://home.agh.edu.pl/~bkw/contact.html
[2] University of Rzeszow, 16c Rejtana Av., 35-959 Rzeszów, Poland
mkepski@univ.rzeszow.pl

Abstract. In the presented system to person fall detection a body-worn accelerometer is used to indicate a potential fall and a ceiling-mounted depth sensor is utilized to authenticate fall alert. In order to expand the observation area the depth sensor has been mounted on a pan-tilt motorized head. If the person acceleration is above a preset threshold the system uses a lying pose detector as well as examines a dynamic feature to authenticate the fall. Thus, more costly fall authentication is not executed frame-by-frame, but instead we fetch from a circular buffer a sequence of depth maps acquired prior to the fall and then process them to confirm fall alert. We show that promising results in terms of sensitivity and specificity can be obtained on publicly available UR Fall Detection dataset.

Keywords: Smart Home; Human Behavior Analysis; Fall Detection.

1 Introduction

The main goal of user-centered ubiquitous computing is to sense changes in human environments in order to provide effective personal assistance. It is expected that the use of ubiquitous computing and ambient intelligence can help to meet the societal challenges posed by aging of population [1]. Smart home technology [2] is considered as important part of the ubiquitous computing. One of the most critical factors limiting the realization of ambient intelligence in broader scale is limited number of cost-effective and energy-efficient devices for human activity monitoring and/or power saving modules for ubiquitous computing. One of the most promising areas of applications of the embedded vision is healthcare [3]. Particularly, embedded vision technology has strong potential to considerably change the healthcare at home, for example through mobile phone applications that monitor the user's state of health and report it to a medical center.

With the aim to enable prolonged independent living in a safe and homely environment, automatic fall detection is an important task [4]. Inertial sensors such as accelerometers and gyroscopes have proven to be very useful in the analysis of motion patterns [5]. Compared to vision sensors, wearable inertial sensors offer advantages in terms of size, weight, ease of use and, most importantly, power consumption and costs of use. They permit data collection outside of laboratory environments and are considered as one of the best sensors for ubiquitous health monitoring [3]. In context-aware systems various kinds of sensors are usually deployed in the environment in order to detect falls. The most common sensors are cameras, microphones, floor sensors and pressure sensors. One of their advantages is that the person undergoing monitoring does not need to wear any special device. A short time ago the Kinect's sensor has been proposed to be employed in fall detection [6][7][4]. As demonstrated in [6][7], the depth maps delivered by Kinect are sufficient to delineate the person from the background. Moreover, due to using speckle pattern of infrared laser light to estimate the dense depth maps, the object detection can be done any time. A recent survey [4] discusses several approaches to Kinect-based fall detection. However, the available algorithms do not achieve both high sensitivity and specificity. By combining video or depth maps and acceleration data, the detection of emergency situations [8][7] as well as activity recognition [9] can be improved greatly.

2 Embedded system for fall detection

The system detects fall events on the basis of depth maps acquired by a ceiling-mounted Kinect sensor as well as motion data, which is acquired by a body-worn accelerometer. To expand the observation area a pan-tilt head is used to rotate the Kinect and to follow a moving person. Initially, a nearest neighbor interpolation is executed to fill the holes in the depth map and to get the map with meaningful values for all pixels. The interpolation is based on the median filter with a 5×5 window. To improve the delineation of the person in the depth maps the algorithm extracts the floor and then removes their corresponding pixels from the depth map. Given the extracted person's blob in the last depth map, the algorithm uses it as seed region in the subsequent region growing, which is responsible for delineation of the person from the background depth map in the current frame. Afterwards, a Support Vector Machine (SVM) is utilized optionally to acknowledge the person presence within the depth blob. Then, the center of gravity of the blob is determined. Finally, given the person's centroid, the pan-tilt head rotates the Kinect to keep the target in the central part of the depth maps. Given the delineated person, lying pose and dynamic transition classifiers are executed to raise alarm in case of fall event. The features describing the dynamic transitions are calculated using depth maps, which are continuously stored in a circular buffer. Thanks to the use of the accelerometer, the cascade classifier responsible for fall detection is not executed frame-by-frame, but is triggered only if a person acceleration is above a preset threshold, see Fig. 1.

A system for fall detection should to be as cheap as possible in order to be affordable for the elderly. It should also be easy-to-install and consume least amount energy. Hence, our system consists of inexpensive sensing devices such as Kinect, an accelerometer worn on human body, a low-cost pan-tilt unit, and low-cost PandaBoard at which the algorithms are executed. The PandaBoard was

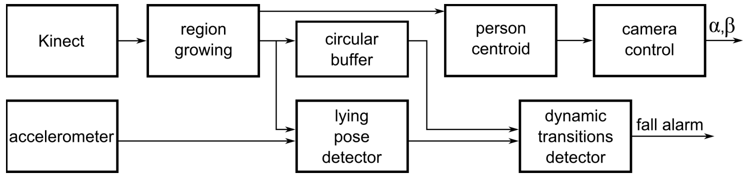

Fig. 1. Block diagram of the embedded system for fall detection using depth maps and body-worn accelerometer.

selected for the implementation of the application since it offers high performance at a low cost with only 3.5 Watt power consumption under full load. The data from the accelerometer are transmitted wirelessly to the processing unit. The Kinect sensor (Asus Xtion Pro Live) is connected to the board via USB. The microcontroller of the head is connected with the PandaBoard through I2C bus.

Figure 2 depicts sample images, which were captured by ceiling-mounted Kinect. As we can observe, thanks to the pan-tilt head the observation area is broaden. For the Kinect sensor mounted on the height of 2.6 m from the floor, the observation area is about 5.5 m^2. Owing to the use of the pan-tilt unit, the Kinect is able to examine a room of average size, say 15-20 m^2. The RGB images are not processed and they are used only for visualization purposes.

Fig. 2. Example top-view images captured by Kinect mounted on pan-tilt head, RGB images (top row), depth images (bottom row).

3 Person delineation using active pan-tilt depth sensor

The pursuing a moving person is achieved by a series of saccades of the pan-tilt head to keep the detected object in central part of permanently acquired depth maps. The object position is expressed as the centroid of the delineated area. Below we detail the depth region growing that delineates the person undergoing tracking in maps acquired by an active camera. We also present an algorithm that supports locating the person's head in case of region chaining.

3.1 Delineation of person using region growing

The person is delineated from the background assuming that he/she occupies an integrated region in 3D space. Owing to extracting the floor in advance, we avoid incorporating of the neighboring pixels from the floor into the person region. The developed depth region growing starts with selecting a seed point in a current frame. Assuming that there is a common depth region between regions belonging to a person in two consecutive frames, such seed region is determined using the `and` operator between the previously delineated depth region belonging to person and the current depth map. Afterwards, the algorithm repeatedly seeks all neighboring pixels of the current region. The selected pixels are sorted according to their depth similarities and then they are stored in a list of candidate pixels. The depth similarity is the Euclidean distance between the depth values of a pixel from such a list and its closest pixel from the current region. It is employed in order to verify if a neighboring pixel around a region pixel is allowed to be merged with the region.

3.2 Finding human in depth maps

Ordinary region growing algorithms suffer from the problem of region chaining (overspill), which occurs when two regions are grown into one region while they are actually separated from each other, see also Fig. 3a. In order to improve the delineation of the person in such situations as well as to improve the pursuing of the person by the active camera, we execute a person detector. The detector permits also automatic initialization of person tracking. The person detection is done by a SVM for linear classification that is built on Histogram of Oriented Depths (HOD) features [10]. The HOD descriptors locally encode the orientation of depth changes, see Fig. 3c. In our approach they are calculated in sub-windows, which are scaled according to their distances to the camera. The scaling is according to the distance between the camera and the closest pixels from the sub-window. Such sub-windows of fixed size are then subdivided into cells. The descriptors are calculated for each cell and then the oriented depth gradients are collected into 1D histograms.

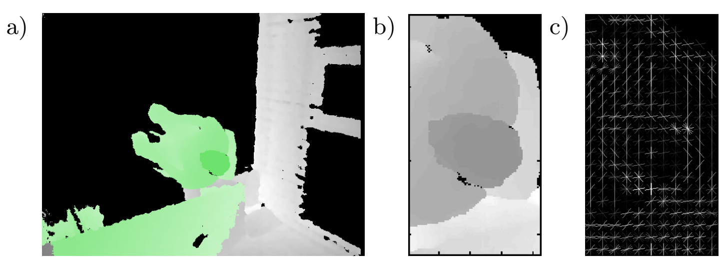

Fig. 3. Illustrative oversegmentation of the person blob a), extracted head b) and corresponding HOD c).

4 Feature extraction

At the beginning of this Section we show how we indicate falls on the basis of motion data. Afterwards, we discuss lying pose recognition in depth maps acquired by the overhead depth sensor. Finally, we present features describing dynamic transitions.

4.1 Fall indicating using body-worn accelerometer

Compared to vision-based motion analysis systems, wearable sensors offer several advantages, particularly in terms of cost, ease of use and, most importantly, portability. They are the only sensors that are used in real fall detection systems as well as outside of laboratory. However, despite many advantages, the inertial sensors-based technology does not meet the seniors' needs, because some activities of daily living are erroneously reported as falls. Present smartphones serve not only as communication and computing devices, but they also come with a rich set of embedded sensors, such as accelerometer, gyroscope and digital compass. Therefore, increasing interest on using this technology for fall detection is observed and the number of relevant papers grows considerably. Being aware of shortcomings of current solutions we believe that such technology will be significantly enhanced and in combination with small devices like smart watches it will be very useful in fall detection. Thus, our system processes data from a wireless body-worn accelerometer.

A lot of different techniques for inertial sensors were proposed to achieve reliable fall detection [5]. Frequently, a single body-worn sensor (tri-axial accelerometer or gyroscope, or both embedded in an IMU) is used to indicate person fall. Tri-axial accelerometer is the most commonly used device. The accelerometer-based algorithms raise the alarm when the signal reaches a certain threshold value. In [11] an accelerometer-based algorithm, relying on change in body orientation has been proposed. It signals potential fall if the root sum vector of the three squared accelerometer outputs exceeds an assumed threshold.

In our algorithm a fall is indicated if the Total Sum Vector SV_{total} is greater than 2.5 g. The value of SV_{total} has been calculated in the following manner:

$$SV_{total}(t) = \sqrt{A_x^2(t) + A_y^2(t) + A_z^2(t)} \tag{1}$$

where $A_x(t)$, $A_y(t)$, $A_z(t)$ is the acceleration in the $x-$, $y-$, and $z-$axes at time t, respectively. The SV_{total} contains both the dynamic and static acceleration components, and thus it is equal to 1 g for standing. The sensor signals were acquired at a frequency of 256 Hz and resolution of 12 bits. A survey of the relevant literature reveals that for a single inertial device the most valuable information can be obtained for devices attached near the centre of subject mass. Therefore, the accelerometer was attached near the spine on the lower back using an elastic belt around the waist.

4.2 Lying pose recognition

The lying pose has been distinguished from ADLs using classifiers trained on features representing the extracted person in the depth maps. We selected 214 maps from UR Fall Detection (URFD) dataset[1] with normal activities like walking,

[1] http://fenix.univ.rzeszow.pl/~mkepski/ds/uf.html

sitting down on a chair, taking or putting an object from floor, bending right. Such representative images were then used to train a k-NN classifier and a linear SVM classifier responsible for checking whether a person is lying on the floor. Both classifiers have been trained on three features:

- H/H_{max} - a ratio of head-floor distance to the height of the person
- $area$ - a ratio expressing the person's area in the image to the area of the blob at assumed distance to the camera, representing a top-view of the person
- l/w - a ratio of major length to major width of a blob representing the person on the depth image.

4.3 Dynamic transitions for fall detection

Person fall entails an abrupt and significant change of head-floor distance with accompanying change from a vertical orientation to horizontal one. The distance of the person's centroid to the floor also changes significantly and rapidly during the accidental fall period. In the images acquired from a ceiling-mounted camera the area ratio also changes considerably in the case of the fall. Thus, through analysis of the cues above mentioned we can determine whether a transition of the body is intentional or not.

In order to incorporate the information about the speed of the head towards the floor we utilize the following ratio:

$$h(t) = \frac{H(t)}{H(t - \Delta T)} \tag{2}$$

where $H(t)$ is determined in the moment of the impact, and $H(t - \Delta T)$ is calculated ΔT before the fall. It quite reliably characterizes the dynamics of the fall using a ceiling-mounted Kinect. In the depth images from an overhead camera the peak value of $H(t)/H(t-\Delta T)$ is far below one. The ratio $H(t)/H(t-\Delta T)$ can also be determined by analysis of depth image pairs and searching for some local minima, which are below a threshold value. However, the use of accelerometer as indicator of the potential fall simplifies calculation of this ratio since the time t can be determined easily and with low computational cost. Figure 4 demonstrates sample plots of $H(t)/H(t - \Delta T)$ for accidental fall and intentional lying on the floor. As we can observe, for ΔT equal to 600 ms the `Threshold` that is set to 0.6 has been exceeded for the fall.

If the lying pose classifier has decided that person is lying on the floor, the threshold-based classifier is executed to examine if the dynamic transition of the head in the depth maps is smaller than 0.6. Thus, the final decision about the fall is taken by the chain of the classifiers. The discussed chain of the classifiers consists of accelerometer based classifier deciding about the potential fall, as well as lying pose and dynamic transition classifiers, which take decisions on the basis of the depth maps. The resulting fall detection classifier has lower false positive rate.

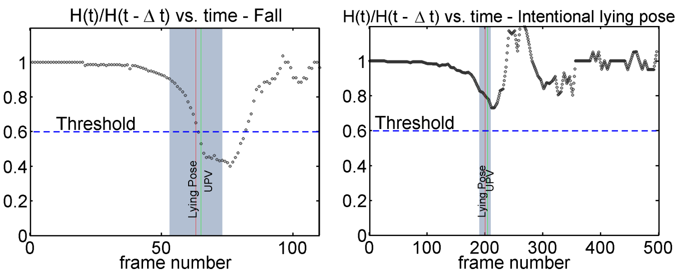

Fig. 4. $H(t)/H(t - \Delta T)$ vs. time for fall and intentional lying pose.

5 Real-time data processing

The fall detection system runs under Linux operating system. The application executes five main concurrent processes that communicate via message queues. The message queues provide asynchronous communication between processes. The first process is accountable for acquiring motion data from the wearable device, the second one acquires depth maps from the Kinect, third process is responsible for preprocessing inertial and depth data, fourth one is accountable for person tracking and camera control, whereas the fifth process is responsible for data classification and triggering the fall alarm. The dual-core processor of the utilized PandaBoard allows parallel execution of acquisition and processing processes. The motion data are acquired with 256 Hz and transmitted wirelessly via Bluetooth to the processing device. The depth images are acquired using OpenNI (Open Natural Interaction) library with a frame rate up to 30 fps.

The region growing is executed in 37 ms, on average. This means that the person detection and tracking can be done with 25 Hz. The SVM-based classifier for distinguishing between falls and non-falls has been trained off-line on a PC. It has been trained using LIBSVM software [12]. The SVM model obtained in such a way has been used to implement the fall predictor, executed on the PandaBoard. The principle behind k-NN methods is to seek a predefined number of training samples closest in a distance to the classified example, and then to predict the label from these. This means that the k-NN algorithm does not learn anything from the training data and just utilizes the training data itself for the classification. As a result, there is no learning cost and all the cost is dedicated to determining the decision. A naive neighbor search implementation involves the brute-force computation of distances in D dimensions between the current example and all instances in the dataset. To handle computational burden of the brute-force neighbor search, the kd-tree data structure has been used to store the ADL and fall examples. Given such a data structure determined in advance during training, the nearest neighbor of an example in question can be determined with only $O(\log(N))$ distance computations.

6 Experimental results

A system for fall detection should be inexpensive, ought to preserve user's privacy, work any time, and in particular it should exhibit both high sensitivity and specificity. The discussed system has been designed to fulfill the requirements mentioned above. A high detection accuracy has been achieved through scrupulous selection of the ingredients of the system as well as arrangement of scenarios for training and evaluation. In the subsequent subsections we present the dataset, evaluation results of the fall detector, and performance of person detector and tracker.

6.1 Fall detection dataset

The fall detector has been trained and evaluated on image sequences from URFD dataset. The dataset contains both depth images and acceleration data acquired by body-worn accelerometer [13]. The depth data were acquired by a ceiling-mounted Kinect Xbox 360 with 30 fps, whereas the motion data were acquired by x-IMU device with sampling rate of 256 Hz. The motion data contains the acceleration over time in the $x-$, $y-$, and $z-$axes together with the precalculated SV_{total} values. The x-IMU device was worn near the pelvis. All depth images are synchronized with motion data. Thirty simulated falls were recorded by two static Kinect devices. The first Kinect was situated in front of the scene and placed at the height of about 1 m, whereas the second ceiling-mounted Kinect was placed at the height of 2.6 m. The dataset recorded by the ceiling-mounted Kinect contains thirty image/acceleration sequences with 30 falls. The total number of the frames amounts to 3000. Two kinds of falls were simulated by five persons, namely from standing position and from sitting on the chair. The part of dataset, which was recorded by the frontal Kinect additionally contains 40 image/acceleration sequences with typical ADLs sitting down, crouching down, picking-up an object from the floor, and ten sequences with fall-like activities as quick lying on the floor and lying on the bed/couch.

6.2 Evaluation of the fall detector

The fall detection system can be configured to utilize both accelerometric and depth data or depth data only. When the accelerometer is involved at indicating that the person's movement is above some preset threshold, the impact can be detected more reliably in comparison to configuration using only depth data. The results are better since the depth map analysis is used to confirm the fall detection hypothesis indicated on the basis of examination of acceleration data from body-worn device. The use of the accelerometer as an indicator of the potential fall simplifies also the extracting of the dynamic features since the time of the impact can be determined easily, see also (2).

At the beginning we conducted experiments to determine acceleration thresholds for fall detection, using accelerometric measurements. Two voluntary subjects performed typical daily activities consisting in walking, taking or putting

Table 1. Performance of lying pose detection on depth maps from URFD dataset [%].

			True		
			Fall	No Fall	
Estimated	SVM	Fall	244	9	Accuracy=97.52%
		No fall	4	268	Precision=96.44%
			Sens.= 98.39%	Spec.= 96.75%	
	k-NN	Fall	244	10	Accuracy=97.33%
		No fall	4	267	Precision=96.06%
			Sens.= 98.39%	Spec.= 96.39%	

an object from floor, bending right or left to lift an object, sitting down on a chair, tying laces, crouching down and lying. Simulated falls (forward, backward, and lateral) were performed in an office towards a carpet with thickness of about 2 cm. The accelerometer was worn near the spine on the lower back using an elastic belt around the waist. For the carried out daily activities during half an hour experiment the acceleration values $2.5 - 3$g were exceeded several times. Hence, large amounts of false alarms would be triggered if the fall detection was carried out only on the basis of the acceleration data. One of the conclusions from the discussed experiments is that the acceleration threshold set to 2.5 allows us to indicate all fall and fall-like activities as non-ADLs.

The algorithm for lying pose recognition has been evaluated on 875 representative images from UR Fall Detection Dataset of which 60% were training examples. Since the discussed sequences were recorded using the static sensor, the person has been delineated through the differencing the current depth map from the depth reference map [7]. The depth reference maps were extracted in advance. A linear SVM and a k-NN with 5 neighbors classifiers have been trained on features discussed in Section 4.2 to discriminate between falls and ADLs. In Tab. 1 are shown results, which were obtained by the classifiers mentioned above. As we can notice, the results achieved by k-NN and SVM-based lying pose detectors are identical. They are promising in terms of both sensitivity and accuracy.

Afterwards, we evaluated the usefulness of the dynamic feature. The experiments aimed at improving the distinguishing between the intentional lying on the floor from the accidental falling. At the beginning we investigated the classification accuracy with regard to decision criterions. Figure 5 depicts the receiver operating characteristic (ROC) curve for the dynamic feature, which illustrates the performance of a binary classifier for varying discrimination threshold. The best classification accuracy has been obtained for ΔT=500 ms and `Threshold`=0.525. Two students attending in the evaluations found that a cascade classifier consisting of lying pose detector and dynamic transition detector has almost null ratio of false alarms. They found that the main reason of the false alarms is imperfect detection of the moment of the body impact. The cascade classifier combined with the accelerometer demonstrated null false alarm. In particular, all falls were detected properly on the images from URFD dataset.

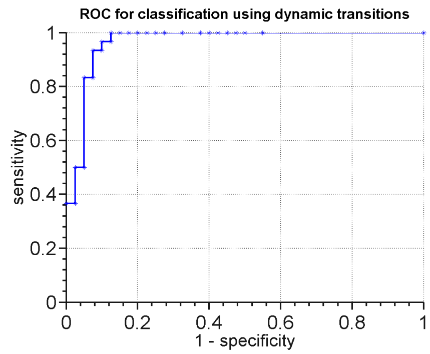

Fig. 5. Receiver operating characteristic (ROC) for dynamic feature.

6.3 Evaluation of person detector and tracker

When a static ceiling-mounted camera is used, the person can be extracted reliably and with low computational burden through differencing the current depth map from the accommodated on-line depth reference map of the scene [7]. On the basis of such a depth reference map the person can be extracted in about 50 ms. However, as mentioned previously, the observation area of ceiling-mounted Kinect is quite small. Thanks to pan-tilt capabilities the monitoring area of the system can be extended considerably. On the other hand, more sophisticated and time consuming techniques are needed to delineate the person followed by an active camera.

Initially, the region growing has been evaluated in depth maps from URFD dataset. In all frames the person was extracted satisfactorily. Afterwards, the region growing has been evaluated on five sequences acquired by the active camera. In the discussed experiments with active camera, the aim was not only to extract in real-time the person, but also to keep he/she in the central part of the depth maps. Prior to the delineation of the subject using region growing, the camera was static for a while, and he/she was initially extracted through differencing the current depth map from the depth reference map of the scene. In all frames from the utilized sequences, including maps with intentional falls, all main body parts were extracted. Figure 6 depicts some results with delineated person, which were obtained on depth images acquired by the active camera. As we already mentioned, on the PandaBoard the average time of person delineation using region growing is from 35 ms to 40 ms depending on the blob size. Sample video illustrating person tracking with the pan-tilt depth camera can be found at the following url: `http://fenix.univ.rzeszow.pl/~mkepski/demo/act.mp4`.

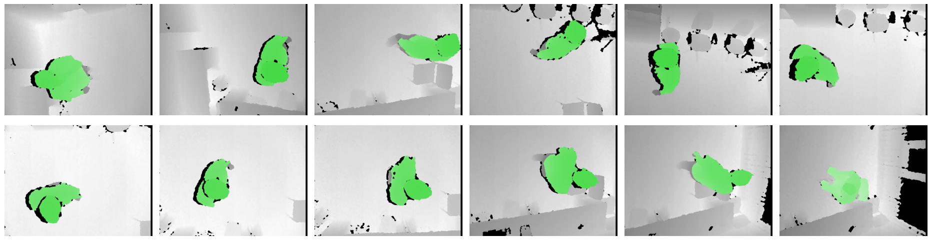

Fig. 6. Region growing - based person delineation on depth maps.

The person detector has been evaluated on 254 positive samples and 638 negative samples of which 60% were used for training. The images with delineated person were scaled according to distance of his/her head to the camera. They were also rotated to a canonical pose using the axis of the person's blob. Table 2 shows results that were obtained using the detector discussed in Section 3.2. As we can observe, the results are better if the silhouettes are rotated to the canonical pose. On the other hand, the difference is not significant, and this means that the algorithm is quite resistant to various head poses. This is because the gradients on the head in depth images seen from an overhead camera form elliptical like structures. The discussed results were obtained for HOD cell size equal to 8×8. On the PandaBoard a single person detection can be done in 41 ms.

Table 2. Performance of person detection [%].

	accuracy	precision	sensitiv.	specificity
rotat.	99.45	98.21	100.0	99.22
no rotat.	98.91	98.18	98.18	99.22

As we mentioned in Subsection 3.2, in some circumstances, during person extraction on the basis of the region growing a person oversegmentation can take place. In such situations the person detector has been found to be useful since it helps to authenticate location of the person in such oversegmented depth maps. Let us consider a fall scenario, where a person is seating at the chair and is oversegmented as demonstrated on Fig. 3a. After the fall we have time t that is indicated by the threshold-based trigger. We then fetch a depth image from the circular buffer, which had been acquired 600 ms earlier. For simplicity, let us assume that this is image depicted on Fig. 3a. Given such a depth image, we execute HOG-SVM head detector to determine the location of the centroid of the head blob. After a while, i.e. max after a few seconds, we can determine $H(t)/H(t - \Delta T)$. Even if after the fall the person is still oversegmented, we can determine $H(t)$ easily because a lying person occupies areas below 40 cm from the ground. This way, we avoid to perform fall detection on features not belonging to person undergoing monitoring.

7 Conclusions

In this paper we presented an embedded system for fall detection using a ceiling-mounted depth sensor. In the proposed architecture a body-worn accelerometer is utilized to indicate an eventual fall, whereas the Kinect sensor is used to authenticate it. In order to expand the observation area the Kinect has been mounted on a pan-tilt head. We demonstrated that owing to thresholding of the motion data we can considerably reduce the computing overheads for processing the depth data. We then showed that a depth sensor can reliably distinguish between such filtered events and the falls. In consequence, the depth maps are not processed frame-by-frame, but instead a circular buffer is used to store the depth maps for processing them in the case of possible fall. The detection performance of the system has been evaluated on UR Fall Detection dataset. The presented embedded system works 365/7/24 days and hours, permits reliable and unobtrusive fall detection as well as preserves privacy of the user.

Acknowledgment. This work was supported by Polish National Science Center (NCN) under a research grant 2014/15/B/ST6/02808 and a grant 11.11.230.124.

References

1. Friedewald, M., Raabe, O.: Ubiquitous computing: An overview of technology impacts. Telemat. Inf. **28**(2) (May 2011) 55–65
2. Wilson, C., Hargreaves, T., Hauxwell, R.: Smart homes and their users: a systematic analysis and key challenges. Pers. Ubiquitous Comp. **19** (2015) 463–476
3. Acampora, G., Cook, D., Rashidi, P., Vasilakos, A.: A survey on ambient intelligence in healthcare. Proceedings of the IEEE **101**(12) (Dec 2013) 2470–2494
4. Webster, D., Celik, O.: Systematic review of Kinect applications in elderly care and stroke rehabilitation. Journal of NeuroEngineering and Rehabilitation **11** (2014)
5. Bourke, A., O'Brien, J., Lyons, G.: Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm. Gait & Posture **26**(2) (2007) 194–199
6. Rougier, C., Auvinet, E., Rousseau, J., Mignotte, M., Meunier, J.: Fall detection from depth map video sequences. In: LNCS. Volume 6719., Springer (2011) 121–128
7. Kepski, M., Kwolek, B.: Fall detection on embedded platform using Kinect and wireless accelerometer. In: Proc. of ICCHP, LNCS, Springer (2012) II:407–414
8. Cuppens, K., Chen, C.W., Wong, K.B.Y., Van de Vel, A., Lagae, L., Ceulemans, B., Tuytelaars, T., Van Huffel, S., Vanrumste, B., Aghajan, H.: Integrating video and accelerometer signals for nocturnal epileptic seizure detection. In: Proc. of the 14th ACM Int. Conf. on Multimodal Interaction, New York, NY (2012) 161–164
9. Chen, C., Jafari, R., Kehtarnavaz, N.: Improving human action recognition using fusion of depth camera and inertial sensors. IEEE Trans. HMS **45** (2015) 51–61
10. Spinello, L., Arras, K.: People detection in RGB-D data. In: IEEE/RSJ Int. Conf. on IROS. (Sept 2011) 3838–3843
11. Chen, J., Kwong, K., Chang, D., Luk, J., Bajcsy, R.: Wearable sensors for reliable fall detection. In: Int. Conf. on Eng. in Medicine and Biology Soc. (2005) 3551–3554
12. Chang, C.C., Lin, C.J.: LIBSVM: A library for Support Vector Machines. ACM Trans. Intell. Syst. Technol. **2**(3) (May 2011) 1 – 27
13. Kepski, M., Kwolek, B.: Fall detection using ceiling-mounted 3d depth camera. In: Int. Conf. on Computer Vision Theory and Appl. (VISAPP). 2:640–647