Multimed Tools Appl (2018) 77:6405–6424
DOI 10.1007/s11042-017-4549-5

Combined curvelets and hidden Markov models for human fall detection

Nabil Zerrouki[1] · **Amrane Houacine**[1]

Received: 12 August 2016 / Revised: 7 January 2017 / Accepted: 24 February 2017 /
Published online: 8 March 2017

Abstract Fall events detection is one of the most crucial issues in the health care of elderly people. This paper proposes an innovative approach for reliably detecting fall incidents based on human silhouette shape variation in vision monitoring. This mission is achieved by: (i) introducing the curvelet transform and area ratios for identifying human postures in images; (ii) reducing the feature vector dimension using differential evolution technique; (iii) identifying postures by a support vector machine, and (iv) adapting a hidden Markov model for classifying video sequences into non-fall and fall events. Experimental results are obtained on several "Fall Detection" datasets. For evaluation, several assessment measures are computed. These evaluation measures demonstrate the effectiveness of the proposed methodology when compared to some state-of-the-art approaches.

Keywords Human fall classification · Video monitoring · Curvelet transform · Hidden Markov Model

1 Introduction

In view of the imminent progress of information techniques, communication technologies and intelligent video systems; "looking at people's behaviors" research field has received an important consideration. Human gesture interpretation has made a large progress since it helps to replace traditional passive monitoring systems. Many applications are developed in the literature, such as smart rooms, people detection [22], environment modeling [36],

✉ Nabil Zerrouki
nzerrouki@usthb.dz

Amrane Houacine
ahouacine@usthb.dz

[1] LCPTS, Faculty of Electronics and Computer Science, University of Sciences and Technology Houari Boumédienne (USTHB), Algiers, Algeria

face detection [43], motion estimation [37, 38], just to name a few . In this work we have picked one of the most challenging applications in behavior understanding domain known as "Human Fall Detection" [41]. Several statistical studies showed that falls are the main reason of hospitalizations for injuries for individuals with special needs like elderly people, disabled, overweight and obese, etc. In a recent report from the Health Evidence Network for the World Health Organization [1], it is established that 30% of elderly over 65 years old falls at least once each year. In addition, Eurostat study predicted that the percentage of people older than 65 years old will rise from 17.1% to 30%. This corresponds to an increase of 84.6 million individuals during 2008 to 151.1 million individuals in 2060 [14]. Not only falls are the main motive for injuries, but these lead to serious wound and even to deaths, especially when elderly remain lying on the floor for a long periods of time or even days after accidents [1, 14]. Because of this important need and the availability of different data repositories, several fall detection studies have been proposed in the literature. Where many devices have been invoked, including: cameras, infrared sensors, accelerometers, microphones, pressure and floor sensors. However, these fall detection systems are generally distinguished into two classes: non-computer vision based methods and computer vision based methods.

2 Problem statement and background

This section is dedicated to a brief review of the existing fall detection methods (extracted features, classification method, and obtained performances), and the introduction of the proposed methodology. The basic principle is to detect sequences representing human fall through sequences describing different human behaviors. In non-vision-based approaches, different sensors (including floor vibration devices, acoustic sensors and wearable acceleration devices) are exploited to measure accelerations, vibration, sound, etc. [20, 44]. However, these sensors present some limitations, which make their usage inappropriate in fall detection application. Acoustic sensors (microphones) are frequently disrupted by sounds and noises present in the environment, which can generate false alarms [20]. In applications using floor vibration sensors [44], the detection is limited only to surfaces equipped with devices. The charges of the sensor installation on a large area of ground are very important, which makes these methods inappropriate in real life. On the other hand, systems using wearable sensors (e.g., inertial measurement units) present an emerging tendency towards automatic fall identification [18]. However, even if wearable sensors are considered as promising approach due to their availability at low costs, they present many limitations namely: (i) only falling instant is detected, and fall duration remains unknown. (ii) Many seniors forget or find an inconvenient to continuously wear accelerometers. (iii) In addition, some sensors are sensitive to the presence of noise in the living environment, and then generate false alarms [18].

To overcome these shortcomings, many researchers adopted the fall detection based on computer vision approach. In this context, Rougier et al. [28, 29] have extracted attributes from the captured video sequences and simple threshold techniques have been applied to determine whether there is a fall or not. The body's form change information and the head's velocity were extracted, and an appropriate threshold was set manually to separate between fall and non-fall activities. However, this method produces an important false detection rate (several fast sitting activities were misclassified as falls) and the performance was strongly depended on the threshold value. Auvinet et al. [4] proposed a fall identification method using the reconstructed 3-D shape of people. Fall events were identified by considering the

volume distribution along the vertical axis, and an alarm is generated when the main portion of distribution is suddenly spanned close to the ground over a fixed period of time (according a predefined time threshold). However, many cameras are required (four or more), and a graphic process unit was necessary for computation. Foroughi et al. [13] proposed an ellipse approximation of a body silhouette as features. M. Yu et al. [39] presented a home care system with fall detection application. They have also adopted an ellipse approximation combined with horizontal and vertical projection histograms. X. Ma et al. [21] proposed shape attributes based on curvature scale space to describe the human body. Mueller et al. [25] introduced Boolean features to express geometric associations with body articulations. Chen et al. [9] introduced a coarse-to-fine strategy for detecting falls, principally in dark scenes. The downward optical flow attributes are calculated using thermal images. Olivieri et al. [26] used the spatiotemporal characteristics to differentiate different gestures defined as gesture vector flow templates. Doukas et al. [12] introduced the spatiotemporal energy plan for detecting fall activities. In [19], Lee et al. used carefully engineered features for fall detection, such as silhouette features, lighting features, and flow features to separate between people and multiple moving objects. Anderson et al. [3] extracted a bounding box and motion information from consecutive silhouettes as features.

On the other hand, the evolution of machine learning and pattern recognition algorithms led to a considerable developing of automatic fall detection systems. In [2], Alhimale et al. used a neural network classifier to analyze the silhouette's binary map images and then identified the fall situations. In [18], Kwolek et Kepski presented k-nearest neighbor (KNN) algorithm for fall detection, where a correct detection rate of 95.71% was obtained according to their experimental results. In [6], Bourke et al. proposed two fall detection systems based on lower and upper fall thresholding. Yun et al. [40] used support vector machine classifier for developing a real-time fall detection method. Yu et al. [39] introduced a modified version of SVM: a Directed Acyclic Graph SVM (DAGSVM) for posture classification and fall detection.

2.1 Motivation

Various attributes have been employed to characterize human body. However, most of these features are strongly dependent on variation of human body size and position in the image (caused by the variation of distance between people and cameras). An obvious need for scale invariant features appears. The second notice that motivates our work is associated to the classification task. Even if several algorithms have been used for people activity understanding, many algorithms do not consider the time information. These methods are more appropriate when dealing with static problems like posture classification, but present some limitations with sequential problems. In particular, depending on duration of the transition from a particular human posture to the lying position, one can clearly distinguish between real falls and like fall activities.

2.2 Contribution

As mentioned before, some body shape features, which do not consider the multi-resolution aspects, suffer from some major shortcomings when body silhouette size and position vary. In this work, alternative features based on the curvelet coefficients are proposed to avoid these limitations. Not only that the curvelet coefficients are known to be translation and scaling invariant, but no prior camera calibration is required. Five body's partial occupancy areas are also introduced to permit a precise characterization for human posture classification, and

resolve the problem of rotation invariance. An SVM is used to classify the statistic postures in the sequence frames. The posture classes so obtained are then defined as HMM states for the following task of fall detection. This approach gives thus a physical meaning to the HMM states.

The remaining part of this paper is structured as follows: Section 3 presents a global system overview. In Section 4 the environment modeling and people segmentation is reported. Section 5 describes the use of curvelets and area ratios as body'features. In Sections 6 and 7 the SVM posture classification and HMM action recognition are respectively presented. The classification with reject chart and evaluation measures are described in Sections 8 and 9 respectively. Finally, comparative tests and the results of the proposed scheme over the two publicly existing fall detection databases: the University of Rzeszow's fall detection dataset (URFD) [18] and the fall detection dataset (FDD) [8] are given in Section 10.

3 System overview

The global flowchart of the proposed fall detection system is illustrated in Fig. 1. In the next sections, we will describe different blocks of this flowchart in detail. The system implementation comprises three main parts namely: background subtraction, feature generation and recognition. In this work, we segment and track the foreground within frames using a background subtraction method. After that, the curvelet coefficients are used as human body's attributes. However, in addition to translation and scaling, curvelet transform is also rotation invariant. The rotation invariance is rather a drawback for posture recognition. To surmount this limitation and consider the angular information, five body's partial occupancy areas are computed and added as complementary attributes.

To maintain a raisonable dimensionality of feature vector, a feature selection technique based on Differential Evolution (DE) is invoked. The reduced feature vector is then sent to

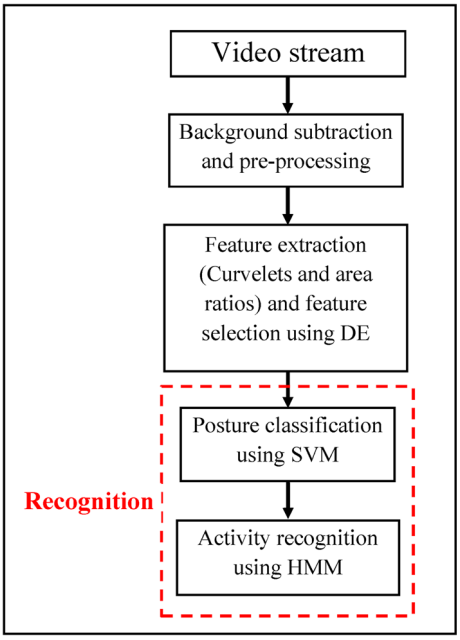

Fig. 1 Proposed system flow

the SVM for posture classification. The last module performs human activity recognition, where an HMM endowed with reject chart is adapted to model non-fall and fall activities. The choice of HMM was motivated by its capacity to exploit the variability in time, which can be very beneficial to the application at hand. In various applications, classifier results are meaningless when the sequential nature of samples is not considered. For example, in brain machine interface, brain signals can represent visible information whereas series of stimuli can be assimilated to sequential data. Likewise, in face identification across age can be further improved if the models devised incorporate naturally the age information as sequence. In fact, fall detection accuracy will increase if the time duration between different postures is exploited.

4 Environment modeling and people segmentation

The segmentation consists of extracting pixels belonging to the body silhouette [17], and eliminating all background's pixels [30]. In this paper, background subtraction is applied as segmentation method, where a background image is used as reference to eliminate unchanged pixels in the following frames. A background template is built by using successive frame differences for locating stationary pixels. A threshold is then required to decide whether pixel is affected to the foreground or the background from two consecutive frames. In image difference, if the pixel is considered as moving, the corresponding label is set to zero. On the contrary, if the pixel is stationary during N frames, it is then considered as a background region. In this study, N is set experimentally to 10, with consideration of indoor and outdoor speed of moving people. Another value of N could also be taken. For an input image, the absolute value of difference between current image I(x, y) and background model B(x, y) for each pixel (x, y) is computed and compared with a fixed threshold. In this case, the threshold value is defined by $2\sigma(x, y)$, where $\sigma(x, y)$ is the standard deviation of the pixel (x, y) in the registered background. A simple technique based on gradient filter that is proposed in [10] is also used to eliminate shadow effect.

After segmentation, some noise regions can be observed. To eliminate this noise, the morphological processing based on erosion and dilation operators with 3×3 structuring elements, is applied. A sample of background elimination technique is shown in Fig. 2. The first and the second column illustrate the background and the input frames respectively, while third and last columns correspond to the segmented images, respectively before and after applying morphological processing.

5 Human body feature extraction

Feature extraction is one of the most crucial phases in the classification procedure, since the extracted attributes have an important effect on the recognition accuracy. As stated previously, several characteristics have been used until now. However, most of these features are strongly dependent on the human silhouette size and position. The silhouette size changes as the distance from people to camera varies. A scaling operation with distance factor is frequently used, but this operation requires a prior camera calibration. For that reason, a curvelet transform is introduced to characterize the posture shapes. Essentially that the curvelet coefficients are considered as multi-resolution features [31], and they can extract information that is usually missed with other transforms like directionality and anisotropy [42]. It is well known that curvelets are more popular in image preprocessing tasks such as:

Fig. 2 Result of background subtraction algorithm: the *first column* represents background images, the *second column* represents input images; and third and fourth columns illustrate segmentation results, respectively before and after morphological processing

image enhancement, compression, and restoration; but during the last few years, curvelet transforms are exploited in solving recognition problems, one can cite character [23] or face recognition [33].

In our work, curvelet transform is applied on the binary images (representing only the human body silhouette, extracted during segmentation phase). Compared to wavelet or gabor wavelet transforms [16, 24] which are more adapted for grayscale images, curvelets provide better directional information and constitute an appropriate representation for edges and other singularities along curves. Another motivation for introducing curvelet coefficients as attributes, is tied to its ability of capturing structural information (singularity) in frequency domain using multiple radial directions. At any scale j, curvelets provide a sparse representation $O(2^{j/2})$ of the images compared to wavelets $O(2^{j})$. The curvelet function applied on an image I can be defined as follow:

$$C(I) = 2^{j/2} C(D_j R_l I - K_t), \tag{1}$$

where D_j, R_l and K_t represent respectively the scaling matrix, the orientation, and the translation parameters.

The binary image of human silhouette is first decomposed and then devided into wedges in the frequency domain [31]. The parabolic shape of wedges is the result of partitioning into concentric circles and angular divisions (thus the decomposition of the image into multiple scales j or subbands and the angular divisions into different angles or orientations l. In this work, we applied curvelet trasforms using four levels of scale (j = 4) and with eight directions (l = 8). This produces the approximate image which contains the low-frequency components and the detailed coeficients representing high-frequency details along different orientations. Figure 3 shows an example of the decomposition of a binary image representing human body in a standing posture. However, curvelet transforms do not consider the rotation information, since they are rotation invariant coefficients. This information is

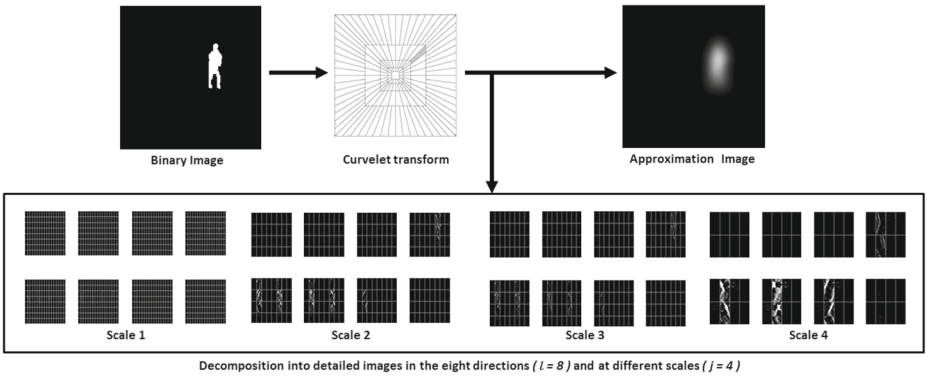

Fig. 3 Curvelet transform flowgraph. The figure illustrates the decomposition of the original image into subbands followed by the spatial partitioning of each subband

necessary in posture identification, and the rotation invariance can cause confusion among some postures that are mainly discriminated through rotation information. To get through this issue, a complementary five partial occupancy areas of the body are added as features.

These areas typically correspond to the body parts in a standing posture, namely: head, arms, and legs as shown in Fig. 4. The partitioning is performed by tracing five segments from the body's center of gravity. The first segment is the vertical one, the second and third segments are located at 45° on either side of the vertical segment, and the fourth and the fifth

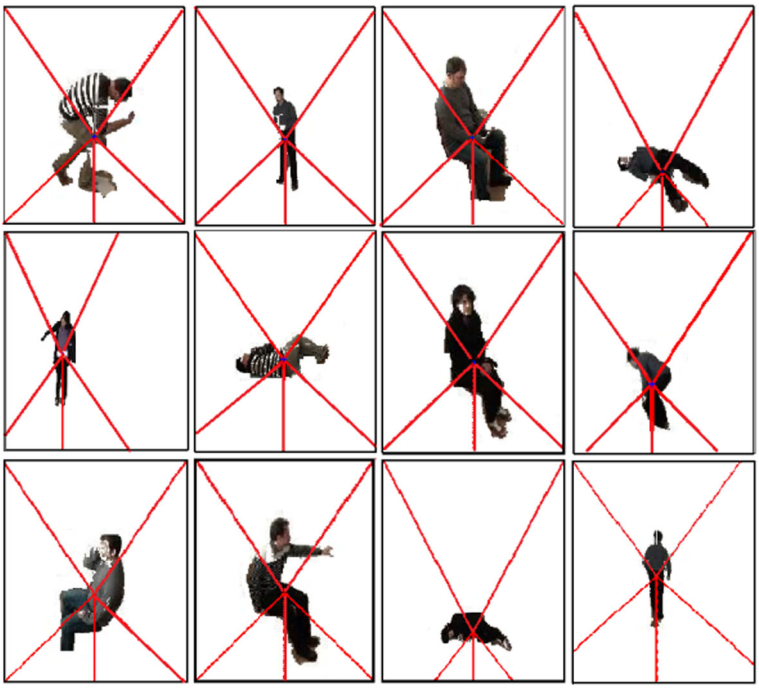

Fig. 4 Samples of partitioned human body areas with different postures

segments are situated at 100° on either side of the second and the third segments respectively (refer to Fig. 4). To maintain the scale invariance gained through the use of curvelets, we rather use normalized areas by dividing each area by the whole body area. Given the total human body area A, and the partial areas A_i; $i = 1, ..., 5$, the partial normalized areas are defined by the following ratios:

$$Ri = \frac{A_i}{A} \tag{2}$$

Together these ratios are used as supplementary descriptors in the fall detection process, these ratios are (i) translation invariant, (ii) scaling invariant, and (iii) consider the rotation information, necessary for fall identification. These ratios are representative enough to achieve an attractive classification accuracy, and not too complex in order to allow a fast processing. In Fig. 5, we present a time evolution of the five ratios while performing daily and fall activities respectively.

These ratios are then added to curvelet coefficients to form the entire feature vector. However the resulting vector size remains large, thus a feature selection procedure is applied. The objective is to achieve a new feature vector with a reduced dimension, by reducing unnecessary and correlated features. This issue is addressed using differential evolution (DE) algorithm for feature selection. The selected features allow a considerable reduction in computation time, what is important for real time application.

The DE algorithm was originally introduced by Storn and Price in 1995 [32], and it has been adapted in different fields [11]. In this work DE is utilized as feature selection algorithm applied on curvelet coefficients in order to reduce redundant information. Initially, original features are randomly and iteratively modified. The process is then enhanced using mutation, crossover and selection operations as shown in Fig. 6. During mutation operation, DE generates new element (called mutant element V_i) by combining original features X_i, where weighted difference between two features namely X_1 and X_2 is added to a third original feature X_3:

$$V_i = X_1 + F(X_2 - X_3) \tag{3}$$

Where X_1, X_2, and X_3 are randomly chosen among original features to create the mutant element V_i; and $F \in (0, 1)$ represents the scale element that tests the rate corresponding to the population variations. In order to measure the efficiency of mutant elements, DE crosses

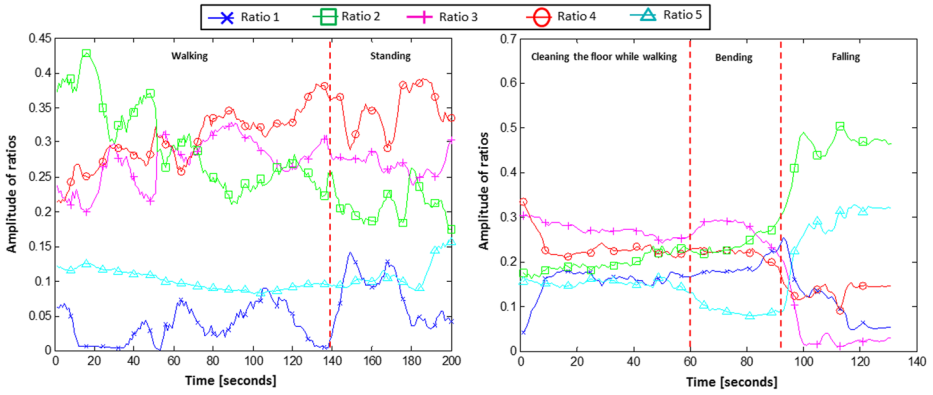

Fig. 5 A sample of five ratios time evolution of daily (*left*) and fall (*right*) activities

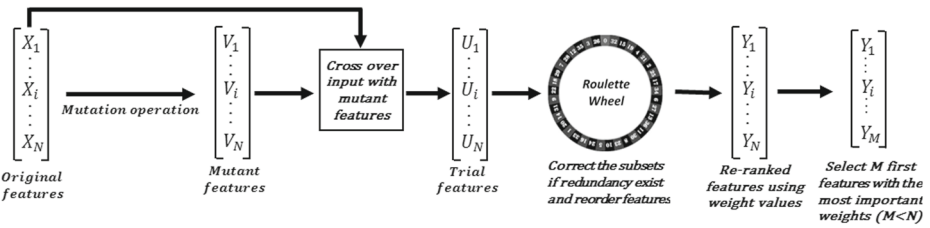

Fig. 6 Feature selection technique using Differential Evolution algorithm

each new mutant element with an original feature. This step is needed to choose the best elements U_i between the original X_i and the mutant V_i features:

$$U_i = \begin{cases} V_i & \text{if } rand(0,1) < C_r \\ X_i & \text{Otherwise} \end{cases} \tag{4}$$

Where U_i is the trial element, and $C_r \in [0, 1]$ is a user defined value that controls the fraction of parameter values that are copied from the mutant. This new vector U corresponds to the new features either the mutant feature V_i (corrected version) or the original feature X_i depending on which one of them achieved a better fitness.

The next phase of DE is to rank these features from higher to lower, according to their corresponding weight values. The estimation of weights is based on the classification accuracy, since selected features depend strongly on the application at hand. To do that a roulette wheel weighting scheme is employed. The process starts with all U_i elements, and then applies recursive feature elimination to pick features (using roulette wheel) by considering smaller set of features. In this work, the feature weights are calculated from the distribution factors associated with each feature. Thus, features are ranked by the roulette wheel according to the highest distribution factors (for more details, see reference [11]). Finally, DE feature selection is completed by keeping only features with the most important weights.

The next section is dedicated to posture classification, where each frame is represented by the selected features (from curvelet transforms) and the area ratios. However, during fall detection, the classification will be applied on videos. Since frames of a video sequence are assimilated to an observation sequence, the set of feature vectors, each consisting of the selected features and the five area ratios, are concatenated to define the whole feature vector associated to the video sequence.

6 Posture classification using Support Vector Machines (SVMs)

The classification procedure comprises two main stages: (i) assign the system certain postures as training samples, and (ii) assign the rest of image postures according to their feature spaces via a trained model. The key idea behind the choose of SVM as classifier is its ability to behave as both a linear and a nonlinear classifier through the use of nonlinear kernels. Their training consists to find a function f (i.e., an optimal separating hyperplane) to map each class into their corresponding label space. Specifically, let $(P_n, y_n) \in R^M \times \{\pm 1\}$ be a set of training patterns, M is the data space dimension, $(n = 1, N_c)$, N_c is the number of samples in the class ω and let $f : R^M \rightarrow \{\pm 1\}$ be a set of functions. The training process

selects the function f which maximizes the margin between the two classes. Then, data are classified according to the signal of the decision function so that:

$$f(P) = sign\left[\sum_{i=1}^{SV} y_i \alpha_i k(P_i, P) + b\right], \tag{5}$$

The optimal hyperplane corresponds to $f(P) = 0$ where b is a bias. The number of support vectors SV is obtained with training samples whose Lagrange multipliers α_i are not zero. The kernel $k(P_i, P)$ is any mathematical function, respecting Mercer's conditions [35]. The Radial Basis Function (RBF) kernel provides commonly the best performance for pattern recognition applications. For two samples P_1 and P, this kernel is expressed as:

$$k(P_i, P) = exp\left[-\frac{1}{2\sigma^2}\omega(P_1, P)\right] \tag{6}$$

$$\omega(P_1, P) = \|P_1 - P\|^2 \tag{7}$$

where, σ is user-defined parameter.

To settle the multi-class cases, two approaches are generally employed. The One-Against-All (OAA), requiring N binary SVMs to solve a problem with N classes. The second approach is the One-Against-One (OAO) working with $N \times (N-1)/2$ SVMs, and each SVM is used to separate two classes. In this work, an OAA approach with the RBF kernel was implemented using the LIBSVM package [7].

The multi-class SVM is then applied to distinguish between five main posture classes, namely standing, bending, sitting, squatting, and lying. The main idea of the definition of these particular postures as classes is that any human gesture (daily activity or fall) can be represented by a combination of these posture classes. This SVM classification is applied on statistic image (each frame separately), according to their extracted features. During SVM classification phase, each frame is labeled by one of the posture classes. So, for the next phase (HMM classification) frames constituting video sequences are already tagged and these posture classes will be defined as states of the HMM classifer.

7 HMM action recognition

Based on learning probabilistic models, HMM play a key role in solving classification problems, essentially for sequential data. The relevance of these statistical paradigms was first developed by Baum and Petrie [5] and demonstrated in speech identification in the late 1980's by Rabiner and Juang [27]. Neighbor domains like signal processing, handwriting and text identification have also showed promise using stochastic models. Given a set of two action classes $\lambda = \{\lambda_1, \lambda_2\}$, namely 'not falling' and 'falling', the objective is to find the class λ^* which maximize the probability $P(\lambda/O)$, where $O = \{O_1 ... O_T\}$ is the entire sequence of frame-wise observations. The principle of HMM is to construct a model for the two actions one wants to track. HMM should be trained for each action and the classification of a sequence O is then assigned to the model presenting the highest probability. In his famous tutorial, Rabiner mentioned that HMMs can solve a challenging classification problems without having directly observable states (hence a designation of hidden states) [27]. In this study, a physical meaning was given to generated states.

To define the components constituting the model, an HMM is generally characterized by:

- N, the number of hidden states q_i in the model. The states are usually named hidden but can have a physical significance. In our case, they are described by the different body postures (standing, sitting, lying, etc.).
- M, the number of distinct observation per hidden state. The observation symbols O_i correspond to the output of the system being modeled.
- The initial state distribution $\pi = \{\pi_i\}$, where: $\pi_i = P(q_1 = S_i)$, $1 \leq i \leq N$, and $\sum_i \pi_i = 1$.
- The state transition probability distribution $A = \{a_{ij}\}$, where: $a_{ij} = P\left(q_{t+1} = S_j / q_t = S_i\right)$, $1 \leq i, j \leq N$, and $\sum_j a_{ij} = 1$.
- The observation symbol probability distribution in state j, $B = b_i(k)$, $where : b_i(k) = P(O_k \text{ at time t}/q_t = S_i)$, $1 \leq k \leq M$ and$1 \leq j \leq N$ and $\sum_k b_i(k) = 1$

An HMM model is usually represented as $\lambda = [\pi, A, B]$. There are three main steps which describe an HMM:

- Evaluation: Determine the probability allocated to an observation by the model λ.
- Decoding: Find the most likely sequence of hidden states $q_1, q_2, ..., q_T$ that generated the sequence of observation O.
- Learning: this phase consists of determining the HMM elements $\left(a_{ij}, b_{jk}, etc\right)$ based on the training observation sequences.

Let's focus now on the evaluation problem. In the HMM concept, observations are supposed to be state conditionally independent. The evaluation step is to estimate matching between the model λ and the observation sequence by calculating $P(O/\lambda)$. The mathematical description can be written as follows:

$$P(O/\lambda) = \sum_{all_q} P(O, q/\lambda) \tag{8}$$

$$P(O, q/\lambda) = P(O, /q, \lambda) \times P(q/\lambda) \tag{9}$$

and using state conditional independence, we finally can write:

$$P(O/q, \lambda) = \prod_{t=1}^{T} P(O_t/q_t, \lambda) \tag{10}$$

However, if both activities are not distinct enough and the class probabilities are close, the system will use the reject chart. Since in fall detection application, the case of quasi-equal probabilities may be encountered.

- **Instantiation of the HMM in human fall detection**

In this paper, we remind that "fall detection" is viewed as classification problem with only two classes. The HMM formalism fits seamlessly with the application at hand since: (i) The samples represent a sequential aspect (sequence of frames), (ii) The states have a physical meaning (posture classes). Thus, one can perform the following instantiation:

- The training phase is performed using sequences from both classes "non-fall" and "fall" separately, since the observation sequence is tagged "fall" or "non-fall" depending on the video sequence. Therefore, two optimal models are obtained during each training

phase namely: $\lambda_1 = [\pi_1, A_1, B_1]$ and $\lambda_2 = [\pi_2, A_2, B_2]$. It is worth noting that during the supervised training, transition and emission matrices are estimated using respective frequencies.
- An observation sequence Oi is represented by a posture sequences (which are defined as HMM's states).
- A state qi is associated to different posture classes (already given by SVM classifier) that emitted the posture sequences. A state can be: Lying (Ly), Standing (St), Sitting (Si), Bending (Be), or Squatting (Sq).

8 Losses and risks

The integration of the reject chart boosts significantly the classification performance. It is well established that a classification problem with a reject condition (whereby the classifier has the possibility to withhold its decision) incurs a cost lower than a misclassification. The question that arises is whether reject cases occur in this application? Indeed, there are numerous scenarios in which a human posture is barely described due to the presence of shadows, or partial occlusions of body parts with environment objects. A reject condition is then determined via risk minimization associated to the loss matrix $\Im$ [34].

Let's denote $\alpha_i (i = 1, 2)$ the action of assigning the video sequence O to class λ_i ("fall" or "non-fall") and α_3 the action that corresponds to the reject decision. Each line in the loss matrix corresponds to an action α_i to be undertaken. An optimal classifier advocates the action α_i with a minimum risk: $\Re(\alpha_i/O) = min_k\ \Re(\alpha_k/O)$. A correct classification is assigned a zero-loss, so $\omega_{11} = \omega_{22} = 0$. However, the classification errors are not equally costly in fall detection problem ($\omega_{12} \neq \omega_{21}$). In other words, misclassifying fall sequence is more critical than wrongly classifying non-fall sequence. Therefore, to discriminate losses between these two wrong decisions, different values of ω_{12} and ω_{21} have been used during experiment. The loss matrix is presented as:

$$\Im = \begin{bmatrix} \omega_{11} & \omega_{12} \\ \omega_{21} & \omega_{22} \\ \omega & \omega \end{bmatrix}$$

Therefore, the risk of reject is:

$$\Re(\alpha_3/O) = \sum_{k=1}^{2} \omega P(O/\lambda_k) = \omega \tag{11}$$

and the risk of choosing class $\lambda_i (i = 1, 2)$ is:

$$\Re(\alpha_i/O) = \sum_{k \neq i} \omega_{ik} P(O/\lambda_k) \tag{12}$$

Therefore, the fall detection risk can be written as:

$$\begin{cases} \Re(\alpha_1/O) = \omega_{12} P(O/\lambda_2) \\ \Re(\alpha_2/O) = \omega_{21} P(O/\lambda_1) \\ \Re(\alpha_3/O) = \omega \end{cases} \tag{13}$$

Finally, the optimal decision classes are:

– Choose λ_1 if: $\begin{cases} \Re(\alpha_1/O) < \Re(\alpha_2/O)\ and \\ \Re(\alpha_1/O) < \Re(\alpha_3/O) \end{cases}$
  Thus select λ_1 if: $\begin{cases} P(O/\lambda_1) > \frac{\omega_{12}}{\omega_{21}} \times P(O/\lambda_2) and \\ P(O/\lambda_1) > 1 - \frac{\omega}{\omega_{12}} \end{cases}$
– Choose λ_2 if: $\begin{cases} \Re(\alpha_2/O) < \Re(\alpha_1/O) and \\ \Re(\alpha_2/O) < \Re(\alpha_3/O) \end{cases}$
  Thus select λ_2 if: $\begin{cases} P(O/\lambda_2) > \frac{\omega_{21}}{\omega_{12}} \times P(O/\lambda_1) and \\ P(O/\lambda_2) > 1 - \frac{\omega}{\omega_{21}} \end{cases}$
– The reject option if: $\begin{cases} \Re(\alpha_3/O) < \Re(\alpha_1/O) and \\ \Re(\alpha_3/O) < \Re(\alpha_2/O) \end{cases}$
  Thus select reject if: $\begin{cases} P(O/\lambda_1) > \frac{\omega}{\omega_{21}} and \\ P(O/\lambda_2) > \frac{\omega}{\omega_{12}} \end{cases}$

9 Experimental results

9.1 Evaluation measures

To evaluate significantly the proposed methodology, different statistical measures are computed, such as the accuracy, the receiver operating characteristic (ROC) analysis, the area under curve (AUC) and the F-measure [15]. The ROC curve is the simplest graphical tool for illustrating classifier's performance. We recall that ROC plot sensitivity is relative to the true positive rate (TPR) against (1-specificity), which is relative to the false positive rate (FPR) over a range of a classification method's threshold. The TPR is obtained by the division of the number of correctely classified falls by the total number of fall incidents, while the FPR is given by the ratio of the number of misclassified non-falls by the total number of non-fall incidents.

The AUC is a significant statistical measure for evaluating a classifier, where an AUC of 1 corresponds to an ideal classification and an AUC value of 0.5 corresponds to a classification by chance. In this study, ROC curves were plotted and AUC values were then computed to compare classifications algorithms. We also used the F-measure to qualify the influence of random testing on the classification rate. The F-measure is computed by combination of recall (r) and precision (p) ratios, where the recall represents the ratio between the true positive number (tp) and the total number of samples in the positive class, while the recall represents the ratio between the true positive (tp) number and the sum of true positive and false negative elements. The F-measure is obtained by [15]:

$$F = (1 + \beta^2) \times \frac{p \times r}{(\beta^2 \times p) + r} \tag{14}$$

where

$$p = \frac{tp}{tp + fp}\ and\ r = \frac{tp}{tp + fn} \tag{15}$$

It is well known that recall and precision values can vary from 0 to 1, thus the F-measure tries to combine them into a single value to obtain certain effectiveness. In the present work, the parameter β is set to 1.

9.2 Dataset and experiments

To analyze the efficiency of the proposed model, two fall detection datasets have been conducted, namely: "Fall Detection Database" (FDD) [8] that comprises 191 video sequences of several actions performed in diverse ways. The frame rate is 25 frames/s and the resolution is 320 × 240 pixels. "URFD dataset" is the second data that has been used [18]. It comprises 70 (30 falls + 40 activities of daily living ADL) sequences of several actions performed in diverse ways. The images are acquired at 25 frames per second, and with a resolution of 640 × 480 pixels. In both data, videos are recorded from different environments and contain variable illumination like shadows and reflections that can be detected as moving objects. We have first conducted a posture identification task on the posture images, where a posture class is assigned to each human body image. To investigate whether the proposed classification is capable of identifying human postures; we manually denoted the ground truth of data samples for 5 posture classes. To evaluate the proposed method, we have selected 5000 images. The selected images should contain one of the five postures previously defined. These images were then divided into training and testing samples using a 3-fold cross-validation procedure. Furthermore, we have iteratively tested different SVM parameter settings (The parameters for RBF kernel: Cost C and Sigma σ). The pair that has produced a higher accuracy has been selected. We have also conducted a comparative task between two posture classifications. In the first one, only curvelet transforms are used as features, and in the second classification the five ratios have been included in the feature vector. On the other hand, to evaluate the fall detection system, more than 500 sub-videos sequences have been used. These sequences were also divided into learning and testing samples using a 5-fold cross-validation procedure and different numbers of iterations (HMM parameter) have been tested. The iteration number that has produced a higher accuracy has been selected. The crucial point of classification with reject decision remains to find a compromise among the error rate and the rejection rate. To discriminate losses between the misclassification decisions and reject option, we have tested during experiment different values. We have set $\omega = 0.3$, $\omega_{12} = 0.7$ and $\omega_{21} = 0.5$, the corresponding error rate and rejection rate are 0.02% and 0.016% respectively.

9.3 Results and interpretation

For the proposed posture classification, a confusion matrix is calculated (see Table 1), as well as the overall accuracy, F-measure and the Kappa coefficient, where their corresponding values achieved 95.26%, 0.908 and 0.9091, respectively. It is clear from these results that classification enables a robust identification even in challenging situations. It is also clear (from Table 1) that the bending class remains a challenge for the proposed approaches where it is characterized by the lowest classification accuracy (91.79%). In fact, the bending class is slightly confused with the sitting class as often as 6.19% of cases. This confusion is due to: (i) the important similarity presented by the two classes, and (ii) segmentation errors generally induced by the presence of shadows; or confusion of body limbs with environment objects. It is well-established that segmentation conducted in numerous areas is always error prone.

On the contrary, the standing and squatting classes are correctly classified in the most of cases, and the Lying class is perfectly accurate with 100% of its reference images; this result is explained by the incorporation of the area ratios with curvelet features in the classification process.

Table 1 Human posture classification results

		Reference data				
		Standing	Lying	Bending	Sitting	Squatting
Classified data	Standing	**96.17**	0	0.21	0.62	0.92
	Lying	0	**100**	0	0	0
	Bending	2.89	0	**91.79**	3.98	2.33
	Sitting	0.92	0	6.19	**93.31**	1.71
	Squating	0	0	1.81	2.09	**95.04**

Overall Accuracy = 95.26%;

Kappa Coefficient = 0.908; F-measure = 0.9091

(i) highliting the accuracies obtained by the proposed method and (ii) indicating the difference between the data parts used during the comparison

The ROC curve is also exploited during evaluation, where different values of HMM's iteration number were considered. Each variation generates different specificity and sensitivity values. Figure 7 illustrates two ROC curves obtained by classification using (i) only curvelet transforms as features and (ii) when the five ratios are added as features. Their AUC values are equal to 0.895 and 0.9178, respectively . One can clearly note that classification result with the additional five area ratios is more accurate.

We evaluated the SVM-HMM fall detection and compared it with KNN, Neural Network, Naive Bayes and SVM classifiers. To allow a fair comparison, the same video sequences are used for evaluation. Table 2 compares the proposed classification strategy with that using the other classifiers, where overall accuracy, F-measure, and the average testing times are calculated. The results shown in Table 2 demonstrate that SVM-HMM classification is more accurate at detecting falls than any classifier alone. In term of computation time, the training time is not too of importance compared to the testing time. Since the training is done off-line unlike the testing phase which is done on-line. According to Table 2, the neural network classifier has a lower processing time than does Naive Bayes or KNN classifiers, and SVM classifier is considered to be the shortest classifier for this application. However,

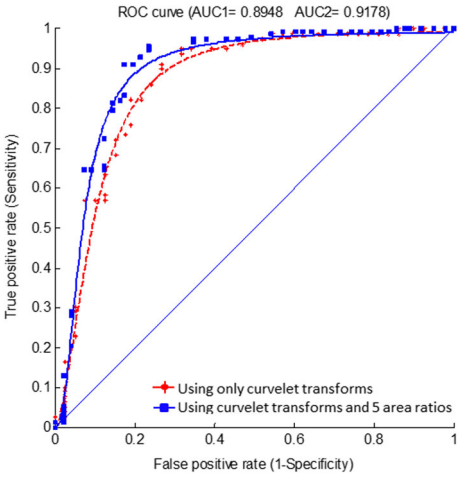

Fig. 7 ROC curves of fall classification using: (i) only curvelet transforms and (ii) curvelet transforms with the five ratios

Table 2 Performance comparison between HMM-SVM, KNN, Neural network and Naive Bayes algorithms

Classifier	Overall accuracy (%)	F-measure	Testing time (s)
KNN	91.95	0.91	0.22
Neural network	94.92	0.94	*0.15*
Naive Bayes	93.56	0.93	*0.31*
SVM	95.28	0.95	*0.062*
The proposed method	**96.88**	**0.96**	***0.12***

(i) highliting the accuracies obtained by the proposed method and (ii) indicating the difference between the data parts used during the comparison

compared to all of these classifiers HMM-SVM combination remains significantly better, since it presents the highest accuracy and an acceptable processing time.

Finally, to measure the real contribution of the present work, comparisons with some existing systems conducted on FDD and URFD datasets are reported in Tables 3 and 4 respectively. Where several powerful classifiers are used namely: neural network [3, 21], upper and lower thresholding [6], KNN [18], SVM [17, 40], and DAGSVM [39]. To allow a fair comparison, the percentage of rejected cases was subtracted from the final accuracy. From both Tables 3 and 4, the results demonstrate the outperformance of the proposed approach over detecting falls by the state-of-the-art approaches, even in critical scenarios. Furthermore, one can note that all like-fall events (as intentional lying or missing step) are correctly classified as non-fall activities, even if some sleeping gestures are made so quickly.

The reason that the SVM-HMM combination outperformed the neural network [2] is that SVMs utilize a geometric aspect (in postures separation) and provide a sparse solution via structural risk minimization, unlike neural networks, which is founded on empirical risk minimization. In addition, the proposed method can deal with the major problem than ANNs, as SVM and HMM are less prone to over fitting. One can also note, that detection

Table 3 Performance of the proposed fall detection system on FDD database

	The extracted features	Approach	Overall accuracy
L. Alhimale et al. [2]	Height and width of the bounding box	Neural network	94.27
X. Ma et al. [21]	Silhouette, lighting, and flow features	VPSO-neural network	92.17
A. Bourke et al. [6]	Upper and lower peak values	Upper fall threshold (UFT)	95.00
		Lower fall threshold (LFT)	86.67
M. Yu et al. [39]	Projection, histogram along the axes of the ellipse	DAGSVM	96.09
The proposed method	**Curvelet transforms and area ratios**	**SVM-HMM**	**97.02**

(i) highliting the accuracies obtained by the proposed method and (ii) indicating the difference between the data parts used during the comparison

Table 4 Performance of the proposed fall detection system on URFD database

	The used features	Approach	Data used	Overall Accuracy
B. Kwolek et al. [17]	Accumulator of the Hough transform applied on v-disparity values	SVM	*using only RGB data*	90.00
		SVM	*using RGB, accelerometer and depth data*	98.33
B. Kwolek et al. [18]	Depth features and, P40 descriptor	SVM	*using RGB, accelerometer and depth data*	94.28
		k-Nearest Neighbor	*using RGB, accelerometer and depth data*	95.71
Y. Yun et al. [40]	Velocity statistics, based on geodesic distances on the manifold	SVM	*using RGB and v*	*96.77*
The proposed method	**Curvelet transforms and area ratios**	***SVM-HMM***	***using only RGB data***	***96.88***

(i) highliting the accuracies obtained by the proposed method and (ii) indicating the difference between the data parts used during the comparison

based on lower and upper thresholds [6] presented several misclassification cases compared to SVM-HMM classification. This misclassification is due to the dependence of threshold values on the video sequences, where no learning step is performed. HMM-SVM combination has also outperformed K-nearest neighbor fall classification [18] due to the ability of SVM and HMM to train important data sets, compared to the complexity of k-NN algorithm in searching the nearest neighbors for each video sample.

In fact, even in the case of HMM-SVM combination, some intentional lyings are missclassified as falls. This confusion is due to the dark environment (degraded illumination conditions). Since the proposed approach is completely based on information extracted from RGB images, the performance may decrease if there is degradation of image quality. Indeed, it will be hard to extract the silhouette image of the human body from the conventional RGB camera in a dark condition, Specifically when a silhouette of the human body becomes unclear or masked by darkness (refer to Fig. 8).

Otherwise, the fall identification with SVM presented in [17] (using depth data, accelerometer measures, and RGB camera information) achieved the highest accuracy (98.33%). However, processing with sequences of sensors, RGB and depth information is considerably more computationally intensive, which is a limitation for real time applications. On the other hand, the SVM-HMM combination has outperformed the rest of SVM fall detection methods [39, 40]. This observation confirms that the SVM-HMM formalism is more adapted to fall detection application than other classification methods.

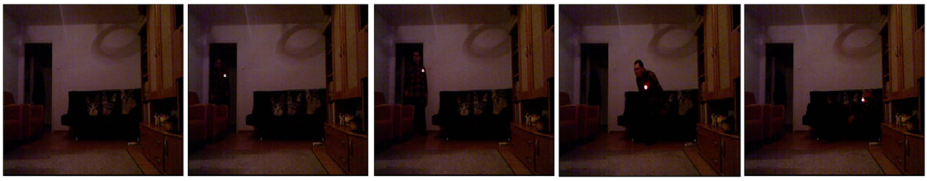

Fig. 8 Example of missclassified video sequence due to the dark environment

In particular, the fusion of curvelet transforms and area ratios adequately described movements made by the human body. Due to the translation and scaling invariance, their application is adapted to all video sequence types, and no prior camera calibration is required. The SVM posture classification has also helped to define meaningfull of HMM's states. In addition, the sequential information and duration, which are considered in HMM, are necessary to separate between real falls and like-fall activities. Distinguishing these two activities is strongly linked to the transition time between "lie" and the other postures. However, in the presented state-of-the-art techniques, time information was not exploited by the classifier formalisms. Finally, the incorporation of the reject possibility allows for a further reduction in the error rates, when mistakes are too costly like misclassifying fall sequences, error rates are reduced by including a reject decision. Because rejecting a sequence is not as disastrous as missing a fall and an appropriate practical response may be assigned by the monitoring center.

10 Conclusion

In this work, pattern recognition techniques and video camera monitoring are both used to design reliable fall detection strategy. The human body characterization was performed by the introduction of curvelet coefficients and body area ratios as attributes. First, a posture identification was performed using an SVM classifier. Then, to discriminate fall events from other daily activities, an HMM model with reject decision was applied. The experimental validation was applied on real frame sequences of consisting of daily activities and falls. The obtained result showed that the algorithm allows reliable fall detection with a reduced error rate. The complete system, comprising human body segmentation, feature extraction and selection, posture classification and fall identification is working in real time. This system can be applied to home care emergency detection when old people suddenly fall or remain in the lying posture.

As future work, we plan to investigate the viability of using thermal or infrared cameras to further enhance the effectiveness of fall detection system in the dark, since RGB cameras are not capable of distinguishing human silhouettes in such environment conditions.

References

1. Ageing WHO, Unit LC (2008) WHO global report on falls prevention in older age. World Health Organization
2. Alhimale L, Zedan H, Al-Bayatti A (2014) The ation of an intelligent and video-based fall detection system using a neural network. Appl Soft Comput 18:59–69
3. Anderson D, Keller JM, Skubic M, Chen X, He Z (2006) Recognizing falls from silhouettes. In: IEEE international conference of engineering in medicine and biology society, pp 6388–6391
4. Auvinet E, Multon F, Saint-Arnaud A, Rousseau J, Meunier J (2011) Fall detection with multiple cameras: An occlusion-resistant method based on 3-d silhouette vertical distribution. IEEE Trans Inf Technol Biomed 15(2):290–300
5. Baum LE, Petrie T, Soules G, Weiss N (1970) A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Ann Math Stat 41(1):164–171
6. Bourke AK, O'Brien JV, Lyons GM (2007) Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm. Gait Posture 26(2):194–199
7. Chang CC, Lin CJ (2011) LIBSVM: a library for support vector machines. ACM Trans Intell Syst Technol 2(3):27
8. Charfi I, Miteran J, Dubois J, Atri M, Tourki R (2012) Definition and performance evaluation of a robust svm based fall detection solution. In: International conference on signal image technology and internet based systems, pp 218–224

9. Chen YN, Chuang CH, Yu CC, Fan KC (2014) Fall detection in dusky environment, embedded and multimedia for human-centric computing in advanced technologies. Springer, Netherlands, pp 1131–1138
10. Chien SY, Ma SY, Chen LG (2002) Efficient movingobject segmentation algorithm using background registration technique. IEEE Trans Circuits Syst Video Technol 12(7):577–586
11. Das S, Suganthan PN (2011) Differential evolution: a survey of the state-of-the-art. IEEE Trans Evol Comput 15(1):4–31
12. Doukas C, Maglogiannis I, Katsarakis N, Pneumatikakis A (2009) Enhanced human body fall detection utilizing advanced classification of video and motion perceptual components. In: International conference on artificial intelligence applications and innovations, pp 185–193. Springer US
13. Foroughi H, Aski BS, Pourreza H (2008) Intelligent video surveillance for monitoring fall detection of elderly in home environments. In: IEEE international conference on computer and information technology, pp 219–224
14. Giannakouris K (2008) Ageing characterises the demographic perspectives of the European societies. Statistics in focus, p 72
15. Hand DJ (2012) Assessing the performance of classification methods. Int Stat Rev 80(3):400–414
16. Hiremath PS, Shivashankar S, Pujari J (2006) Wavelet based features for color texture classification with application to CBIR. Int J Comput Sci Netw Secur 6(9A):124–133
17. Kwolek B, Kepski M (2014) Human fall detection on embedded platform using depth maps and wireless accelerometer. Comput Methods Prog Biomed 117(3):489–501
18. Kwolek B., Kepski M. (2015) Improving fall detection by the use of depth sensor and accelerometer. Neurocomputing 168:637–645
19. Lee T, Mihailidis A (2005) An intelligent emergency response system: preliminary development and testing of automated fall detection. J Telemed Telecare 11(4):194–198
20. Li Y, Ho KC, Popescu M (2012) A microphone array system for automatic fall detection. IEEE Trans Biomed Eng 59(5):1291–1301
21. Ma X, Wang H, Xue B, Zhou M, Ji B, Li Y (2014) Depth-based human fall detection via shape features and improved extreme learning machine. IEEE J Biomed Health Inform 18(6):1915–1922
22. Ma Z, Yang Y, Sebe N, Hauptmann AG (2014) Knowledge adaptation with partially shared features for event detection using few exemplars. IEEE Trans Pattern Anal Mach Intell 36:1789–1802
23. Majumdar A (2007) Bangla basic character recognition using digital curvelet transform. J Pattern Recognit Res 2(1):17–26
24. Malathi T, Bhuyan MK, Local Gabor wavelet-based feature extraction and evaluation (2016) International conference on advanced computing, networking and informatics, pp 181–189
25. Muller M, Roder T (2006) Motion templates for automatic classification and retrieval of motion capture data. In: ACM SIGGRAPH/Eurographics symposium on computer animation, pp 137–146
26. Olivieri DN, Conde IG, Sobrino XAV (2012) Eigenspace-based fall detection and activity recognition from motion templates and machine learning. Expert Syst Appl 39(5):5935–5945
27. Rabiner L, Juang BH (1993) Fundamentals of speech recognition
28. Rougier C, Meunier J (2010) 3D head trajectory using a single camera. In: International conference on image and signal processing. Springer Berlin, pp 505–512
29. Rougier C, Meunier J, St-Arnaud A, Rousseau J (2007) Fall detection from human shape and motion history using video surveillance. In: International conference on advanced information networking and applications workshops, pp 875–880
30. Shoushtarian B, Bez HE (2005) A practical adaptive approach for dynamic background subtraction using an invariant colour model and object tracking. Pattern Recogn Lett 26(1):5–26
31. Starck J, Candes EJ, Donoho L (2002) The Curvelet transform for image denoising. IEEE Trans Image Process 11:670–684
32. Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. J Glob Optim 11(4):341–359
33. Ucar A, Demir Y, Guzelis C (2016) A new facial expression recognition based on curvelet transform and online sequential extreme learning machine initialized with spherical clustering. Neural Comput Appl 27(1):131–142
34. Vapnik VN, Vapnik V (1998) Statistical learning theory, vol 1. Wiley, New York
35. Vladimir VN, Vapnik V (1995) The nature of statistical learning theory
36. Yan C, Zhang Y, Dai F (2014) Parallel deblocking filter for HEVC on many-core processor. Electron Lett 50:367–368
37. Yan C, Zhang Y, Dai F, Li L (2013) Highly parallel framework for HEVC motion estimation on many-core platform. In: IEEE conference in data compression, pp 63–72

38. Yan C, Zhang Y, Dai F, Zhang J, Li L, Dai Q (2014) Efficient parallel HEVC intra-prediction on many core processor. Electron Lett 50:805–806
39. Yu M, Rhuma A, Naqvi SM, Wang L, Chambers J (2012) A posture recognition-based fall detection system for monitoring an elderly person in a smart home environment. IEEE Trans Inf Technol Biomed 16(6):1274–1286
40. Yun Y, Gu IYH (2015) Human fall detection via shape analysis on Riemannian manifolds with applications to elderly care. In: IEEE international conference on image processing, pp 3280–3284
41. Zerrouki N, Harrou F, Sun Y, Houacine A (2016) Accelerometer and camera-based strategy for improved human fall detection. J Med Syst 40(12):284
42. Zerrouki N, Houacine A (2014) Automatic classification of human body postures based on curvelet transform. In: International conference image analysis and recognition. Springer International Publishing, pp 329–337
43. Zhang D, You X, Wang P, Yanushkevich SN, Tang YY (2009) Facial biometrics using nontensor product wavelet and 2d discriminant techniques. Int J Pattern Recognit Artif Intell 23:521–543
44. Zigel Y, Litvak D, Gannot I (2009) A method for automatic fall detection of elderly people using floor vibrations and sound—proof of concept on human mimicking doll falls. IEEE Trans Biomed Eng 56(12):2858–2867

Nabil Zerrouki was born in Algiers, Algeria. He received the engineer degrees and Master in Electrical Engineering from the University of Sciences and Technology Houari Boumédienne, Algiers, Algeria. His current research interests are computer vision, image processing and their applications to scene analysis and human-machine interaction.

Amrane Houacine was born in Algeria. He is a professor at the Faculty of Electronics and Informatics, University of Sciences and Technology Houari Boumédienne of Algiers, Algeria. His current research interests include signal and image processing and their applications to human-machine interaction.