A Neurocognitive Robot Assistant for Robust Event Detection

German I. Parisi and Stefan Wermter

Abstract Falls represent a major problem in the public health care domain, especially among the elderly population. Therefore, there is a motivation to provide technological solutions for assisted living in home environments. We introduce a neurocognitive robot assistant that monitors a person in a household environment. In contrast to the use of a static-view sensor, a mobile humanoid robot will keep the moving person in view and track his/her position and body motion characteristics. A learning neural system is responsible for processing the visual information from a depth sensor and denoising the live video stream to reliably detect fall events in real time. Whenever a fall event occurs, the humanoid will approach the person and ask whether assistance is required. The robot will then take an image of the fallen person that can be sent to the person's caregiver for further human evaluation and agile intervention. In this paper, we present a number of experiments with a mobile robot in a home-like environment along with an evaluation of our fall detection framework. The experimental results show the promising contribution of our system to assistive robotics for fall detection of the elderly at home.

1 Introduction

Falls represent a major concern in the public health care domain, especially among the elderly population. According to the World Health Organization, fall-related injuries are common among older persons and represent the leading cause of pain, disability, loss of independence and premature death [1]. Although fall events do not necessarily cause a fatal injury, fallen people may be unable to get up without

G.I. Parisi (✉) · S. Wermter
Department of Informatics, Knowledge Technology, University of Hamburg, Vogt-Koelln-Strasse 30, 22527 Hamburg, Germany
e-mail: Parisi@informatik.uni-hamburg.de
URL: http://www.informatik.uni-hamburg.de/WTM/

S. Wermter
e-mail: wermter@informatik.uni-hamburg.de

K.K. Ravulakollu et al. (eds.), *Trends in Ambient Intelligent Systems*,
Studies in Computational Intelligence 633, DOI 10.1007/978-3-319-30184-6_1

assistance, thereby resulting in "long lie" complications such as hypothermia, dehydration, bronchopneumonia, and pressure sores [2]. Moreover, fear of falling has been associated with a decreased quality of life, avoidance of activities, and mood disorders such as depression (among fallers and non-fallers) [3].

As a response to increasing life expectancy, plenty of research has been done to provide technological solutions for supporting living at home and smart environments for assisted living. The motivation of assistive fall systems is the ability to promptly report a fall event and by this enhancing the person's safety perception and avoiding the loss of confidence due to functional disabilities. Recent systems for elderly care aim mostly to detect hazardous events such as falls and allow the monitoring of physiological measurements (e.g. heart rate, breath rate) with the use of wearable sensors to detect and report emergency situations in real time [4, 5]. Vision-based fall detection is currently the predominant approach due to the constant development of computer vision techniques that yield increasingly promising results in both experimental and real-world scenarios. Additionally, in the last half decade the advent of low-cost depth-sensing devices such as the Microsoft Kinect [6] and ASUS Xtion Live [7] has led to a great number of vision-based applications using depth information instead of, or in combination with, color information. In this setting, the use of machine learning and neural network approaches has been shown to be an appropriate methodology to achieve knowledge generalization of a set of training activities for the classification of unseen situations [8], and the detection of abnormal behaviors such as fall events in domestic environments [9].

Contrary to fixed sensors, mobile assistive robots may be designed to process the sensed information and undertake actions that benefit people with disabilities and seniors in a residential context. There exists an increasing number of ongoing research projects using assistive robotics in smart environments to provide tools for self-care, independence at home, and telematic diagnosis. Moreover, advanced robotic technologies may encompass socially-aware assistive solutions for interactive robot companions, able to support basic daily tasks of independent living and enhance user experience through human-robot interaction (e.g. dialogues and vocal commands). Recent studies support the idea that the use of socially assistive robots leads to positive effects on the senior's well-being in domestic environments [10]. On the other hand, the use of robotic technologies brings a vast set of challenges and technical concerns.

In this work, we introduce a humanoid robot assistant that monitors a person in a household environment and reports abnormal user behavior such as a fall event. The underlying motivation is that the robot keeps the person in the scene while he/she performs daily activities, thereby anonymously tracking the user's position, body posture, and motion characteristics. The processed visual information is fed into the neural system which is responsible for triggering alarms in case a fall is detected. Whenever a fall event occurs, the humanoid will approach the user and ask whether assistance is required. The robot will then take a picture of the scene that can be sent to the user's caregiver or relatives for telematic evaluation.

This chapter is organized as follows. In Sect. 2, we provide an overview on the state of the art in fall detection, in particular vision-based approaches using depth sensors and assistive robotics. In Sect. 3, we introduce our learning-based neural framework for detecting abnormal events. We show experiments in a home-like environment and an evaluation of the system for a person falling down or crawling. In Sect. 4, we present an assistive humanoid robot for detecting fall events in a domestic scenario. We first depict an overview of our system and then go into detail about the software, hardware, and the communication interface. We conclude in Sect. 5 with a discussion on open issues in fall detection, trends and challenges for assistive robots, and future work directions for *aging at home* systems.

2 Trends in Fall Detection

Broadly speaking, a fall detection system can be defined as an assistive service with the main goal to promptly report a fall event. From a technical perspective, this service represents a pervasively challenging task in real-world scenarios in terms of reliability and robustness, since it raises a vast set of issues and technological concerns. As reported by an extensive number of works in the literature, fall detection systems may be designed, implemented, and evaluated on the basis of a manifold of approaches using different types of sensing devices and methodologies to process the sensed information.

The purpose of this section is to provide a concise overview of the state of the art in fall detection technologies with a particular focus on vision-based approaches, the developing use of low-cost depth sensing devices for 3D tracking, and emerging technologies in assistive robotics for aging at home and telematic caregiving.

2.1 Fall Detection Systems

There seems to exist an agreed taxonomy in the literature that classifies fall detection systems into two main categories according to the type of sensor used to monitor the user: wearable-based and ambient-based approaches [11, 12].

Wearable-based approaches relate to the use of small electronic devices that can be worn by the user, for instance, on top of clothing or as accessories. The most extensively used wearable devices consist of accelerometers and gyroscopes attached to the body that measure the user's location and motion. There is a vast number of applications that use these measurements to evaluate the user's gait and balance, and assess the risk of a fall event [13–16]. In the last years, this trend has seen a significant boost due to the availability of low-cost sensors embedded in smartphones [17–20]. On the other hand, ambient-based approaches relate to the use of sensing devices deployed in the environment, thereby not requiring the user to wear any sensor. Fall detection systems of this kind, also referred to as non-intrusive and context-aware, may encompass a wide spectrum of sensor types such as cameras, microphones, pressure and floor sensors [21, 22].

We focus on the use of cameras for vision-based fall detection with increasingly promising results in both experimental and real-world scenarios. Lee and Mihailidis [23] presented a vision-based method with a ceiling camera for monitoring falls at home. The authors considered falls as lying down in a stretched or tucked position. The system accuracy was evaluated with a pilot study using 21 subjects consisting of 126 simulated falls. Personalized thresholds for fall detection were based on the height of the subjects. The system detected fall events with 77 % accuracy and had a false alarm rate of 5 %. Miaou et al. [24] presented a customized fall detection system using an omni-camera for capturing 360° scene images. Falls were detected based on the change of the ratio of people's height and width. Two scenarios were used for the detection: with and without considering user health history, for which the system showed 81 and 70 % accuracy respectively. Rougier et al. [25] presented a method for fall detection by analysing human shape deformation in video sequences. Falls were detected from normal activities using a Gaussian mixture model with 98 % accuracy. The overall system performance increased when taking into account the lack of significant body motion after the detected fall event. Liu et al. [26] detected falls considering privacy issues, thereby processing only human silhouettes without featural properties such as the face. A k-nearest neighbor (kNN) algorithm was used to classify the postures using the ratio and difference of a body silhouette bounding box. Recognized postures were divided into three categories: standing, temporary transitional, and lying down. Experiments with 15 subjects showed a detection accuracy of 84.44 % on fall and lying down events.

In a multi-camera scenario, Cucchiara et al. [27] presented a vision system with multiple cameras for tracking people in different rooms and detecting falls based on a hidden Markov model (HMM). People tracking was based on geometrical and color constraints and then sent to the HMM-based posture classifier. Four main postures were considered: walking, sitting, crawling, and lying down. When a fall was detected, the system triggered an alarm via SMS to a clinician's PDA with a link to live low-bandwidth video streaming. Experiments showed that occlusions had a strong negative impact on the system's performance. Hazelhoff et al. [28] detected falls using two fixed perpendicular cameras. The foreground region was extracted from both cameras and the principal components (PCA) for each object were computed to determine the direction of the main axis of the body and the ratio of the variances. Using these features, a Gaussian multi-frame classifier was used to recognize falls. In order to increase robustness and mitigate false positives, the position of the head was taken into account. The system was evaluated also for partially occluded people. Experiments showed real-time performance with an 85 % overall detection rate.

In contrast to the use of color cameras, Diraco et al. [29] addressed the detection of falls and the recognition of several postures with 3D information. The system used a fixed time-of-flight camera that provided robust measurements under different illumination settings. Moving regions with respect to the floor plane were detected applying a Bayesian segmentation to the 3D point cloud. Posture recognition was carried out using the 3D body centroid distance from the floor plane and the estimated body orientation. The system yielded promising results on synthetic data with threshold-based clustering for different centroid's height thresholds.

An enduring bottleneck for vision-based approaches is the segmentation of human shape from acquired 2D image sequences, which is often constrained in terms of computational effort and robustness to illumination changes. Recent research work has indicated a trend towards fall detection systems using 3D sensing devices for more accurate and efficient estimations of human motion and body posture.

2.2 3D Human Tracking

In the last half decade, the emergence of low-cost depth sensing devices such as the Microsoft Kinect [6] and ASUS Xtion Live [7] has led to a great number of vision-based applications using depth information instead of, or in combination with, color information. This prominent sensor technology provides depth measurements used to obtain reliable estimations of 3D human motion in cluttered environments, including a set of body joints in real-world coordinates and their orientations. As shown by a broad number of recent applications for human action recognition, this sensor trend represents a significant contribution to overcome a set of limitations related to traditional 2D sensors (e.g. RGB cameras), thereby increasing robustness under varying illumination conditions and reducing computational effort for motion segmentation and body pose estimation. Depth sensors have the additional advantage of avoiding privacy issues regarding the identity of the monitored person, since color information is not required at any stage. An extensive review of the depth sensor Kinect and its application to diverse research fields, e.g. action recognition and navigation, was presented by Han et al. [30].

A combination of computational efficiency, robustness to light changes in indoor environments, and lower cost factors have made fall detection systems using depth information increasingly popular in the research community. Rougier et al. [31] used 3D information from a depth sensor to estimate a person's centroid height and velocity relative to the ground plane. Thresholds on ground distance and velocity computed from training data were used to detect fall events also with occluded persons (e.g. fallen down behind a sofa). The system was evaluated on simulated falls and normal activities (e.g. walking, sitting down, crouching) with an overall success rate of 98.7 %. Planinc and Kampel [32] used depth information to compute a body axis that described the overall orientation of a person. Thresholds for similarity to the ground and the height were used to distinguish falls from other daily activities. The system was evaluated on a dataset of 72 video sequences containing 40 falls with accuracy of 95 % after eliminating tracking errors. In this approach, occlusions were not considered. Mastorakis and Makris [33] presented a depth-based fall detection system taking into account body velocity and inactivity periods. The velocity was measured on the basis of the contraction or expansion of a 3D bounding box built around the person's body. The detection algorithm was designed as a Boolean decision tree for distinguishing falls from other actions. Good results were obtained from different sensor perspectives (frontal, side) on a customized dataset.

Approaches using depth information in combination with machine learning and neural networks have shown to provide promising results. Zhang et al. [34] presented a depth-based system to recognize different types of falls, i.e. fall from a standing position and fall from a chair. Body features such as structure similarity and height were extracted from a kinematic model and fed to a hierarchical Support Vector Machine (SVM) classifier. Promising results for detecting falls from other three daily actions (i.e. standing, sitting on a chair or on the floor) were obtained on a dataset of 200 video sequences with different light conditions. Parisi and Wermter [9] presented a neural network approach to detect abnormal behaviors such as falling, fainting, and crawling while monitoring domestic daily actions. A self-organizing neural architecture was trained on a set of domestic actions (e.g. walking, sitting, picking up objects) from body features such as velocity and orientation. The system detected abnormal behavioral patterns not shown during the training phase in two different tracking scenarios with fixed and mobile depth sensors. Best results were obtained by automatically detecting and removing tracking errors.

In contrast to most of the approaches using the depth sensor positioned parallel to the horizontal surface, Gasparrini et al. [35] detected falls using a ceiling sensor. A segmentation algorithm was used to extract blobs in the scene and track human silhouettes on the basis of several anthropometric relations. Falls were detected for a tracked person under a threshold-based distance to the floor. Experiments showed promising results also for scenarios with more persons present in the top-view scene.

While the number of advantages introduced by low-cost depth sensors is significant in terms of body motion and posture estimation, these approaches lead to issues that may prevent them from operating in real-world environments. For instance, their operation range (distance covered by the sensor) is quite limited (between 0.8 and 5 m), as well as their field of view (see Table 1 for details), thereby requiring a mobile or multi-sensor scenario to monitor an extensive area of interest.

Table 1 ASUS Xtion Live sensor specifications [7]

Depth image size	VGA (640 × 480): 30 fps, QVGA (320 × 240): 60 fps
Field of view	58 H, 45 V, 70 D (horizontal, vertical, diagonal)
Distance of use	0.8–3.5 m
Dimensions	18 × 3.5 × 5 cm
Power consumption	Below 2.5 W
Interface	USB 2.0/3.0
Weight	227 g

2.3 Assistive Robotics

Mobile robots have been characterized by a constant development for "aging at home" scenarios. In contrast to fixed sensors, mobile assistive robots may be designed to process the sensed information and undertake actions that benefit people with disabilities and seniors in a residential context. In fact, the mobility of robots represents a big benefit for non-invasive monitoring of users, thereby better addressing fixed sensors' limited field of view, blind spots, and occlusions.

There has been an increasing number of ongoing research projects using assistive robotics in smart environments to provide tools for self-care, independence at home, and telematic diagnosis. Advanced robotic technologies may encompass socially-aware assistive solutions for interactive robot companions able to support basic daily tasks of independent living and enhance user experience through flexible human-robot interaction (e.g. dialogues, vocal commands). A number of experimental studies support the idea that the use of socially assistive robots implies positive effects on the seniors' well-being in domestic environments [10]. Examples of recent and current interdisciplinary research projects using interactive mobile robots for aging in place include: Cogniron (Cognitive Robot Companion) [36], LIREC: Living with robots and interactive companions [37], Hermes: Cognitive Care and Guidance for Active Ageing [38], KSERA (Knowledgable SErvice Robots for Aging) [39], GiraffPlus [40], ROBOT-ERA [41], and Accompany (ACceptable robotics COMPanions for AgeiNg Years) [42]. Despite different functional perspectives concerning elderly care and user needs (e.g. rehabilitation [39], robot companions [42]), there is a strong affinity regarding the intrinsic challenges and issues needed to operate these systems in real-world scenarios. In fact, the use of mobile robots may be generally combined with ambient sensors embedded in the environment (e.g. cameras, microphones) to enhance the agent's perception and increase robustness under real-world conditions. On the other hand, complementary research efforts have been conducted on the deployment of stand-alone mobile robot platforms able to sense and navigate the environment by relying exclusively on onboard sensors.

Specifically for fall detection, promising experimental results have been obtained by combining mobile robots and 3D information from depth sensors. This approach overcomes limitations in the operation range of sensors while preserving reduced computational power for real-time characteristics. Mundher and Zhong [43] proposed a mobile robot with a Kinect sensor for fall detection based on floor-plane estimation. The robot tracks and follows the user in an indoor environment, and can trigger an alarm in case of a detected fall event. The system recognizes two gestures to start and stop a distance-based *user-following* procedure, and three voice commands to enable/disable fall detection, and call for help. The robot is provided with a mobile phone to send notifications via SMS or emergency call if the user does not recover from a fall within five seconds. Volkhardt et al. [44] presented a mobile robot to detect fallen persons, i.e. a user already lying on the floor. The system segments objects from the ground plane and layers them to address partial occlusions. A classifier trained on positive and negative examples is used to detect object layers as a fallen human. Experiments reveal that the overall accuracy of the system is strongly dependent on the type of extracted features and the classifier.

Additional challenges conveyed by the use of mobile robots for detecting fall events regard the tolerance of noise in a moving sensor scenario [9], the robust tracking of occluded persons [45], and effective navigation strategies for following and finding people in domestic environments [46].

3 Learning-based Abnormal Event Detection

Despite extensive research efforts promoted by advanced computer vision techniques and recent low-cost sensor trends, the question remains open on how to better process extracted body features for effectively extrapolating the complex dynamics of actions and fall events exhibiting noise tolerance and robustness in real-world scenarios. Indeed, the vast majority of the presented algorithms rely on domain-specific thresholds to distinguish falls from other activities, often being unable to operate under real-world conditions. On the other hand, learning-based paradigms such as machine learning and neural networks represent prominent tools to achieve knowledge generalization in a set of training activities for the subsequent classification of unseen situations [8, 9, 47]. In this setting, a possible approach for fall detection consists in learning a set of normal actions from training data and subsequently detecting events that do not conform to the expected behavior.

In this section, we present our work on abnormal event detection based on unsupervised neural network learning. The system consists of a hierarchical neural architecture that learns a set of normal actions, e.g. walking, sitting, and picking up objects, captured by a depth sensor. After the training phase, the system will report novel behavioral patterns, e.g., fall event, as abnormal actions and trigger an alarm. To contrast tracking errors and sensor noise, the neural architecture is also responsible for automatically removing noisy samples from the extracted body features. We report a number of experiments in a home-like environment that show our system can detect fall events with high accuracy in real time.

3.1 Feature Extraction

The first stage of our system consists of the extraction of body action features from 3D motion information captured by a depth sensor. We estimate the position of a moving target based on a model of the human skeleton. In previous work [9], we used this skeleton-based representation to compute body centroids that describe actor-independent posture and motion features. Two centroids were estimated as the centers of mass that follow the distribution of the main body masses on each

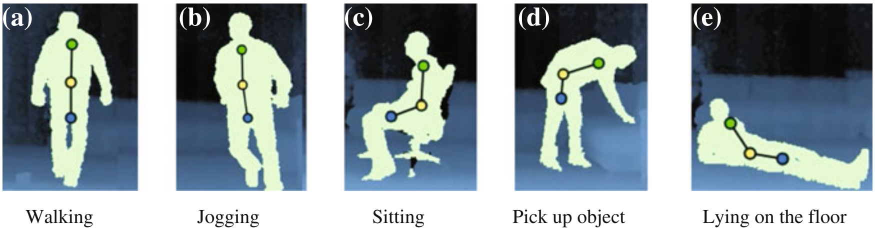

Fig. 1 Full-body representation for pose-motion extraction [47]. We estimate three centroids C_1 (*green*), C_2 (*yellow*) and C_3 (*blue*) for upper, middle and lower body respectively. We compute the segment slopes (θ^u and θ^l) to describe the posture with the overall orientation of the upper and lower body

posture. This technique extrapolates significant motion characteristics while maintaining a low-dimensional feature space and increasing tracking robustness for situations of partial occlusions. We then extended our model to describe more accurately articulated actions by considering three body centroids [47]: C_1 for upper body with respect to the shoulders and the torso; C_2 for middle body with respect to the torso and the hips; and C_3 for lower body with respect to the hips and the knees. Each centroid is represented as a point sequence of real-world coordinates $C = (x, y, z)$. We compute upper and lower orientations θ^u and θ^l given by the slope angles of the segments $\overline{C_1C_2}$ and $\overline{C_2C_3}$ respectively. As shown in Fig. 1, θ^u and θ^l describe the overall body posture as the overall orientation of the torso and the legs, allowing to capture significant posture configurations of actions such as walking, sitting, picking up and lying down on the floor.

To estimate body motion, we compute the pixel difference $D_i = (d^x, d^y, d^z)$ between two consecutive frames of the upper centroid C_1 in the x, y, z direction. The upper centroid was selected based on the consideration that the torso orientation is the most characteristic reference during the execution of a full-body action [48]. We then estimate body velocity with respect to the sensor as

$$S_i = \left\{ \frac{d^x}{s}, \frac{d^y}{s}, \frac{d^z}{s} \right\}, \tag{1}$$

where $s = \sqrt{(d^x)^2 + (d^y)^2 + (d^z)^2}$.

We encode S_i as horizontal and vertical speed with respect to the image plane, respectively expressed as $h_i = \sqrt{(S_i^x)^2 + (S_i^z)^2}$ and $v_i = S_i^y$. The former refers to the target moving on the width and depth axis, i.e. closer, further, right, and left. The latter represents the speed with respect to height, e.g. negative if the target is moving down.

For each processed frame i, we obtain the following pose-motion vector:

$$F_i = \left(\theta_i^u, \theta_i^l, h_i, v_i\right). \tag{2}$$

This representation describes spatio-temporal properties of actions in terms of length-invariant, sequential vectors, particularly suitable for serving as input for neural network architectures.

3.2 Learning Framework

Unsupervised neural network learning has shown to be a prominent approach for the detection of abnormal events [49], also referred to as anomaly detection [50]. We propose a hybrid neural-statistical framework to approximate the normal behavior with trained self-organizing map (SOM) networks and subsequently detect behavioral patterns that do not conform to the expected learned behavior with an abnormality test.

The SOM is a competitive neural network introduced by Kohonen [51] that has shown to be a compelling approach for clustering motion expressed in terms of multi-dimensional flow vectors [52–55]. The proposed learning framework consists of three SOM networks. A first network Φ_0 is trained to detect outlier values from the extracted pose-motion vectors caused by tracking errors and sensor noise. After this initial learning phase, the pose-motion vectors are processed again to perform a threshold-based test and remove outliers from the training set. The denoised training set is then fed to a hierarchical SOM-based architecture composed of two networks, Φ_1 and Φ_2, for clustering the subspace of normal actions taking into account spatio-temporal relationships of action sequences. A flow chart of this learning stage is illustrated by Fig. 2.

At detection time, extracted vectors will be denoised and processed through the hierarchy of trained SOM networks. New observations that deviate from the learned

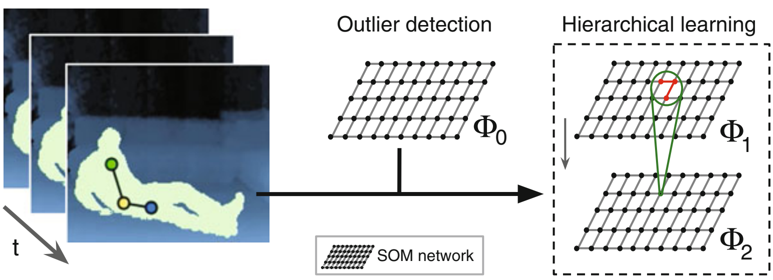

Fig. 2 Flow chart of our SOM-based learning stage. A first network Φ_0 is trained to detect and remove outliers from extracted pose-motion vectors. Preprocessed vectors are fed to a hierarchy of networks (Φ_1 and Φ_2) to cluster spatio-temporal relationships of action sequences

behavior, i.e. below an abnormality threshold, will be reported as abnormal. The detection of noise and abnormal behavior is based on the same abnormality test using two different automatically computed thresholds.

3.2.1 Training Algorithm

The traditional SOM is unsupervised and allows to obtain a low-dimensional discretized representation from high-dimensional input spaces. It consists of a layer with competitive neurons connected to adjacent units by a neighborhood relation. The network learns by iteratively reading each training vector and organizes the units so that they describe the domain space of input observations. Each unit j is associated with a d-dimensional model vector $m_j = [m_{j,1}, m_{j,2}, \ldots, m_{j,d}]$. For each input vector $x_i = (x_1, \ldots, x_n)$ presented to the network, the best matching unit (BMU) b for x_i is selected by the smallest Euclidean distance as

$$b(x_i) = \arg\min_j \|x_i - m_j\|. \tag{3}$$

For an input vector x_i, the quantization error q_i is defined as the distance of x_i from the BMU $b(x_i)$.

We consider two-dimensional networks with units arranged on a hexagonal lattice in the Euclidean space. Each competitive network is trained with a batch variant of the SOM algorithm. This iterative algorithm presents the whole data set to the network before any adjustments are made. The updating is done by replacing the model vector m_j with a weighted average over the samples:

$$m_j(t+1) = \frac{\sum_{i=1}^{n} h_{j,b(i)}(t) x_i}{\sum_{i=1}^{n} h_{j,b(i)}(t)}, \tag{4}$$

where b is the best matching unit (Eq. 3), n is the number of sample vectors, and $h_{j,b(i)}$ is a Gaussian neighborhood function:

$$h_{b,i}(x) = \exp\left(\frac{-\|r_b - r_i\|^2}{2\sigma^2(t)}\right), \tag{5}$$

where r_b is the location of b on the map grid and $\sigma(t)$ is the neighborhood radius at time t.

At the second learning stage step, a hierarchical SOM-based approach is used to learn spatio-temporal properties of action sequences from denoised training samples. We first train the network Φ_1 with pose-motion vectors (Eq. 2) from the denoised training set. After this training phase, chains of activated best matching units (Eq. 3) for ordered training sequences produce time varying trajectories on the network map. We empirically define a BMU trajectory for a training vector x_i as

$$\tau_i = (b(x_{i-2}), b(x_{i-1}), b(x_i)). \quad (6)$$

We denote the set of all activation trajectories X as $T(X)$. This step produces a time-selective mapping with action segments from 3 consecutive vectors.

3.2.2 Tracking Errors

An outlier can be seen as an observation that does not follow the pattern suggested by the majority of the observations belonging to the same data cloud [53]. From a geometrical perspective, outliers are to be found detached from the dominating distribution of the subspace of normal actions.

In our approach, we differentiate between outliers introduced by tracking errors and outliers caused by tracked abnormal events. For this purpose, we assume that the behavior of a moving target must be consistent over time. Therefore, we consider highly inconsistent changes in body posture and speed to be caused by tracking errors rather than actual tracked motion. As shown by our experiments, the presence of tracking errors in the training set may negatively affect the SOM-based clustering of pose-motion features. Figure 3 illustrates these effects after the learning phase. A first SOM was trained with the full set of extracted motion vectors, for which outliers in the data decreased the unfolding of the projected feature map (Fig. 3a). These noisy samples were detected by our algorithm and removed from the training set. As seen from the second SOM trained with the denoised training set (Fig. 3b), the absence of outliers allowed a more representative clustering of the motion vectors for the subspace of normal actions.

While we use the same algorithm to detect outliers, two different abnormality thresholds are automatically computed that take into account the different

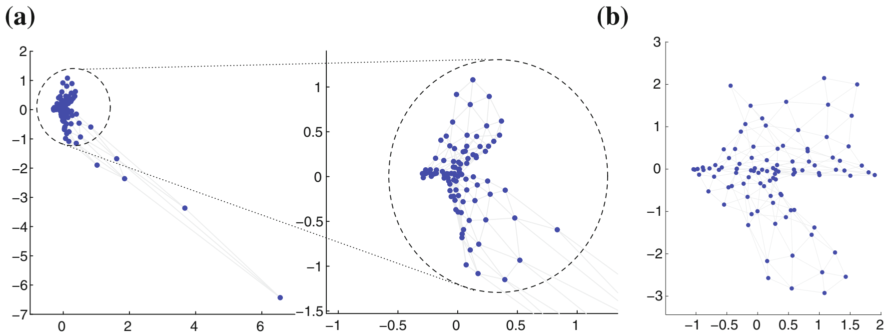

Fig. 3 Effects of outliers in the clustering of training data [9]. **a** The first SOM was trained with the full set of extracted motion vectors. The presence of highly noisy observations in the training set decreased the unfolding of the projected feature map. **b** This second SOM was trained after removing outliers from the training set, resulting in a more representative clustering of the observations from tracked motion

characteristics of tracking noise and abnormal pose-motion vectors. Using this first trained SOM network as reference, also tracking errors in the test set are detected and removed.

3.2.3 Abnormality Detection Algorithm

The goal of the detection algorithm is to test if the most recent observation is abnormal or not. For this purpose, the degree of abnormality for every test observation is expressed with the estimation of a P-value. If the P-value is smaller than a given threshold, then the observation is considered to be abnormal and reported as such.

For a given training set X and a new test observation x_{n+1} presented to the network Φ, the algorithm is summarized as follows [56]:

0. Compute the set of quantization errors $Q = (q_1, q_2, \ldots, q_n)$.
1. Compute $q_{(n+1)}$ with respect to Φ.
2. Define B as the number of quantization errors $(q_1, \ldots, q_n)$ greater than $q_{(n+1)}$.
3. Define the abnormality P-value as $P_{(n+1)} = B/n$.

As an extension of the algorithm proposed in [56], abnormality thresholds are automatically computed for the trained networks Φ_0 and Φ_2. The choice of convenient threshold values that take into account the characteristics of the distributions can have a significant impact on the successful rates for abnormality detection. From a neural network perspective, the threshold values will consider the distribution of the quantization errors from each trained SOM. Based on related research [9], we empirically define two different thresholds, T_O for outlier detection and T_A for abnormality detection:

$$T_O = \beta \sqrt{\overline{Q_o} + \sigma(Q_o) + \max(Q_o) + \min(Q_o)}, \tag{7}$$

$$T_A = \gamma \left[\frac{\overline{Q} + \sigma(Q)}{\max(Q) + \min(Q)} \right], \tag{8}$$

where Q_0 and Q denote the quantization error sets for Φ_0 and Φ_2 respectively, $\overline{Q}$ denotes the mean value operator, $\sigma(Q)$ denotes the standard deviation, and $\beta = 0.5$, $\gamma = 0.1$. In the case of Φ_0, observations with P-values under the abnormality threshold T_O are considered as outlier values and therefore removed from the training set. For Φ_2, if $P_{(n+1)}$ is smaller than T_A, the test observation x_{n+1} is considered abnormal.

3.3 Experimental Results

For the acquisition of training data, we monitored a home-like environment with an ASUS Xtion Live sensor installed on a platform 1.30 m above the ground and positioned parallel to the horizontal surface. Depth maps were acquired with a VGA resolution of 640 × 480 and the depth operation range was set from 0.8 to 4 m. The main technical characteristics of the Xtion live sensor are listed in Table 1 [7]. Video sequences were sampled at a constant frame rate of 30 Hz. To reduce sensor noise, we sampled the median value of the last 3 estimated points. Body centroids were estimated from depth map sequences based on the tracking skeleton model provided by the publicly available OpenNI/NITE framework.[1]

For the training phase and the system evaluation, we used video sequences from our data set with full-body actions performed by 13 different participants of the study with a normal physical condition [47]. To avoid biased execution, the participants had not been explained how to perform the actions. Training video sequences consisted of domestic actions such as walking, sitting down, standing up, and bending to pick up objects; abnormal actions consisted in falling down and crawling. We did not take into account those cases in which the user has already fallen on the ground since the tracking framework built on top of OpenNI would fail to provide a reliable recognition of the user and therefore, the extraction of body features would be highly compromised.

At detection time, new extracted vectors were processed to remove outliers. For the last three denoised vectors, a new test trajectory τ_{i+1} was obtained from Φ_1 and then fed to Φ_2 to compute the abnormality test $\lambda(\tau_{i+1})$. We took the last 3 abnormality test results and returned as abnormality output the result of the statistical mode:

$$Mo(\lambda(\tau_{i+1}), \lambda(\tau_{i+2}), \lambda(\tau_{i+3})). \tag{9}$$

A new output was therefore returned every 9 samples, which corresponds to approximately less than 1 s of captured motion. As shown by our experiments, this approach led to increased detection accuracy.

We evaluated the detection algorithm on abnormal actions using standard measurements defined by Van Rijsbergen [57]:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{10}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{11}$$

[1]OpenNI/NITE: http://www.openni.org/software.

Table 2 Evaluation of our abnormality detection algorithm on a data set of 13 participants

	Raw (%)	Denoised (%)	Improvement (%)
Recall	88	**95**	7.02
Precision	90	**97**	7.02
F-score	89	**96**	7.02
TN rate	90	**97**	6.90
Accuracy	89	**96**	6.96

Best results in bold

$$\text{F-score} = 2 * \frac{\text{Recall} \cdot \text{Precision}}{\text{Recall} + \text{Precision}}, \tag{12}$$

$$\text{True negative rate} = \frac{\text{TN}}{\text{TN} + \text{FP}}, \tag{13}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \tag{14}$$

A true positive (TP) was obtained when an abnormal event was detected between the first and the last frame where the abnormal action took place. True negatives (TN) refer to normal actions not detected as abnormal. False positives (FP) and false negatives (FN) refer respectively to normal actions reported as abnormal and abnormal behaviors not reported by the system.

The system evaluation is shown in Table 2. Our system detected abnormal fall and crawling events with 96 % accuracy. The removal of noise from the training and test set was of significant importance for reducing detection errors in presence of partial occlusions and tracking errors introduced by the mobile sensor, with an improvement in accuracy of 6.96 %. On the other hand, the accuracy of our system would be negatively influenced by: (1) highly-occluded users, leading to tracking errors and compromised feature extraction; and (2) the presence of actions sharing similar body features subject to classification ambiguity, i.e. detecting lying down as a fall, leading to a greater number of false positives.

4 Neurocognitive Robot Assistant

We now present a robot assistant that monitors a person in a household environment and reports abnormal user behavior such as a fall event. The underlying idea is that the robot will track a person in the scene while they perform daily activities, thereby tracking the user's position, body posture and speed. The information processed by the tracking framework is fed into the neural system which is responsible for triggering alarms of abnormal events. Whenever an abnormal action, e.g., a fall, is detected, the humanoid will approach the user and ask whether assistance is required. The robot will then take a picture of the scene that can be sent to the user's caregiver for telematic evaluation and agile intervention.

In this section, we first depict an overview of our system and its components. We then go into detail about the software, hardware, and the communication interface, and finally present the experimental set-up for fall detection in a home-like environment.

4.1 System Overview

Our fall detection system consists of a humanoid robot Nao extended with a depth sensor, a tracking framework to keep the user in the scene, and a learning-based system to process the visual information and detect abnormal user behaviors (Fig. 4). All these components communicate over a middleware layer based on Robot Operating System (ROS) that supports different hardware elements and programming languages.

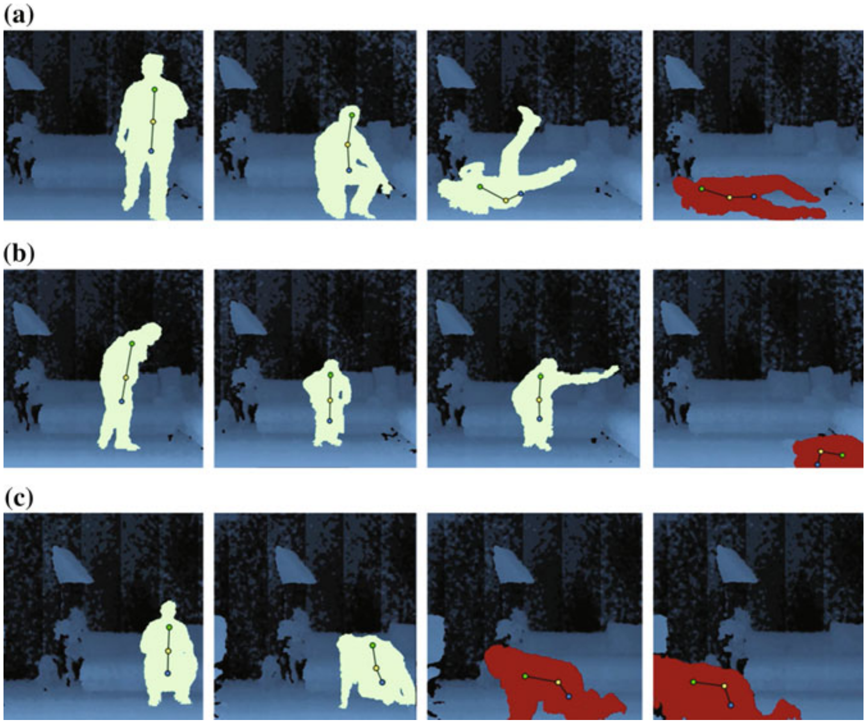

Fig. 4 Abnormal event detection from video sequences. The system can successfully detect abnormal actions and report them (*red body*). **a** Fall event, **b** fall event with partially occluded person, and **c** crawling sequence

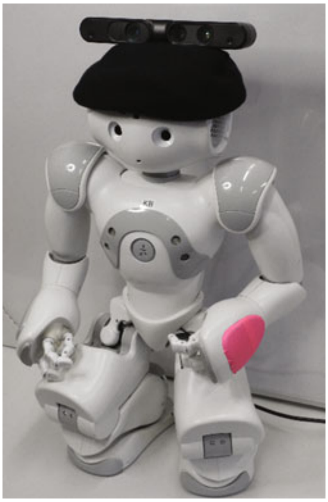

Fig. 5 Humanoid Nao extended with ASUS Xtion Live sensor on the head [9]. This approach allows to use Nao's actuators and sensed depth information to actively track a moving person in the environment

Table 3 Nao next gen specifications

Height	58 cm
Weight	4.3 kg
Autonomy	60–90 min (active/normal use)
Degrees of freedom	21–25
CPU	Intel Atom @ 1.6 GHz
Compatible OS	Linux, Windows, Mac OS
Vision	2 × HD cameras (1280 × 960)
Connectivity	Ethernet, Wi-Fi

Robot Nao (shown in Fig. 5) is a middle-size mobile humanoid robot developed by Aldebaran Robotics.[2] Since the beginning of the Nao project in 2004, the humanoid underwent a significant number of enhancements, thereby becoming the standard robot platform for a number of research institutions and robot competitions (e.g. RoboCup[3]). Nao includes an embedded multimedia system with microphones, speakers and two cameras. The main technical characteristics of Nao Next Gen are listed in Table 3. We extended the robot Nao with an ASUS Xtion depth sensor installed on top of the head (Fig. 5). The Xtion sensor was chosen over the Kinect because of its reduced power consumption and weight. A set of experiments with the extended Nao showed that wearing the sensor does not affect the overall stability of the humanoid while standing or walking. In contrast to the use of a fixed sensor, Nao will pan its head to seamlessly keep the moving person in the scene. In case of a fall event detection, a color picture of the scene will be taken using Nao's camera.

[2]Aldebaran Robotics: http://www.aldebaran-robotics.com/.

[3]RoboCup Project: http://www.robocup.org/.

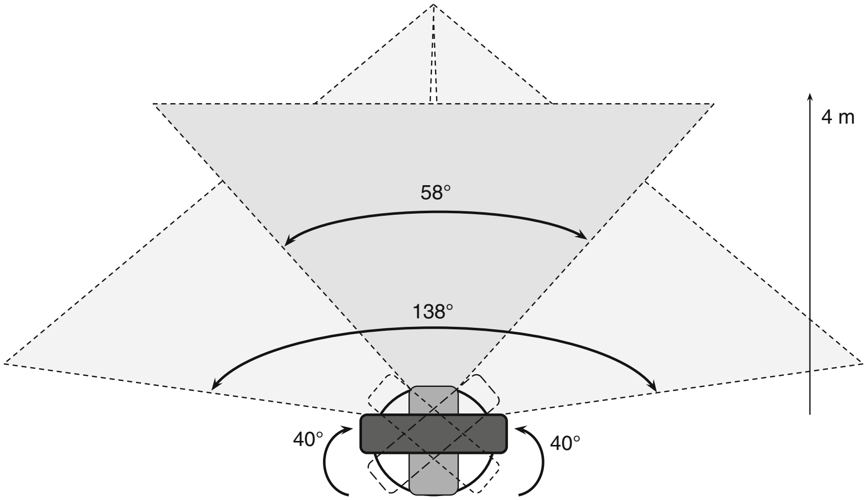

Fig. 6 Nao with Xtion sensor: extended horizontal field of view from 58° to 138° with a maximum head pan angle of 40° in each direction

4.2 Tracking with a Mobile Sensor

Depth sensors such as Microsoft Kinect and ASUS Xtion Live Pro are characterized by a reduced field of view (58 horizontal, 45 vertical, 70 diagonal), and therefore limiting their use in expansive environments. The idea of active tracking consists of seamlessly keeping the person in the scene, thereby moving the sensor when the person is approaching an area outside the field of view (FOV).

We use Nao's head to move the sensor and increase the horizontal FOV from 58° to 138° (Fig. 6). As a strategy for active tracking, we define a bounding box in which the target can act without the sensor being moved (Fig. 7a). We consider the upper-body centroid as the reference of the person's position. When the centroid lies outside the threshold, the tracking application will compute the needed operations to keep the person within the bounding box (Fig. 7b). Nao will then smoothly pan its head by 10° in the required direction, for a maximum pan angle of 40° in each direction (Fig. 6).

The body tracking application is built on top of simple-openni [58], which wraps the OpenNI–NITE framework for user identification, calibration and estimation of skeletal joints. We use this library with Processing IDE[4] with the purpose to enable a simplified access to some functionalities provided by the OpenNI such as skeleton tracking and scene analysis.

All system modules for active tracking communicate over Robot Operating System (ROS), a software framework for robot software development with

[4]Processing IDE: http://processing.org/.

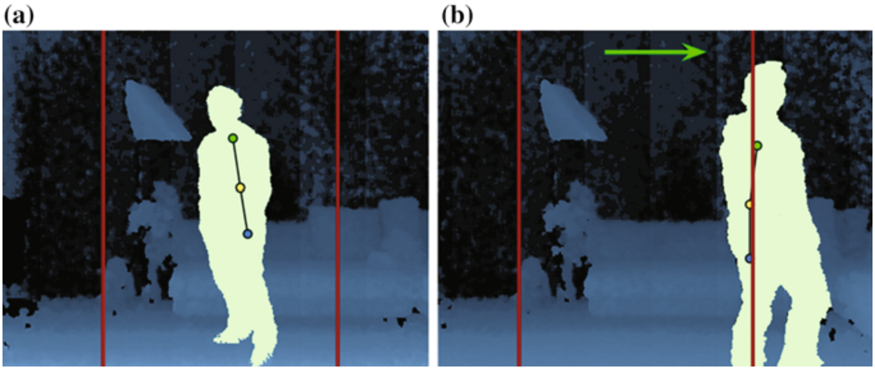

Fig. 7 Threshold-based active tracking strategy. When the upper-body centroid lies outside the threshold, the tracking application will compute the needed operations to keep the person within the bounding box (*red lines*)

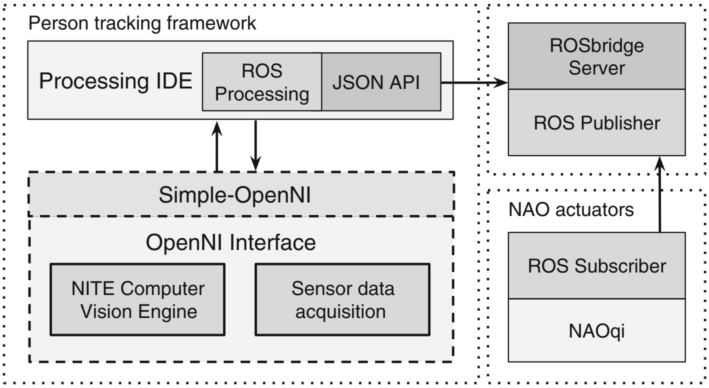

Fig. 8 A diagram of the communication network for interfacing the tracking framework with Nao's actuators over ROS

operating system-like functionality on a heterogeneous computer cluster [59]. It provides hardware abstraction, device drivers, libraries, visualizers, message-passing between processes and package management. A diagram of the overall architecture for active tracking is illustrated in Fig. 8. To interface our system modules, we use a ROS-based communication network implemented with *publisher-subscriber* nodes. We implement publisher nodes to continually broadcast a message over the network using a message-adapted class. The subscriber node will receive the messages on a given topic via a master node, which keeps a registry of publishers and subscribers. This specific architecture represents a robust interface to connect different applications, e.g. written in different programming languages, over a common network of communication. The tracking framework

Fig. 9 Person monitoring in a home-like environment. The Nao will seamlessly track the person while performing daily activities

communicates to ROS over Rosbridge[5] and a modified version of ROSProcessing,[6] extended to publish ROS topics. Rosbridge provides a JSON API[7] to ROS functionality for non-ROS programs. The rosbridge_suite package is a collection of packages that implement the rosbridge protocol and provides a WebSocket transport layer. We program Nao to move its head according to the tracking application via NAOqi framework,[8] which allows homogeneous communication between different Nao modules (motion, audio, video), and ROS integration.

4.3 Fall Detection Scenario

To test our system, we run the experiments in a home-like environment with a person performing daily activities, such as walking around the room, bending to collect objects, and sitting down to read (Fig. 9). The Nao was initially positioned on one side of the room to monitor the scene and connected to the system using wireless communication. The depth sensor was connected to a laptop (i5-3320 M

[5]ROSbridge_suite: http://wiki.ros.org/rosbridge_suite.
[6]ROSProcessing: https://github.com/pronobis/ROSProcessing.
[7]JSON API: http://jsonapi.org/.
[8]NAOqi framework: https://community.aldebaran-robotics.com/doc/1-14/dev/naoqi/index.html.

Fig. 10 Fall detection scenario: In case a fall event is detected by the system, the Nao will approach the fallen person (**a**) and take a picture of the scene (**b**)

2.6 GHz Processor and 4 GB of RAM) running all system modules under Ubuntu desktop 12.04[9] and ROS Groovy.[10] Whenever the person approaches the edge of the field of view of the sensor, Nao will pan its head to keep him/her in the scene. When a fall event is detected by the neural framework, the system will trigger an alarm. As shown in Fig. 10, the humanoid will approach the person by using the last tracked position before the fall and ask whether assistance is required. The color camera will be used to take a picture of the fallen person that can be sent to a relative or to the person's caregiver for further human assessment.

5 Discussion

The robust detection of falls in home environments represents a paramount component for assistive systems aiming to enhance the person's safety perception and avoid the loss of confidence due to, for instance, functional disabilities. In this context, vision-based fall detection has been shown to be a predominant approach due to the substantial advances in computer vision techniques and reduced cost factors with respect to wearable sensors. In this setting, the visual recognition of human actions is a key issue introducing a vast set of challenges for traditional 2D cameras. The use of low-cost depth sensors capable of performing 3D human motion segmentation and body posture estimation has led to promising results in experimental scenarios. However, despite the latest sensor trends, the question remains open on how to better process extracted body features for effectively extrapolating the complex dynamics of actions and fall events, exhibiting noise

[9]Ubuntu Desktop: http://www.ubuntu.com/desktop.
[10]ROS Groovy Galapagos: http://wiki.ros.org/groovy.

tolerance and robustness in real-world scenarios. Learning-based paradigms such as machine learning and neural networks have been shown to be a promising methodology for achieving knowledge generalization of a set of training activities and the subsequent classification of unseen situations [8, 9, 47].

In Sect. 3, we presented research on abnormal event detection based on unsupervised neural network learning. Our system consists of a hierarchical neural architecture that can learn a set of normal actions, e.g. walking, sitting, and lying down, captured by a depth sensor. After the training phase, the system will report novel behavioral patterns, e.g. fall events, as abnormal actions and trigger an alarm. The combination of a depth sensor with our neural network approach allows to tailor the robust detection of fall events independently from the background surroundings and changing light conditions. In addition, to contrast sensor noise and tracking errors, the neural architecture is also responsible for automatically removing noisy samples from the extracted body features during the training and test stage. Experiments run in a home-like environment showed that our system can detect fall events with high accuracy in real time (as shown in Table 2).

Contrary to the use of fixed ambient sensors, mobile assistive robots can undertake actions that benefit people with disabilities and seniors in a residential context. As supported by recent studies, socially-aware assistive solutions can provide positive effects on the senior's well-being in domestic environments [10], for instance, by supporting basic daily tasks of independent living and enhancing the user's experience through flexible human-robot interaction. On the other hand, this technology introduces new technical challenges and issues.

In Sect. 4, we introduced a humanoid robot to assist a person in a household environment and report abnormal user behavior such as fall events. The underlying motivation is to use a mobile robot to track the user's position, body posture, and motion characteristics when the user is performing daily activities. The processed visual information from the mobile depth sensor is fed into our neural system for abnormality detection. The removal of noise is of significant importance for reducing detection errors in presence of partial occlusions and tracking errors introduced by the mobile sensor, with an improvement in detection accuracy of 6.96 %. In case a fall event is detected, the humanoid will approach the user and then take a picture of the scene that can be sent to the user's caregiver for telematic evaluation and agile intervention. For our experiments, we did not consider those cases in which the user has already fallen on the ground when the robot starts to monitor the scene. This is due to the fact that a fallen person would not be detected by the tracking framework built on top of OpenNI that works better with moving users for user calibration and pose estimation. Therefore, the reliable detection and segmentation of a fallen user are open issues to be addressed, e.g. by using complementary RGB information to recognize a body on the ground [60].

The obtained results motivate future work in several directions. For instance, the ability of the robot to navigate in the environment for following the person through different rooms and finding a better angle of view to avoid body occlusions. At the current state of the system, the depth sensor must be wired to an external, fixed processing unit to perform the tracking, thereby limiting the mobility of the humanoid. To achieve better mobility, the sensor could be wired to an onboard processing unit and then transmit the depth information via WiFi for further processing to be carried out in the cloud. Moreover, video files could be adopted instead of a single picture to better support telematic human evaluation, e.g. sending a video with the last five seconds of the user's activity before the fall event. In fact, the role of human assessment is of crucial importance to determine the seriousness of the detected event and to undertake effective intervention.

To cope with the dynamic nature of real-world scenarios, a learning artificial system may not only be robust to unseen situations, but also adaptive. In fact, in addition to detecting short-term behavior such as fall events and domestic daily actions (e.g. walking, drinking, lying down), it may be of particular interest to monitor and learn the user's behavior over longer periods of time [5]. In this setting, it would be desirable to collect sensory data to, e.g., perform medium- and long-term gait assessment of the person, which can be an important indicator for a variety of health problems, e.g. physical diseases and neurological disorders such as Parkinson's disease [61]. To enhance the user's experience, assistive robots may be given the capability to adapt over time to better interact with the monitored user. This would include, for instance, a more natural human-robot communication including the recognition of hand gestures and full-body actions, speech recognition, and a set of reactive behaviors based on the user's habits. In this context, interdisciplinary research that takes into account the vast set of technical, social, and ethical issues regarding robots for assisted living is fundamental to provide feasible and reliable solutions in the near future.

Acknowledgements The authors would like to thank Erik Strahl for his invaluable technical contribution and help. The authors gratefully acknowledge funding by the DAAD German Academic Exchange Service (Kz:A/13/94748)—Cognitive Assistive Systems Project, by the DFG German Research Foundation (grant #1247)—International Research Training Group CINACS (Cross-modal Interaction in Natural and Artificial Cognitive Systems), and the DFG under project CML (TRR169).

References

1. World Health Organization: Global Report on Falls Prevention in Older Age. http://www.who.int/ageing/publications/Falls_prevention7March.pdf
2. Tinetti, M.E., Liu, W.L., Claus, E.B.: Predictors and prognosis of inability to get up after falls among elderly persons. J. Am. Med. Assoc. **269**(1), 65–70 (1993)
3. Scheffer, A.C., Schuurmans, M.J., van Dijk, N., van der Hooft, T., de Rooij, S.E.: Fear of falling: measurement strategy, prevalence, risk factors and consequences among older persons. Age Ageing **37**(1), 19–24 (2008)
4. Kaluza, B., Cvetkovic, B., Dovgan, E., Gjoreski, H., Gams, M., Lustrek, M.: A multi-agent care system to support independent living. Int. J. Artif. Intell. Tools **23**(1), 1–20 (2013)
5. Vettier, B., Garbay, C.: Abductive agents for human activity monitoring. Int. J. Artif. Intell. Tools **23** (2014)
6. Microsoft Kinect for Windows: http://www.microsoft.com/en-us/kinectforwindows/. Cited 10 Sept 2014

7. ASUS Xtion PRO LIVE sensor: http://www.asus.com/Commercial_3D_Sensor/Xtion_PRO_LIVE. Cited 10 Sept 2014
8. Jiang, Z., Lin, Z., Davis, L.S.: Recognizing human actions by learning and matching shape-motion prototype trees. IEEE Trans. Pattern Anal. Mach. Intell. **31**(3), 533–547 (2012)
9. Parisi, G. I., Wermter, S.: Hierarchical som-based detection of novel behavior for 3D human tracking. In: Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN), pp. 1380–1387, Dallas, Texas, USA (2013)
10. Kachouie, R., Sedighadeli, S., Khosla, R., Chu, M.-T.: Socially assistive robots in elderly care: a mixed-method systematic literature review. Int. J. Hum. Comput. Interact. **30**(5), 369–393 (2014)
11. Igual, R., Medrano, C., Plaza, I.: Challenges, issues and trends in fall detection systems. BioMed. Eng. OnLine 12–66 (2013)
12. Mubashir, M., Shao, L., Seed, L.: A survey on fall detection: principles and approaches. Neurocomputing **100**, 144–152 (2013)
13. Bourke, A.K., de Ven, V., Gamble, M., O'Connor, R., Murphy, K., Bogan, E., McQuade, E., Finucane, P., OLaighin, G., Nelson, J.: Assessment of waist-worn tri-axial accelerometer based fall-detection algorithms using continuous unsupervised activities. In: Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pp. 2782–2785. Institute of Electrical and Electronics Engineers, Buenos Aires (2010)
14. Lai, C.F., Chang, S.Y., Chao, H.C., Huang, Y.M.: Detection of cognitive injured body region using multiple triaxial accelerometers for elderly falling. IEEE Sens. J. **11**, 763–770 (2011)
15. Kerdegari, H., Samsudin, K., Ramli, A.R., Mokaram, S.: Evaluation of fall detection classification approaches. In: Proceedings of the 4th International Conference on Intelligent and Advanced Systems, pp. 131–136. Institute of Electrical and Electronics Engineers, Kuala Lumpur (2012)
16. Cheng, J., Chen, X., Shen, M.: A framework for daily activity monitoring and fall detection based on surface electromyography and accelerometer signals. IEEE J. Biomed. Health Inf. **17** (1), 38–45 (2013)
17. Albert, M.V., Kording, K., Herrmann, M., Jayaraman, A.: Fall classification by machine learning using mobile phones. PLoS ONE **7**, e36556 (2012)
18. Lee, R.Y.W., Carlisle, A.J.: Detection of falls using accelerometers and mobile phone technology. Age Ageing **0**, 1–7 (2011)
19. Fang, S.H., Liang, Y.C., Chiu, K.M.: Developing a mobile phone-based fall detection system on android platform. In: Proceedings of the Conference on Computing, Communications and Applications, pp. 143–146, Hong Kong (2012)
20. Abbate, S., Avvenuti, M., Bonatesta, F., Cola, G., Corsini, P., Vecchio, A.: A smartphone-based fall detection system. Pervasive Mob. Comput. **8**, 883–899 (2012)
21. Patsadu, O., Nukoolkit, C., Watanapa, B.: Survey of smart technologies for fall motion detection: techniques, algorithms and tools. In: Papasratorn, B., et al. (eds.) IAIT 2012, CCIS 344, pp. 137–147. Springer, Heidelberg (2012)
22. Botia, J.A., Villa, A., Palma, J.: Ambient assisted living system for in-home monitoring of healthy independent elders. Expert Syst. Appl. **39**, 8136–8148 (2012)
23. Lee, T., Mihailidis, A.: An intelligent emergency response system: preliminary development and testing of automated fall detection. J. Telemed. Telecare **11**, 194–198 (2005)
24. Miaou, S.G., Sung, P.H., Huang, C.Y.: A customized human fall detection system using omni-camera images and personal information. In: Proceedings of the 1st Distributed Diagnosis and Home Healthcare Conference, pp. 39–42. Institute of Electrical and Electronics Engineers, Arlington (2006)
25. 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. **21**, 611–622 (2011)
26. Liu, C.L., Lee, C.H., Lin, P.M.: A fall detection system using k-nearest neighbor classifier. Expert Syst. Appl. **37**, 7174–7181 (2010)

27. Cucchiara, R., Prati, A., Vezzani, R.: A multi-camera vision system for fall detection and alarm generation. Expert Syst. **24**, 334–345 (2007)
28. Hazelhoff, L., Han, J., de With, P.H.N.: Video-based fall detection in the home using principal component analysis. In: Bland-Talon, J., Bourennane, S., Philips, W., Popescu, D., Scheunders, P. (eds.) Proceedings of the 10th International Conference on Advanced Concepts for Intelligent Vision Systems, pp. 298–309. Springer, Juan-les-Pins (2008)
29. Diraco, G., Leone, A., Siciliano, P.: An active vision system for fall detection and posture recognition in elderly healthcare. In: Conference and Exhibition: Design, Automation and Test in Europe, pp. 1536–1541. European Design and Automation Association, Dresden (2010)
30. Han, J., Shao, L., Xu, D., Shotton, J.: Enhanced computer vision with microsoft kinect sensor: a review. IEEE Trans. cybern. **43**(5), 1318–1334 (2013)
31. Rougier, C., Auvinet, E., Rousseau, J., Mignotte, M., Meunier, J.: Fall detection from depth map video sequences. In: Abdulrazak, B., et al. (eds.) ICOST 2011. LNCS 6719, pp. 121–128 (2011)
32. Planinc, R., Kampel, M.: Introducing the use of depth data for fall detection. In: Personal Ubiquitous Computing, vol. 17, pp. 1063–1072. Springer, Heidelberg (2012)
33. Mastorakis, G., Makris, D.: Fall detection system using Kinects infrared sensor. J. Real-Time Image Process (2012) (Springer)
34. Zhang, C., Tian, Y., Capezuti, E.: Privacy preserving automatic fall detection for elderly using RGBD cameras. In: Miesenberger, K., Karshmer, A., Penaz, P., Zagler, W. (eds.) Proceedings of the 13th International Conference on Computers Helping People with Special Needs, pp. 625–633. Springer, Linz (2012)
35. Gasparrini, S., Cippitelli, E., Spinsante, S., Gambi, E.: A depth-based fall detection system using a kinect sensor. Sensors **14**, 2756–2775 (2014)
36. Cogniron: Cognitive Robot Companion. http://www.cogniron.org. Cited 15 Jan 2015
37. LIREC: Living with Robots and Interactive Companions. http://lirec.eu/. Cited 15 Jan 2015
38. Hermes: Cognitive Care and Guidance for Active Ageing. http://www.fp7-hermes.eu. Cited 15 Jan 2015
39. KSERA: Knowledgable SErvice Robots for Aging. http://ksera.ieis.tue.nl. Cited 15 Jan 2015
40. GiraffPlus: http://www.giraffplus.eu. Cited 15 Jan 2015
41. ROBOT-ERA: Implementation and integration of advanced robotic systems and intelligent Environments in real scenarios for the ageing population. http://www.robot-era.eu. Cited 15 Jan 2015
42. Amirabdollahian, F., Bedaf, S., Bormann, R., Draper, H., Evers, V., Gallego Perez, J., Gelderblom, G.J., et al.: Assistive technology design and development for acceptable robotics companions for ageing years. Paladyn J. Behav. Robot. **4**(2), 1–9 (2013)
43. Mundher, Z.A., Zhong, J.: A real-time fall detection system in elderly care using mobile robot and kinect sensor. Int. J. Mater. Mech. Manuf. **2**(2), 133–138 (2014)
44. Volkhardt, M., Schneemann, F., Gross, H.-M.: Fallen person detection for mobile robots using 3D depth data. In: Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (IEEE-SMC), pp. 3573–3578, Manchester, GB (2013)
45. Martinson, E.: Detecting occluded people for robotic guidance. In: IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), pp. 744–749, Edinburgh, Scotland, UK (2014)
46. Volkhardt, M., Gross, H.-M.: Finding people in home environments with a mobile robot. In: European Conference on Mobile Robots (ECMR), pp. 282–287, Barcelona, Spain (2013)
47. Parisi, G.I., Weber, C., Wermter, S.: Human action recognition with hierarchical growing neural gas. In: Wermter, S., et al. (eds.) Proceedings of the International Conference on Artificial Neural Networks (ICANN), pp. 89–96, Hamburg, Germany (2014)
48. Papadopoulos, G.Th., Axenopoulos, A., Daras, P.: Real-time skeleton-tracking-based human action recognition using kinect data. In: Gurrin, C., et al. (eds.) MultiMedia Modeling. LNCS, vol. 8325, pp. 473–483, Springer International Publishing (2014)
49. Hu, W., Tan, T., Wang, L., Maybank, S.: A survey on visual surveillance of object motion and behaviors. IEEE Trans. Syst. Man Cybern. **34**(3), 334–352 (2004)

50. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: a survey. ACM Comput. Surveill. **15** (2009)
51. Kohonen, T.: Self-organizing map, 2nd edn. Springer, Heidelberg (1995)
52. Hu, W., Xie, D., Tan, T.: A hierarchical self-organizing approach for learning the patterns of motion trajectories. IEEE Trans. Neural Netw. **15**(1), 135–144 (2004)
53. Nag, A.K., Mitra, A., Mitra, S.: Multiple outlier detection in multivariate data using self-organizing maps title. Comput. Stat. **2**(2), 245–264 (2005)
54. Parisi, G.I., Barros, P., Wermter, S.: FINGeR: framework for interactive neural-based gesture recognition. In: Proceedings of the European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), pp. 443–447, Bruges, Belgium (2014)
55. Parisi, G.I., Jirak, D., Wermter, S.: HandSOM: neural clustering of hand motion for gesture recognition in real time. In: Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), pp. 981–986, Edinburgh, Scotland, UK (2014)
56. Hoglund, A.J., Hatonen, K., Sorvari, A.S.: A computer host-based user anomaly detection system using self-organizing maps. In: IEEE-INNS-ENNS International Joint Conference on Neural Networks, vol. 5, pp. 411–416 (2000)
57. Van Rijsbergen, C.J.: Information retrieval, 2nd edn. Information Retrieval, Butterworth-Heinemann, London (1979)
58. simple-openni—OpenNI library for Processing: https://code.google.com/p/simple-openni/. Cited 15 Jan 2015
59. The Robot Operating System (ROS): http://www.ros.org/. Cited 15 Jan 2015
60. Wang, S., Zabir, S., Leibe, B.: Lying pose recognition for elderly fall detection. In: Conference on Robotics: Science and Systems VII, pp. 44–50, Los Angeles, CA, USA (2011)
61. Aerts, M.B., Esselink, R.A.J., Post, B., van de Warrenburg, B.P.C., Bloem, B.R.: Improving the diagnostic accuracy in parkinsonism: a three-pronged approach. Pract. Neurol. **12**(2), 77–87 (2012)

Authors Biography

German I. Parisi received his Bachelor's and Master's degree in Computer Science from the University of Milano-Bicocca, Italy. Since 2013, he is a research associate in the Knowledge Technology Group at the University of Hamburg, Germany, where he is part of the research project CASY (Cognitive Assistive Systems) and the international Ph.D. research training group CINACS (Cross-Modal Interaction in Natural and Artificial Cognitive Systems).

His main research interests include neurocognitive systems for human-robot assistance, computational models of the visual cortex, and bio-inspired action and gesture recognition.

Stefan Wermter received the Diploma from the University of Dortmund, the M.Sc. from the University of Massachusetts, and the Ph.D. and Habilitation from the University of Hamburg, all in Computer Science. He has been a visiting research scientist at the International Computer Science Institute in Berkeley before leading the Chair in Intelligent Systems at the University of Sunderland, UK. Currently Stefan Wermter is Full Professor in Computer Science at the University of Hamburg and Director of the Knowledge Technology institute.

His main research interests are in the fields of neural networks, hybrid systems, cognitive neuroscience, bio-inspired computing, cognitive robotics and natural language processing. In 2014 he was general chair for the International Conference on Artificial Neural Networks (ICANN). He is also on the current board of the European Neural Network Society, associate editor of the journals "Transactions on Neural Networks and Learning Systems", "Connection Science", "International Journal for Hybrid Intelligent Systems" and "Knowledge and Information Systems" and he is on the editorial board of the journals "Cognitive Systems Research" and "Journal of Computational Intelligence".