Multimed Tools Appl (2019) 78:9101–9128
https://doi.org/10.1007/s11042-018-5638-9

# Human fall detection using slow feature analysis

Kaibo Fan[1] · Ping Wang[1] · Shuo Zhuang[1]

Received: 5 January 2017 / Revised: 11 December 2017 / Accepted: 10 January 2018 /
Published online: 24 January 2018

**Abstract** Falls are reported to be the leading causes of accidental deaths among elderly people. Automatic detection of falls from video sequences is an assistant technology for low-cost health care systems. In this paper, we present a novel slow feature analysis based framework for fall detection in a house care environment. Firstly, a foreground human body is extracted by a background subtraction technique. After morphological operations, the human silhouette is refined and covered by a fitted ellipse. Secondly, six shape features are quantified from the covered silhouette to represent different human postures. With the help of the learned slow feature functions, the shape feature sequences are transformed into slow feature sequences with discriminative information about human actions. To represent the fall incidents, the squared first order temporal derivatives of the slow features are accumulated into a classification vector. Lastly, falls are distinguished from other daily actions, such as walking, crouching, and sitting, by the trained directed acyclic graph support vector machine. Experiments on the multiple-camera fall dataset and the SDUFall dataset demonstrate that our method is comparable to other state-of-the-art methods, achieving 94.00% recognition rate on the former dataset and 96.57% on the latter one.

**Keywords** Slow feature analysis · Support vector machine · Fall detection · Background subtraction · Human silhouette

## 1 Introduction

Human action recognition is one of the major topics in computer vision community. It has a wide application prospect in the related fields such as abnormal behavior analysis, especially

✉ Kaibo Fan
kaibofan@126.com; fankaibo@tju.edu.cn

[1] School of Electrical and Information Engineering, Tianjin University,
Tianjin, People's Republic of China

the vision-based fall detection, video surveillance, and human-computer interaction [32, 39]. Fall detection is one of the most actively discussed topics, many works on fall event recognition have been reported in [8, 17, 18, 28].

Falls of the elderly people are one of the major health problems because they cause many injuries or even death. According to the statistics from the World Health Organization [1] approximately 28.00% to 35.00% of people aged 65 and above suffer a fall accident every year, increasing to 32.00% to 42.00% for those over 70 years of age. This makes falls one of the five most common causes of death among the elderly people [29]. In a fall incident, they typically receive moderate to severe injuries such as bruises, hip fractures or head trauma. Moreover, billions of fall-related health and medical costs become a heavy burden not only in China but also in other countries with the ageing population problem. A quick response to a fall incident has already been proved to be a critical step to reduce the medical appliance charges for a fallen person. Therefore, the fall detection, fall prevention and the protection of a person living alone have recently become significant research topics for many scientists all over the world.

In recent years, the number of proposed fall detection systems and algorithms has increased rapidly. An overview of this topic can be found in [13, 45]. Very recent review in [10, 22] highlights the challenges and issues of multisensory approach in this field. Furthermore, the work [21] presents the taxonomy of fall detection from the perspective of the availability of the fall data. Wireless body area network (WBAN) as the moving platforms for pervasive computing and communication has been widely applied in healthcare domains. Hassan et al. [14] propose an efficient network model that combines WBAN and Cloud to deliver and share the media healthcare data to remote terminals with the quality of service support. Health Internet of Things (IoT) [19], makes various medical devices, sensors, and diagnostic and imaging devices the final building blocks in the development of smart healthcare frameworks. In the future, the integrated healthcare models adopted to consolidate the fragmented care services will be enhanced to deliver personalized and precise healthcare tailored to particular individuals. Lei Meng et al. [26] present a novel framework named the online daily habit modeling and anomaly detection model for the real-time personalized activities of daily living (ADLs) recognition, habit modeling and anomaly detection for the solitary elderly. Moreover, their system can obtain very minute details of the detected activities. Alternatively, the recent developments in sensor technology have made the deployment of sensors in various environments. Multiple fall detection approaches using distributed sensors based ambient sensing technologies have also been developed and studied [12, 40]. These systems make it possible to prevent many other health hazard situations apart from fall accidents.

At present, fall detection systems can be based on sensors mounted at home, such as cameras [36, 45], pressure sensors [25], sound sensors [20], or sensors carried by the users [34, 35]. Some other researchers have also turned to use radar signals for fall detection [2]. Recently, most of the commercial types of fall detection systems are based on wearable sensors. The key problem, however, is that the elderly may easily forget to wear it. Besides, it is not convenient to carry these devices all the time. Yet, there are some advantages that make wearable sensors, especially the acceleromometer-based devices, quite popular. The first one is that they measure the body's motion parameters directly. However, there is a new trend towards using a computer vision-based approach in the fall recognition [18], and this is the solution we applied in our work. This approach is indispensable for creating a smart home environment where it is possible to detect, analyze and even give an alarm about some deviations from the normal course of the daily life activities of the monitored person through installing digital video cameras in the rooms.

According to the types of cameras used in the fall detection systems, they can be broadly divided into two categories: the multiple-camera systems and the monocular vision systems [45]. In general, multiple cameras offer the advantages of allowing three-dimensional reconstruction and extraction of three-dimensional features for the fall detection. However, the calibration process of a multi-camera system is very complicated. It is a challenging and time-consuming task. The monocular vision-based approach plays an irreplaceable role in the fall detection systems because of its undeniable merits. These systems are very cheap and easy to set up. Moreover, many other activities except fall events can be detected simultaneously with less intrusion. Practically, a fall incident usually occurs in a very short period. The typical duration of a fall incident is about $0.4 \sim 0.8$ seconds. During this short time, the human posture changes considerably with a high velocity. It is not easy to recognize a fall among daily life activities, especially among such movements as sitting down and crouching down. The two actions have characteristics similar to the fall incident, but have entirely different semantic contents. Since the shape-related features are not enough to distinguish these similar motions accurately, we also consider the temporal information between postures in our method.

This paper proposes a novel slow feature analysis based framework for the fall detection, inspired by the temporal slowness principle. According to the temporal slowness principle in [41], slow feature analysis (SFA) can be used to extract the invariant and slowly varying features from the rapidly changing input signals for encoding the discriminative local features. The slow features contain a high-level semantic content, which can be discovered by the input-output mapping functions. To the best of our knowledge, this is the first work that discusses the application of slowness principle in the fall detection. Our main contributions are summarized as follows:

- The foreground human body extraction is reported in Section 3.1. The human silhouette is refined and covered by a fitted ellipse, which is more compact than the traditional rectangular box. The approximated ellipse is more suitable for describing the human posture than the bounding box.
- In order to describe the postures of a fall incident, six shape features are extracted from the covered silhouette. The sequences of quickly changing shape features are transformed into slow feature sequences with a high level of discriminative ability for a fall accident recognition. To represent the fall accidents, we accumulate the squared first order temporal derivatives of the slow features into a classification vector.
- As for distinguishing a fall incident from other daily activities, the directed acyclic graph strategy is utilized to combine several binary classification SVM for human actions classification. The decision process is based on a sequence of two-class operations, i.e., from the root node to the leaf node. Once a bottom node is reached, the final decision is made.

The rest of the paper is organized as follows. Section 2 provides an overview of the related works. In Section 3, we give a detailed description of our proposed method, which includes the human body extraction, shape features for posture representation, slow feature analysis, and the fall detection through classification. The experimental evaluations are carried out in Section 4. Finally, Section 5 contains the conclusions.

## 2 Related works

Previous works on video-based fall detection are numerous. In this section, we mainly focus on monocular vision fall detection systems. In such a system, human body shape analysis (silhouette analysis), inactivity detection, and head motion analysis are widely used for fall detection.

Monocular vision fall detection systems are usually based on calculating different features of the object under surveillance. More specially, the commonly used features are foreground object's projected height-width ratio, the difference and the centroid of the foreground object. For example, Liu et al. [23] employ the ratio and difference of the width and height of a silhouette bounding box to classify postures into three categories: the standing posture, the temporary posture, and the lay down posture. Meanwhile, a critical time difference is obtained and verified by the statistical hypothesis testing. With the help of the classified postures and critical time difference, the performance of their system is promising when the camera is placed sideways. Yu et al. [43] applies a background subtraction technique to extract the foreground objects. The moving object is located in the image plane by an ellipse whose parameters are obtained by computing the spatial moments of the foreground image. A projection histogram constructed along the axis of the ellipse (the local features) and the ratio between the major axis and the minor axis (the global features) can evidently distinguish the postures of a fall event. Lastly, the support vector machine (SVM) classifier is employed to perform the classification based on the local and global features.

However, the sideward camera mounting makes the fall detection systems sensitive to occlusions. The cameras should be located in a high place to avoid this situation. For real house care environment, the camera's view should cover a vast area. Moreover, human motion velocity is significantly influenced by the distance to the camera. To cope with these problems, Marc et al. [7] propose a low-cost fall detection system based on a single wide-angle camera. Wide-angle cameras are used to reduce the number of cameras required for monitoring a large area. Features based on the gravity vector are introduced to detect the falls. The occlusion problem is also partly solved. Olivier et al. [30] present a spatio-temporal motion representation that captures relevant velocity information by extracting the dense optical flow from a video sequence. Falls can be distinguished with high accuracy and computational efficiency from other daily human actions by introducing both the magnitude and direction of the velocity into a motion vector flow instance template.

Only one fixed camera does not accurately detect falls that occur in various directions. Usually, there are three ways to solve a perspective problem. The first one is based on finding the invariant features of the object [9, 27]. The second way is to learn discriminative features from a training set that contains different views' samples [38, 44]. The third one is to use the depth data directly to prevent perspective problems [3, 24, 42]. For instance, Mirmahboub et al. [27] utilize the variations in the silhouette area of a human to detect falls. The variations of the silhouette area are view-invariant features. Wang et al. [38] propose a new fall detection framework, based on the automatic feature learning method. The frames extracted from video sequences of different views form a training set. A principle component analysis net model is trained by using the training set to predict the label of every frame. A fall event model is further obtained by SVM with the predicted labels of frames in the video sequences. The authors of [3, 24] overcome the perspective problem by presenting a more sophisticated approach based on depth images. In work [24], the curvature scale space (CSS) features and the bag-of-words (BOW) method are combined to detect fall incidents in a depth video. An improved extreme learning machine (ELM) classifier is adopted to distinguish falls and non-falls. In the later work [3], instead of representing an action as a bag of CSS words, the Fisher vector (FV) encoding is used to describe the action based on CSS features. A pre-trained SVM classifier is employed to make the final classification.

Although our system uses a single monocular camera, it overcomes the aforementioned problems by extracting slow features of a moving foreground object. The foreground object is firstly described by six shape features. And then, slow features are derived by analyzing the input shape features and its time derivative. Besides, temporal information between postures is also utilized. Moreover, these slow features are ordered by their degree of invariance. The experiments conducted on a public multi-view fall detection dataset have shown that our method can achieve good results.

## 3 The proposed method

Our fall detection system includes four phases: human body extraction, shape features for posture representation, slow feature analysis, and fall detection through classification. The overall framework of our method is presented in Fig. 1.

The first step of our approach is to detect the elderly person in every video frame. It is accomplished by applying the background subtraction model and pixel classification. Once the person has been detected, the silhouette is fitted by an ellipse after the mathematical morphology operations. In order to describe human postures, six shape features are developed from the covered silhouette. However, postures of a fall incident change considerably in a very short period of time. Most of the traditional vision-based methods just utilize some shape-related features and neglect the temporal information. To take advantage of the temporal information, we represent fall incidents as shape feature sequences and analyze them using slow feature analysis. Moreover, inspired by the observation that slow features contain high-level semantic information, we learn the slow feature analysis functions to explore the slow features during the training stage. In the testing stage, these SFA functions are introduced to transform the input shape feature sequences into slow feature sequences. Falls are described by accumulating the squared first order temporal derivatives of the slow features. Finally, falls can be detected by the trained directed acyclic graph support vector machine. Meanwhile, other normal actions, such as walking, crouching and sitting, can also be classified by the trained classifier.

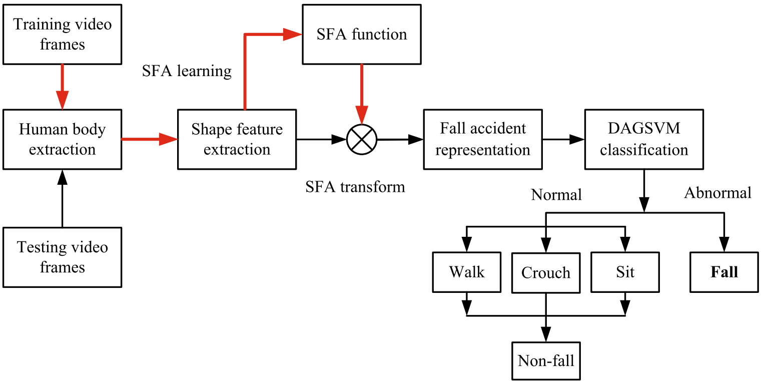


**Fig. 1** The framework of our approach

### 3.1 Human body extraction

Human bodies are extremely non-rigid objects with a high degree of variability in size and shape. When people walk towards or away from a video camera both the shape and size of a human body change greatly. Sometimes, the color and texture are affected significantly by the shadow or ambient light in a living room. Because of these peculiarities, the task of extracting a moving person from an image sequence is one of the most challenging ones in computer vision field. As the shape of a human body is a very significant clue to human motion analysis, there should be a human body extraction technique capable of coping with all the problems listed above. Therefore, some approaches focused on solving these problems have recently been developed. Among them are visual background extractor [5], local binary pattern histogram [15] and so on. They are effective in dealing with problems of illumination changes and with dynamic scenes.

In the video sequence, there is only one moving object, the walker. Besides, there is background that is always static and the camera that does not move. Under this assumption, the background model is obtained by computing several model parameters over some static background frames. We employ the color distortion model proposed by Horprasert et al. [16], which can be used to deal with the problem of slight illumination changes, such as shadows and highlights.

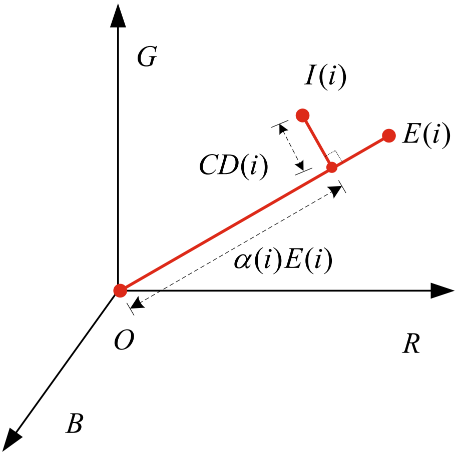
Having adopted the color distortion model, we separate the brightness from chromaticity component. Figure 2 shows the color distortion model in the three-dimensional RGB color space.

Considering a pixel $i$ in the frame, let $E(i) = [E_R(i), E_G(i), E_B(i)]$ represent the pixel's expected RGB color value in the background model, and let $I(i) = [I_R(i), I_G(i), I_B(i)]$ denote the pixel's RGB color value in the current image that needs to be subtracted from the background. The distortion of $I(i)$ from $E(i)$ is decomposed into two parts, namely, brightness distortion $BD(\alpha(i))$ and color distortion $CD(i)$. The brightness distortion $BD(\alpha(i))$ is a scalar value that brings the observed color value close to the expected chromaticity line. It is obtained by minimizing

$$BD(\alpha(i)) = ||I(i) - \alpha(i)E(i)||_2^2 \tag{1}$$

$\alpha(i)$ represents the pixel's brightness strength with respect to the expected pixel value. $\| \cdot \|_2$ stands for the two-norm. The color distortion of pixel $I(i)$ is defined as the dis-

**Fig. 2** Color distortion model



tance between the observed color and the expected chromaticity line which is given by

$$CD(i) = ||I(i) - \alpha(i)E(i)||_2 \tag{2}$$

There are three main steps in human body extraction. The first step is to construct a reference background image using the background model. Second, the threshold selection step determines the appropriate threshold values for pixel classification. The last step is to classify the pixels into the background mask, moving object mask, and shadow mask.

After the three steps, the binary foreground image is obtained. However, the foreground image may be corrupted by bad noises both inside and outside of the object. The noises even make some small holes in the object. The morphological operations are implemented to remove the noise. The crucial steps of human body extraction are shown in Fig. 3.

## 3.2 Shape features for posture representation

After obtaining the foreground region of the human, the analysis of the human silhouette should be performed to extract discriminate features necessary for the body shape change detection. The highly non-rigid moving human body should be exactly identified in the frames. Practically, a foreground object is usually separated into several small blocks, since it rapidly moves across the similar background along with the human body. We gather all the extracted pixels together into a point set. Then, the identification of the human body in the image plane is carried out by fitting an ellipse, so that it covers the pixels of the foreground object. An ellipse is determined by four parameters: its center $(\bar{x}, \bar{y})$, orientation $\theta$, and the lengths $a$ and $b$ of its major semi-axis and minor semi-axis. The parameters of the ellipse that cover most of the points are obtained by calculating the first and second moments of the data points.

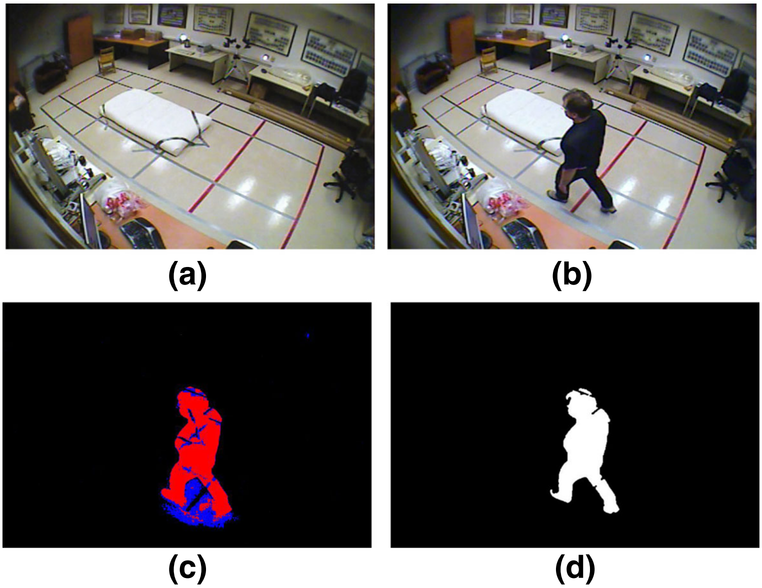


**Fig. 3** Human body extraction. **a** An original background image. **b** The current image with a human body. **c** The extracted human body in the foreground including shadows and holes. **d** The final extracted human object after the morphological operations

### *3.2.1 Parameters of the ellipse*

To compute the parameters of the ellipse, we firstly introduce two useful definitions to the computation, and then use these parameters to calculate the shape features for posture representation in the next subsection.

**Definition 1** (moments)
For a continuous image $f(x, y)$, the moments are given by

$$m_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x^p y^q f(x, y) dx dy \tag{3}$$

for $p, q = 0, 1, 2, \ldots$

The center of the ellipsoid $(\bar{x}, \bar{y})$ is obtained by computing the coordinates of the center of mass with the first and zero order spatial moments

$$\bar{x} = m_{10}/m_{00} \tag{4}$$

$$\bar{y} = m_{01}/m_{00} \tag{5}$$

**Definition 2** (central moments)
For a continuous image $f(x, y)$, and its centroid $(\bar{x}, \bar{y})$, the central moments are computed as follows

$$\mu_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x - \bar{x})^p (y - \bar{y})^q d(x - \bar{x}) d(y - \bar{y}) \tag{6}$$

for $p, q = 0, 1, 2, \ldots$

The angle $\theta$ between the major semi-axis $a$ and the horizontal axis $x$ gives the orientation of the ellipse. It can be computed with the central moments of the second order

$$\theta = \frac{1}{2}\arctan(\frac{2\mu_{11}}{\mu_{20} - \mu_{02}}) \tag{7}$$

To recover the major semi-axis $a$ and the minor semi-axis $b$ of the ellipse, we have to compute $I_{min}$ and $I_{max}$, i.e., the least and the greatest moments of inertia, respectively. They can be obtained by evaluating the eigenvalues of the covariance matrix [33]

$$J = \begin{pmatrix} \mu_{20} & \mu_{11} \\ \mu_{11} & \mu_{02} \end{pmatrix} \tag{8}$$

The eigenvalues $I_{min}$ and $I_{max}$ are given by

$$I_{min} = \frac{\mu_{20} + \mu_{02} - \sqrt{(\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2}}{2} \tag{9}$$

$$I_{max} = \frac{\mu_{20} + \mu_{02} + \sqrt{(\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2}}{2} \tag{10}$$

Then the major semi-axis $a$ and the minor semi-axis $b$ of the best fitting ellipse are given by

$$a = \left(\frac{4}{\pi}\right)^{1/4} \left[\frac{(I_{max})^3}{I_{min}}\right]^{1/8} \tag{11}$$

$$b = \left(\frac{4}{\pi}\right)^{1/4} \left[\frac{(I_{min})^3}{I_{max}}\right]^{1/8} \tag{12}$$

An example of an ellipse fitting result is depicted in Fig. 4, where we also present the compared rectangular fitting result provided in [9]. The approximated ellipse is obviously better in describing the human posture than the bounding box, especially in the presence of noise. The region of the human body covered by an ellipse is more compact than that of the rectangular box.

### *3.2.2 Feature extraction*

With the fitted region of a human silhouette $S$ at a given time $t$, several features are extracted to measure the shape deformation of the human body. The aspect ratio $AR_S(t) = b(t)/a(t)$, fall angle $FA_S(t) = \theta(t)$, and eccentricity $EC_S(t) = \sqrt{1 - \frac{a(t)^2}{b(t)^2}}$ of the fitted ellipse $S$ are used for describing the human shape deformation globally. The ratio between the major semi-axis and the minor semi-axis, the orientation and eccentricity of the ellipse $S$ provide much information about human body postures. If a fall happens in the direction of the optical axis, the aspect ratio $AR_S(t)$ decreases drastically while the fall angle $FA_S(t)$ remains the same. In other cases, the fall angle $FA_S(t)$ changes significantly while the aspect ratio $AR_S(t)$ and eccentricity $EC_S(t)$ do not change so much. Once a fall happens, these three features reflect the global deformation of a human posture. In a sense, they are view-dependent features.

Besides, local information of the silhouette should not be neglected. During the fall incident, there are various human postures. The area of a human posture plays a significant role in human activity analysis. We consider the relations between the area of a human posture in the fitted ellipse $A_H(t)$, the area of the ellipse $A_S(t)$ and even the whole image area $A_I(t) = MN$ ($M$, $N$ are the width and height of image $I$, respectively). The effective area $EA_S(t) = A_S(t)/A_I(t) = \pi a(t)b(t)/MN$ measures the occupancy of the fitted silhouette in the image plane. It is a good indicator of the self deformation of the human posture. When large deformation occurs, the effective area $EA_S(t)$ will change greatly to illustrate the distortion. Another feature is the affinity $AF_S(t) = \frac{A_H(t)}{A_S(t)}$. This feature describes the area of the human posture in the fitted ellipse $S$. The other feature is the roundness $RO_S(t) = \frac{4\pi A_S(t)}{p_S(t)^2}$. Here $p_S(t)$ is the perimeter of the ellipse S at a given time $t$. It mainly concerns the compactness of the human posture in the region covered by the fitted ellipse $S$. All the six features can be aggregated into a single feature vector $F_S(t)$. It is expressed as follows

$$[AR_S(t), FA_S(t), EA_S(t), AF_S(t), RO_S(t), EC_S(t)] = F_S(t) \tag{13}$$

These features $F_S(t)$ have been proved experimentally to be sufficient to describe the posture of a human body at a given time. One example sequence of the fall event is given in

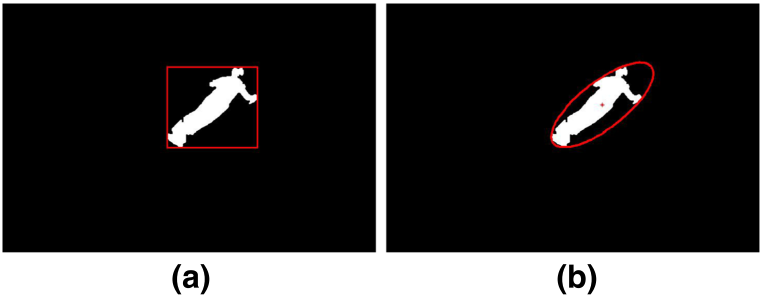


**Fig. 4** Region of interest. **a** Rectangular fitting. **b** Ellipse fitting

Fig. 5, where the process of human body extraction and ellipse fitting are shown. Figure 6 demonstrates the extracted features of the whole fall incident sequence. The circle illustrates the critical time of the fall incident. From the results, we can observe the distinct differences in the patterns between various postures. These features are helpful in describing a posture.

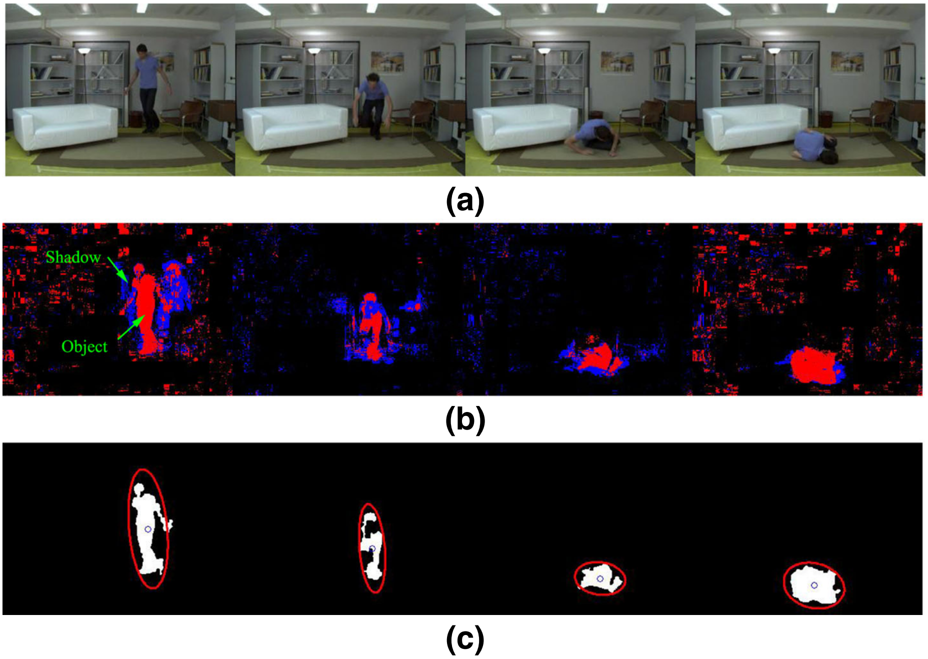
### 3.3 Slow feature analysis

The invariant features of temporally varying signals are useful for analysis and classification. Slow feature analysis is a new method for learning invariant or slowly varying features from a vectorial input signal. It is based on a nonlinear expansion of the input signal and on the application of the principle component analysis (PCA) to this expanded signal and its time derivative. It is guaranteed to find the optimal solution within a family of functions directly and can learn to extract a large number of decorrelated features, which are ordered by their degree of invariance.

Therefore, SFA is a potential candidate technique to be used in extracting approximately invariant features for the human fall event recognition. Mathematically, SFA can be defined as follows:

Given an $I$-dimensional input signal $X(t) = [x_1(t), \cdots, x_I(t)]^T$ with $t \in [t_0, t_1]$ indicates time, and $[\cdots]^T$ stands for the transpose of $[\cdots]$. SFA finds out an input-output function $G(x) = [g_1(x), \cdots, g_I(x)]^T$ so that the generated $J$-dimensional output signal $Y(t) = [y_1(t), \cdots, y_J(t)]^T$ with $y_j(t) = g_j(X(t))$ varies as slowly as possible, i.e., for each $j \in \{1, 2, \cdots, J\}$,

$$\Delta_j = \Delta(y_j) = \langle \dot{y}_j^2 \rangle_t \quad \text{is minimal} \tag{14}$$



(a)

(b)

(c)

**Fig. 5** Postures of a fall event. **a** Typical postures. **b** The extracted fall human body. **c** Ellipse fitting

subject to

$$\langle y_j \rangle_t = 0 \quad \text{zero mean} \tag{15}$$

$$\langle y_j^2 \rangle_t = 1 \quad \text{unit variance} \tag{16}$$

$$\text{and} \quad \forall j' < j : \quad \langle y_{j'} y_j \rangle_t = 0 \quad \text{decorrelation} \tag{17}$$

where $\dot{y}$ denotes the operator of computing the first order derivative of $y$, and the $\langle y \rangle_t$ indicates the temporal average of signal $y$ over time, that is, $\langle y \rangle_t = \frac{1}{t_1 - t_0} \int_{t_0}^{t_1} y(t)dt$. Equation (14) expresses the primary objective of minimizing the temporal variation of the output signal, where the temporal variation is measured by the temporal average of the squared first order derivative. Constraint (15) is presented here just for convenience so that constraint (16) and constraint (17) can take a simple form. Constraint (16) means that the output signal should carry some useful information and avoid the trivial solution $y_j(t) = \text{const}$. Constraint (17) ensures that different output signal components carry different types of information and do not simply reproduce each other. It also induces an order, where the first output signal $y_1(t)$ is the optimal signal, the slowest one, while $y_2(t)$ is the less optimal one, etc.

It is an optimization problem of variational calculus to find out the input-output function. In general, it is very difficult to solve this problem explicitly. If the input-output function components $g_j(x)$ are the constraint that is a linear combination of a finite set of nonlinear functions, i.e., $g_j(x) = w_j^T X$, wherein $X$ is the input signal vector and $w_j$ is the normalized weight vector, it becomes the generalized eigenvalue problem. This problem becomes much more simple and the solution of SFA becomes equivalent to the generalized eigenvalue problem [41].

$$AW = BW\Lambda \tag{18}$$

where $A = \langle \dot{X}\dot{X}^T \rangle_t$ is an expectation of the covariance matrix of the temporal first order derivative of the input signal vector $X$, $B = \langle XX^T \rangle_t$ is an expectation of the covariance

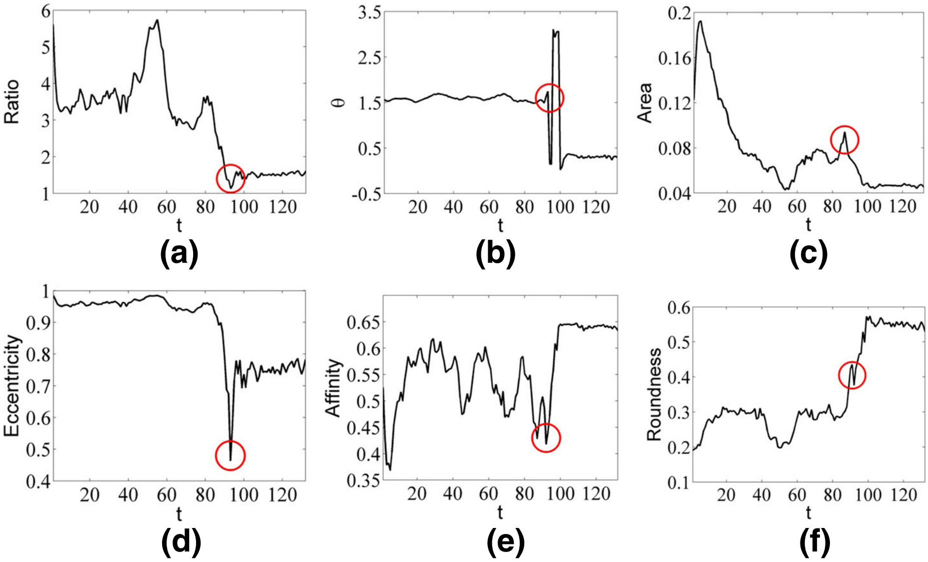


**Fig. 6** The extracted six shape features. **a** Aspect ratio. **b** Fall angle. **c** Effective area. **d** Eccentricity. **e** Affinity. **f** Roundness

matrix of the same input vector $X$, $\Lambda$ is a diagonal matrix of the generalized eigenvalues and $W$ is corresponding generalized eigenvectors. Besides, the orders of the slow features are determined by the eigenvalues. The most slowly varying signal has the smallest index.

As for the nonlinear transformation of $g_j(x)$, it can be deemed as the linear transformation in a nonlinear expansion space [41], each component of which is a weighted sum over a set of $K$ nonlinear functions $h_K(x)$, i.e., $g_j(x) = \sum_{k=1}^{K} w_{jk}h_k(x)$. Usually, $K > \max(I, J)$. The nonlinear function $H(x)$ can be defined as

$$H(x) = [h_1(x), \cdots, h_K(x)] \tag{19}$$

For example, a quadratic expansion for a two-dimensional input $X = [x_1, x_2]$ is $H(x) = \left[x_1, x_2, x_1^2, x_1x_2, x_2^2\right]$. Here all monomials of degree one and two including mixed terms such as $x_1x_2$ are used. The dimensionality of $H(x)$ is $K = I + I(I + 1)/2$. This is a common technique to transform a nonlinear problem into a linear one. Afterwards, SFA can be operated in the expansion space to obtain nonlinear slow feature functions.

In summary, slow feature functions can be obtained by the following steps:

1. Nonlinear expansion

   Apply a nonlinear function $H(x)$ to expand the original input signal and to centralize $H(x)$

   $$Z = H(x) - H_0 \tag{20}$$

   where $H_0 = \langle H(x)\rangle_t$. The centralization makes the constraint (15) valid. Here, we use the quadratic expansion, i.e.,

   $$H(x) = [x_1, \cdots, x_I, x_1x_1, x_1x_2, \cdots, x_Ix_I]$$

2. Solve the generalized eigenvalue problem

   $$AW = BW\Lambda \tag{21}$$

   where $A = \langle \dot{Z}\dot{Z}^T\rangle_t$, $B = \langle ZZ^T\rangle_t$.

In a nutshell, to solve the optimization problem (14), it is sufficient to compute the covariance matrix of the input signals and their derivatives in the expanded nonlinear space and then solve the generalized eigenvalue problem from (21). The derivative of $Z(t)$ is computed by the linear approximation $\dot{Z}(t) \approx (Z(t + \Delta t) - Z(t))/\Delta t$. Assume the dimensionality of matrices $A$ and $B$ are $K$, the first $M$ eigenvectors $w_1, \cdots, w_M (M \ll K)$ associated with the smallest eigenvalues $\lambda_1 \leq \lambda_2 \leq, \cdots, \leq \lambda_M$ are the nonlinear slow feature functions $g_1(x), \cdots, g_M(x)$

$$g_j(x) = w_j^T(H(x) - H_0) \tag{22}$$

which satisfies the constraint (15)–(17) and minimizes the objective function (14).

Here, the input-output function computes the output signal instantaneously. Therefore, the slow variation of the output signal can be achieved by extracting aspects of the input signal that are inherently slow and useful for obtaining a high-level representation.

### 3.4 Fall detection through classification

There are three main steps in the SFA-based fall event recognition that is slow feature function learning, fall accident represenation, and classification.

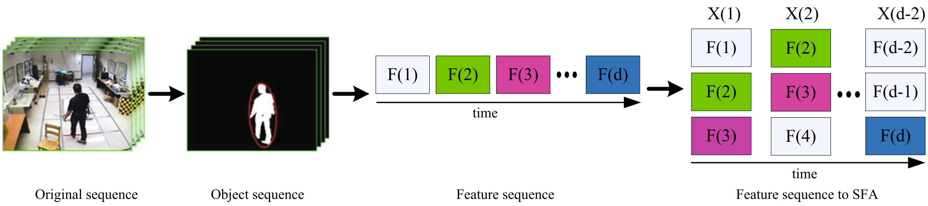


**Fig. 7** The reformatting process of the feature sequence

### *3.4.1 Slow feature function learning*

Before the learning, we should perform a preprocessing for each training image sequence to extract a human body and to calculate the shape features for postures representation. Usually, the fall incident occurs very quickly. The typical duration of an event is approximately 0.4 $\sim$ 0.8 seconds. That is about ten to twenty frames in the collected video samples. Having considered the average length of a fall event, we decided to select fifteen frames as one training sequence. Initialized at time $t$, a sample is obtained with the size $h \times w \times d$ ($1 \times 6 \times 15$ in this paper, i.e., one sample contains fifteen consecutive frames, and six shape features are extracted from each frame). According to [6], we reformat each input vector by $\Delta t$ ($\Delta t = 3$ in this paper). Figure 7 illustrates the reformatting process. The spatial-temporal information of the two neighbor frames can be learnt by the slow feature functions. After the reformatting step, the dimensionality $n$ of the input vector becomes $n = w \times \Delta t$. Before the non-linear expansion step of SFA, the dimensionality of the input signal increases greatly. The dimensionality of the quadratic expansion increases from $n$ to $n + n \times (n + 1)/2$ , i.e., $K = n + n \times (n + 1)/2$. It is not necessary to use the information of the full dimensionality. We apply principal component analysis to reduce the dimension of the original input feature $F_S(t)$ to three, which is sufficient for the subsequent experiment.

$$F(t) = W_{opt}(F_S(t) - \mu_S(t)) \tag{23}$$

where $W_{opt}$ is the optimal projection matrix and $\mu_S(t)$ is the mean of all $F_S(t)$, $F(t)$ is the final reduced dimension input feature vector. Figure 8 illustrates the percentage of variance

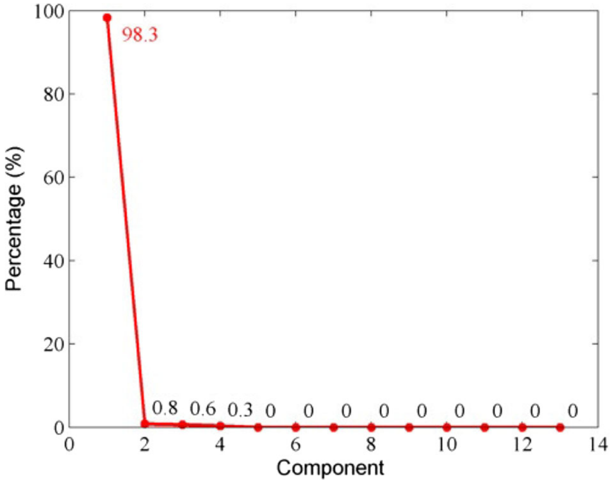


**Fig. 8** The variance carried by each principal component

carried by each component. The percentage of variance reflects the information contained in each component. As shown in Fig. 8, the variance of the first component reaches 98.3%. Moreover, the first three components provide almost all of the total information. The dimension of the reduced input feature $F(t)$ is much lower than the original feature $F_S(t)$. It is a much convenient tool in the procedure of slow feature function learning.

After discussing the basic theory of SFA algorithm, we need to determine the strategy of slow feature function learning for the fall event recognition, i.e., defining the types of input-output function that represent the fall action appropriately after transforming the input features. According to [46], the supervised SFA learning strategy is adopted to extract slow feature functions for each action category independently, mainly focusing on the fall actions. After nonlinear expansion of SFA, we get a nine-dimensional output feature. Figure 9 illustrates an output of nine slow feature functions, while the original input feature is shown in Fig. 6. We can see that the slowest signal is in the first output, while the second output is less optimal one. Compared with the other outputs, there is only a little vibration in the slowest signal, just like a steady signal. According to the theory of SFA, these outputs, from the first to the ninth, carry different types of information of the input signal. Statistical features can be derived from all of the output slow features. However, different input features may

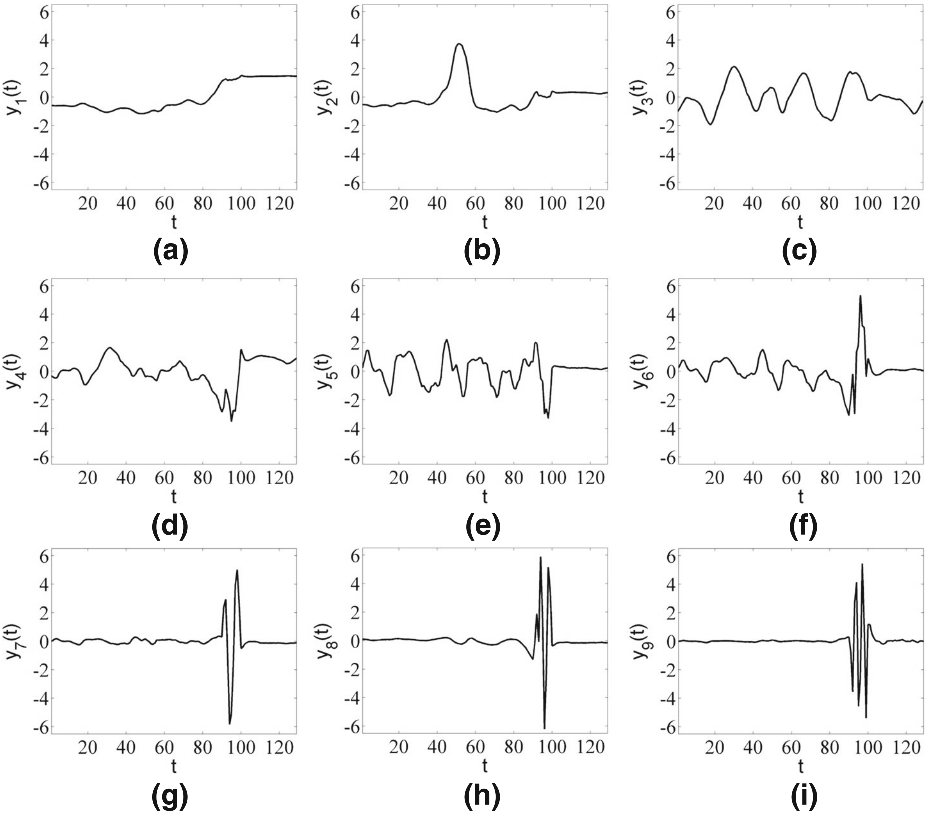


**Fig. 9** The nine learned slow feature functions. **a** The first output. **b** The second output. **c** The third output. **d** The fourth output. **e** The fifth output. **f** The sixth output. **g** The seventh output. **h** The eighth output. **i** The ninth output

share some similar patterns after the transformation of slow feature functions, so different categories of these patterns lead to misclassification.

### *3.4.2 Fall accident representation*

In the literature on action recognition, $d$ successive frames are called an action snippet. The input feature of SFA learning is obtained exactly from $d$ successive frames. Thus, we get a statistical feature from a fall action snippet to represent a fall action sequence. In one action snippet, the accumulated squared derivative feature is computed as follows. Initially, the training samples are preprocessed before the representation. After the reformation of the procedure, each $h \times w \times d$ sample is represented as a vector sequence with the time length of $d - \Delta t + 1$. The vector at each moment of time is obtained by concatenating the features from $\Delta t$ successive frames. With the learned slow feature functions, each input sequence is transformed into a new vector sequence with the size of $K \times (d - \Delta t + 1)$, wherein $K$ ($K = 9$, in this paper) is the number of slow functions.

The objective function of SFA minimizes the average squared derivative, so the fitting degree of a sample to a certain slow feature function can be measured by the squared derivative of the transformed sample. If the value is small, the sample fits the slow feature function very well. For a sample $T$ and slow function $F_j$, the squared derivative $v_j$ is

$$v_j = \frac{1}{d - \Delta t} \sum_{t=1}^{d-\Delta t} \left[ T(t+1) \otimes F_j - T(t) \otimes F_j \right]^2 \tag{24}$$

where $\otimes$ is the transformation operation. We accumulated the squared derivatives to form the feature $F_{asd} = V =< v_1, v_2, \ldots, v_k >$. For the effectiveness of different samples, it is necessary to normalize the accumulated squared derivatives feature vector. Here, we adopt the $L_2$ normalization, as follows

$$\hat{F}_{asd} = V / \|V\|_2 \tag{25}$$

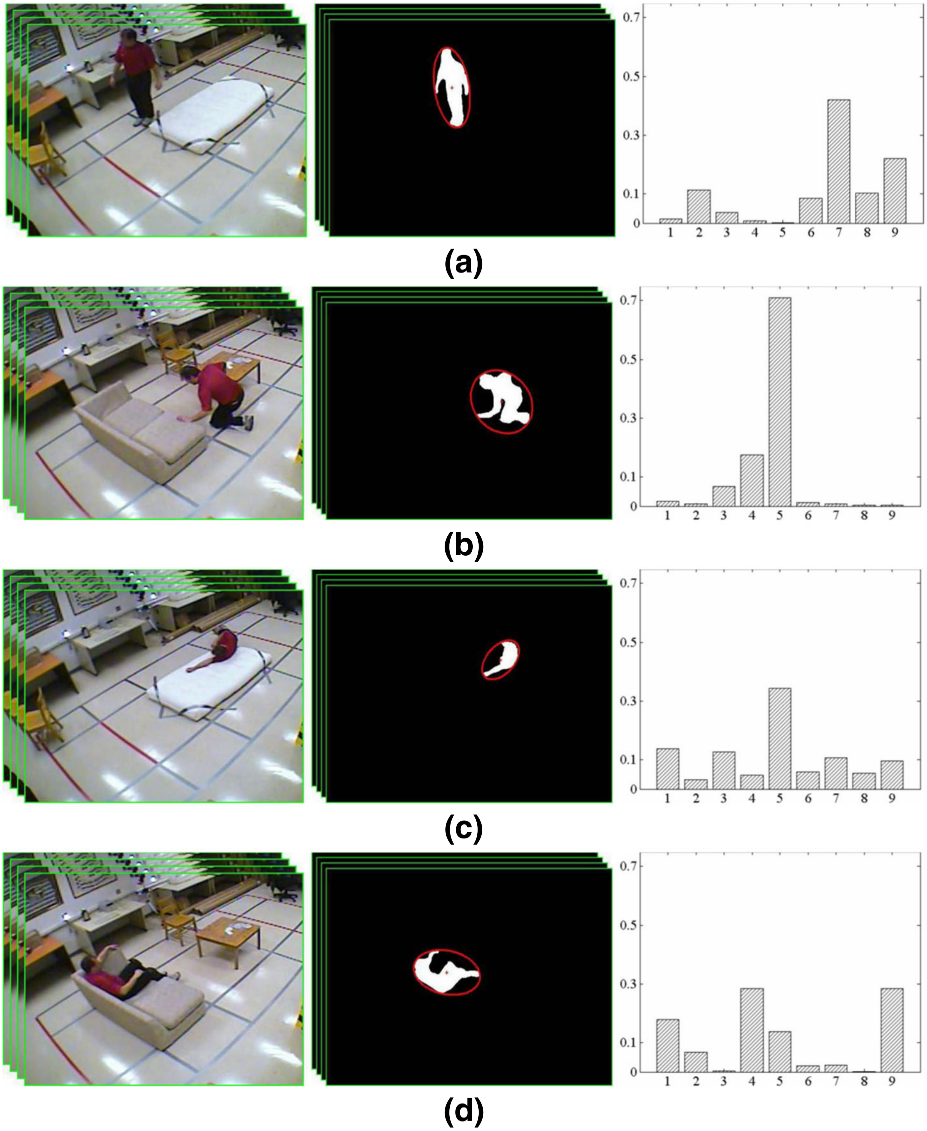
Figure 10 shows examples of accumulated squared derivative feature representation from fall dataset [4]. Each picture of the first column stands for an action snippet. The input features are extracted from these snippets, as is shown in the second column. For each category, the learned nine slow feature functions are used to computer accumulated squared derivatives features as shown in the third column of Fig. 10. We can see that the accumulated values of those slow feature functions corresponding to the four actions are quite different. This is a discriminative information for classification.

### *3.4.3 Fall accident detection*

Once we have obtained the accumulated squared derivatives feature $\hat{F}_{asd}$, actions can be classified into one of the four categories (walking, crouching, falling and sitting) just as shown in Fig. 10. Consider the training set with $N$ samples, defined by $\{u_i, l_i\}$, $i = 1, \ldots, N$, with the data $u_i \in R^n$ and label $l_i \in \{-1, +1\}$. SVM is a binary classifier that constructs an optimal hyper plane according to the minimum structure risk principle [37] in the feature space or transformed high dimensional feature space. It can be used for classification, regression or other tasks. The hyper plane is defined as :

$$f(u) = \omega \cdot \phi(u) + b \tag{26}$$

where $\omega$ indicates a set of weights, $\phi(u)$ is a nonlinear mapping operation on $u$, $b$ is the bias. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to



(a)

(b)

(c)

(d)

**Fig. 10** Examples of feature representation. **a** Walk. **b** Crouch. **c** Fall. **d** Sit

the nearest training data point of any class (so-called functional margin), since in general the larger the margin, the lower the generalization error of the classifier. It is formulated as follows

$$\min \quad \psi(\omega) = \frac{1}{2}\omega^2 + \frac{C}{2}\sum_{i=1}^{N}\xi_i^2 \tag{27}$$

where $C$ is a regularization factor and $\xi_i$ is the error of the $i$th sample. Since it was originally designed for binary classification, the issue of how to extend it for multi-class problems is still an open problem. At present, there are two types of strategies for developing a multi-class SVM. One way is constructing several binary SVM and combining the classified result by some rules such as one against all, one against one. The other is directly considering all the multi-class data in one optimization formulation. Though this method solves a multi-class SVM problem in only one step, much optimization is required. Moreover, parameters of the SVM are very complicated.

In this paper, the directed acyclic graph scheme is employed to combine several SVM for solving the multi-class classification problem. Figure 11 presents the structure of the directed acyclic graph support vector machines (DAGSVM) for the four actions classification. It looks like a tree-structure and each node in this tree-structure corresponds to a simple binary SVM. The decision process just follows the structure based on a sequence of two-class operation, proceeding from the root node to the leaf node. A final decision is made when a bottom node is reached. The adapted DAGSVM scheme has been proved to have a theoretically defined generalization error bound and to be more efficient than other multi-class SVM schemes on the training and computing time [31].

### 3.4.4 Computational complexity

In our proposed method based on slow feature analysis, most of the computation costs mainly focus on the two large covariance matrices $A$ and $B$ for solving the generalized eigenvalue problem from (21), i.e., $AW = BW\Lambda$. The expanded nonlinear function space $H$ on which SFA is performed is chosen here to be the set of all monomials of degree one and two including mixed terms, as discussed at length in Section 3.3. A run with SFA requires the computation of two large covariance matrices, the elements of which are in the order of $O(K^2)$, where $K$ is the dimension of the expanded function space. In the quadratic expansion function space $K$ is determined by $K = I + I(I+1)/2$, where $I$ is the dimension of the input signal. In the case of polynomials of degree two, this corresponds to

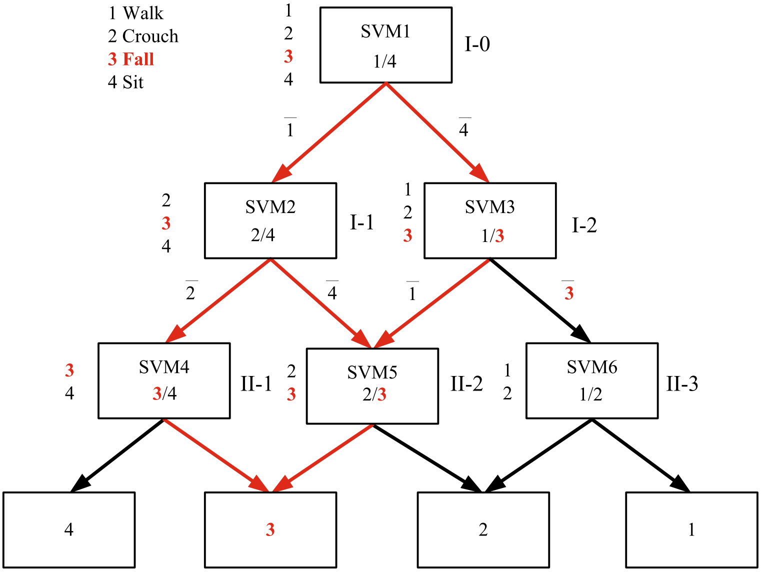


**Fig. 11** Classification structure of DAGSVM

some elements in the order of $O(I^4)$. Obviously, this is computationally expensive. Considering the real-time efficiency, we conduct a standard preprocessing step using principal component analysis (PCA) to reduce the dimensionality of the input signals $I$ for slow feature analysis. In our experiment, the dimensionality of the input features $I$ is reduced from $1\times6\times3$ =18 to $I$=3, capturing 98.30% of the total variance as shown in Fig. 8.

# 4 Experiments

In this section, we show the performance of our fall detection system. All the experiments are carried out on a desktop with Intel (R) Core (TM) i7-6700 CPU and 4.00 GB RAM. We evaluate the proposed method on two public fall detection datasets. Some detailed information about these two datasets is given below.

## 4.1 Datasets

The dataset-I is from multiple-camera fall dataset [4]. This video dataset contains simulated falls and normal daily activities recorded in 24 realistic situations. In each scenario, an actor plays many activities such as falling, crouching, sitting on a sofa, walking, and so on. Every scene is shot simultaneously with eight different cameras mounted around the room where the fall incident happens. All of the actions are performed by the same person with different color garments. In our tests, videos from various camera views are mixed and treated equally. Figure 12 illustrates some key video frames from this dataset. Detailed information of dataset-I is given in Table 1.

The dataset-II is the SDUFall dataset [24]. This dataset comprises various types of samples including RGB videos, depth videos, and 20 skeleton joint positions. There are 20 young men and women in the shooting of the data collection. Each man and woman

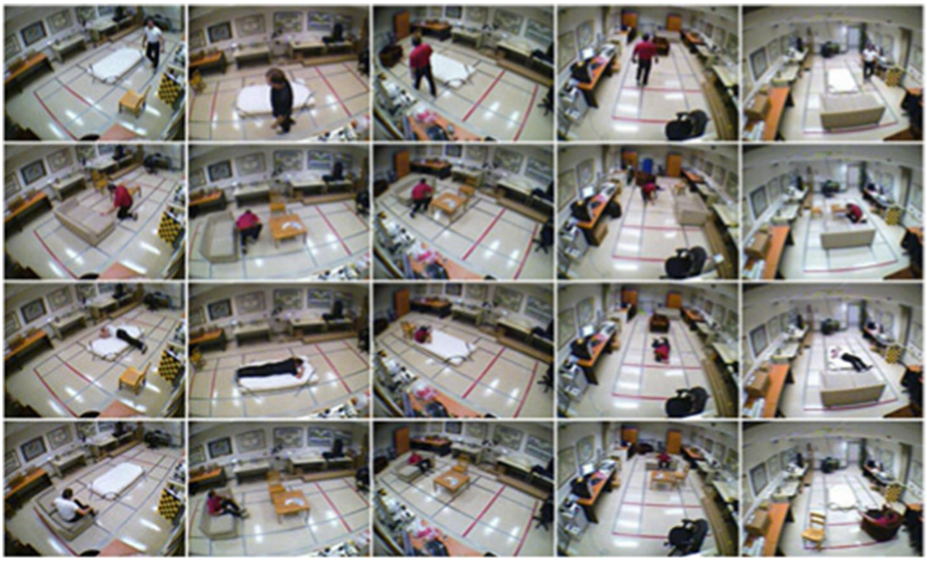


**Fig. 12** Four categories of dataset-I. From top to bottom: walk, crouch, fall, and sit

**Table 1** Different activities depicted in dataset-I

| Activity | Count | Label |
|---|---|---|
| Walk | 192 | 1 |
| Crouch | 80 | 2 |
| Fall | 208 | 3 |
| Sit | 80 | 4 |
| Total | 560 | – |

performs the actions 10 times, including falling, bending, squatting, sitting, lying and walking. Since it is hard to capture real falls, the subjects fall intentionally. It is worth noting that there is a significant proportion of confusing activities, such as falling and lying. For capturing the actions in different conditions, the videos are recorded with the following scenarios: carrying or not carrying an object, changing room layout, changing direction and position relative to the camera. In our experiments, we use only the RGB videos. A total of 1200 RGB videos are collected. Some key video frames from dataset-II are shown in Fig. 13. Detailed information about dataset-II is given in Table 2.

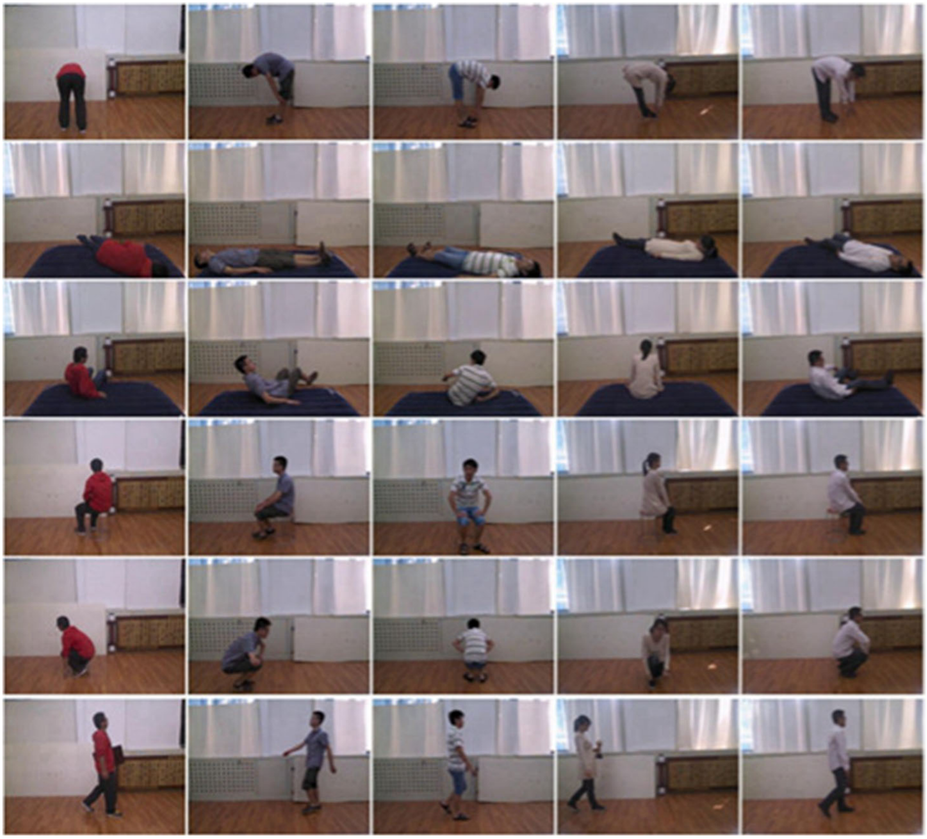


**Fig. 13** Six categories of dataset-II. From top to bottom: bend, fall, lie, sit, squat, and walk

**Table 2** Different activities depicted in dataset-II

| Activity | Count | Label |
|---|---|---|
| Bend | 200 | 1 |
| Fall | 200 | 2 |
| Lie | 200 | 3 |
| Sit | 200 | 4 |
| Squat | 200 | 5 |
| Walk | 200 | 6 |
| Total | 1200 | – |

## 4.2 Experimental setup

Generally, we should choose samples to train the classifier. Therefore, the datasets are split into two parts. The first part contains eighty percent of the samples (448 from dataset-I and 960 from dataset-II ). These are used for training, and the others for testing (112 from dataset-I and 240 from dataset-II). The commonly used cross-validation technique (PRtools – a popular software package for pattern recognition [11]) is applied to tune the parameters of the classification system.

Two different types of evaluations are made. In the first assessment, we perform fall versus non-fall classification on the two datasets. The second evaluation is a multi-class actions classification. Four actions are classified on dataset-I, and six-class classification experiments are carried out on dataset-II. The detailed information about action categories and labels are given in Tables 1 and 2, respectively. In the two types of experiments, the obtained results are compared with other state-of-the-art methods on the two datasets.

## 4.3 Experimental results on dataset-I

In the fall versus non-fall experiments, falls are the positive samples, and the other actions are the negative samples. The classification results are listed in Table 3. As the results show, our proposed SFA-SVM approach yields the best accuracy. The accuracy reaches 94.00%. Furthermore, our method is respectively 2.00% and about 4.00% more accurate than the compared silhouette [30] and shape variation methods [9]. Such good performance is the result of using slow feature analysis, the advantages of which in this respect become very clear. The classification accuracy is greatly improved, about 10.00% higher than that of the bounding box method [23]. To further study the performance of our method, we visualize the receiver operating characteristic (ROC) curve of our method on dataset-I in Fig. 14. ROC curve is a graphical plot that illustrates the performance of a binary classifier system as its discrimination threshold is varied. It is created by plotting the true positive rate (TPR) against the false positive rate (FPR). TPR is also known as sensitivity, recall or probability of

**Table 3** Fall versus non-fall classification results on dataset-I

| Method | Accuracy |
|---|---|
| Bounding box [23] | 84.44% |
| Shape variation [9] | 90.05% |
| Silhouette [30] | 92.00% |
| Proposed method | 94.00% |

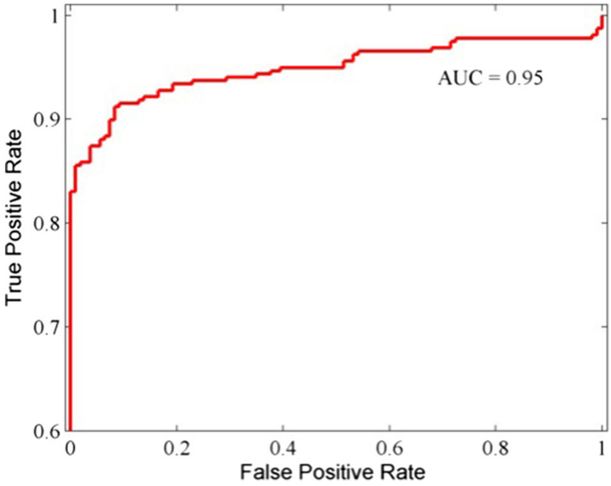


**Fig. 14** The receiver operating characteristic curve of our method on dataset-I

detection in machine learning. FPR is also known as the fall-out or probability of false alarm and can be calculated as 1-specificity. Practically, the area under the ROC curve (AUC) is often adopted as the criterion for evaluation. The AUC value is equivalent to the probability that a randomly chosen positive example is ranked higher than a randomly chosen negative example. It is always between 0 and 1.0. The larger AUC means the better performance of the approach. Figure 14 shows the area under the ROC curve. The AUC of our method is 0.95 which is approximately the ideal value.

The results of the second type of experiments are also obtained. The overall accuracy of multi-class classification is given in Table 4. The results indicate that the DAGSVM is better than the ELM in multi-class action recognition. The accuracy of our DAGSVM classifier is about 2.00% and 3.25% better than that of ELM and SVM, respectively. Our method can reach 98.25% accuracy on four classes action classification. Still, it is not a reliable metric to estimate the performance of the classifiers only using classification accuracy for the four actions classification since classification accuracy is mainly concerned with the percentage of rightly classified instances in the total numbers of samples and neglects the rate of misclassified samples. Classification rate can not accurately reflect the performance of the multi-class classifier because in the case of one class accuracy is very high while in the case of another class is very low.

For adequately evaluating the effectiveness of classifiers, the confusion matrix, also known as an error matrix, is the best choice. A confusion matrix is a specific table that allows visualizing the performance of a classifier on a set of test data for which true values are already known. Each column of the matrix represents the instances (actions) in a

**Table 4** Overall accuracy rates for four actions classification on dataset-I

| Method | Accuracy |
|---|---|
| SVM | 95.00% |
| ELM | 96.50% |
| Our method | 98.25% |

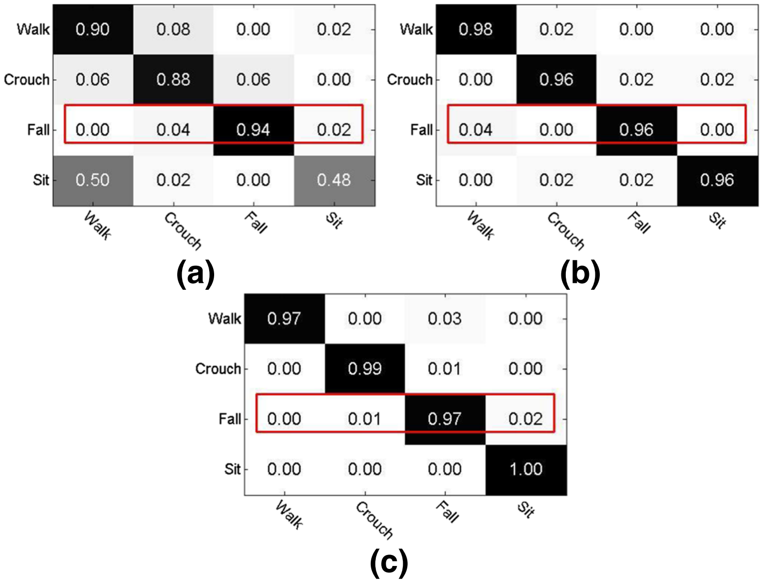
predicted class while each row represents the instances in an actual class (or vice versa). All correct predicted instances are located in the diagonal of the table, so it is easy to inspect the table visually and to find out the errors, which are represented by the values outside the diagonal. It is easy to see if the classifier confuses two classes i.e., perceives the sample of one class as belonging to the other.

The results of the SVM, ELM, and DAGSVM for the second type of experiments are given in Fig. 15. One can easily see from Fig. 15a that SVM is more likely to misclassify other activities as falls. Crouching is more likely to be misclassified as falling with 6.00% misclassification rates. The worst classification accuracy (48.00%) is obtained for the sitting, about 50.00% of sitting are misclassified as walking. The highest classification accuracy (94.00%) is obtained for falling when using the SVM classifier.

The results, which are shown in Fig. 15b, indicate that ELM has smaller misclassification error than the SVM classifier. The best correct classification rate (98.00%) is obtained for walking, and the accuracies of other three actions are all 96.00%. But the movement that is most confounding to walking is still falling since 4.00% of falling are misclassified as walking. Moreover, there are about 2.00% of crouching, and 2.00% of sitting that are still misclassified as falling when using the ELM classifier.

The DAGSVM approach gets better classification performance than the ELM and SVM. The classification accuracy for all actions is improved as is clear from Fig. 15c. The classification rate is satisfying for every action. The highest classification accuracy is recorded for sitting. Walking and sitting are the most confounding activities for the DAGSVM method. About 3.00% of walking are misclassified as falling and 6.00% of sitting are also misclassified as walking.

Generally speaking, DAGSVM has smaller misclassification errors as compared to other two methods. For all the three approaches, the actions that are most frequently misclassified as falling are crouching and sitting. This reflects the similarity of the activities. The characteristics of crouching are similar to the falling on dataset-I.



| | Walk | Crouch | Fall | Sit |
|---|---|---|---|---|
| Walk | 0.90 | 0.08 | 0.00 | 0.02 |
| Crouch | 0.06 | 0.88 | 0.06 | 0.00 |
| Fall | 0.00 | 0.04 | 0.94 | 0.02 |
| Sit | 0.50 | 0.02 | 0.00 | 0.48 |

(a)

| | Walk | Crouch | Fall | Sit |
|---|---|---|---|---|
| Walk | 0.98 | 0.02 | 0.00 | 0.00 |
| Crouch | 0.00 | 0.96 | 0.02 | 0.02 |
| Fall | 0.04 | 0.00 | 0.96 | 0.00 |
| Sit | 0.00 | 0.02 | 0.02 | 0.96 |

(b)

| | Walk | Crouch | Fall | Sit |
|---|---|---|---|---|
| Walk | 0.97 | 0.00 | 0.03 | 0.00 |
| Crouch | 0.00 | 0.99 | 0.01 | 0.00 |
| Fall | 0.00 | 0.01 | 0.97 | 0.02 |
| Sit | 0.00 | 0.00 | 0.00 | 1.00 |

(c)

**Fig. 15** Confusion matrices of the three methods on dataset-I. **a** SVM. **b** ELM. **c** Our method

**Table 5** Fall versus non-fall classification results on dataset-II

| Method | Accuracy |
|---|---|
| BoW-SVM [24] | 63.12% |
| BoW-ELM [24] | 84.36% |
| BoW-VPSO-ELM [24] | 86.83% |
| FV-SVM [3] | 88.83% |
| Proposed method | 96.57% |

## 4.4 Experimental results on dataset-II

Here we carry out two types of experiments, exactly as we did in the previous section. Firstly, the dataset-II is used to verify the performance of our method on fall and non-fall actions. Secondly, six actions of this dataset are tested to confirm the application of the method in multi-class classification.

In the first type of the experiments, the accuracy rates, including the compared results in [24] and [3], are listed in Table 5. As shown in the table, our method outperforms other methods regarding the accuracy. It reaches about 96.57%. The accuracy is 7.74% and 9.74% higher than that of FV-SVM [3] and BoW-VPSO-ELM [24], respectively. The critical point is that after introducing slow feature analysis into the proposed method, the SVM classification accuracy has been highly improved. It is about 10.00% and 30.00% higher than that of FV [3] and BoW [24], respectively. The ROC curve and AUC value of our method on dataset-II are also shown in Fig. 16. The area under the ROC reaches 0.99. There is only a little gap between ideal AUC value and the AUC of our method. This sufficiently demonstrates the effectiveness of the proposed method on dataset-II.

In the second type of experiments, the overall recognition rate is calculated. The results are shown in Table 6, from which we can observe that our method outperforms the ELM and SVM classifiers by 10.00% and 20.00%, respectively. The confusion matrices are shown in Fig. 17. It is clear from Fig. 17a, that SVM easily misclassifies activities, especially

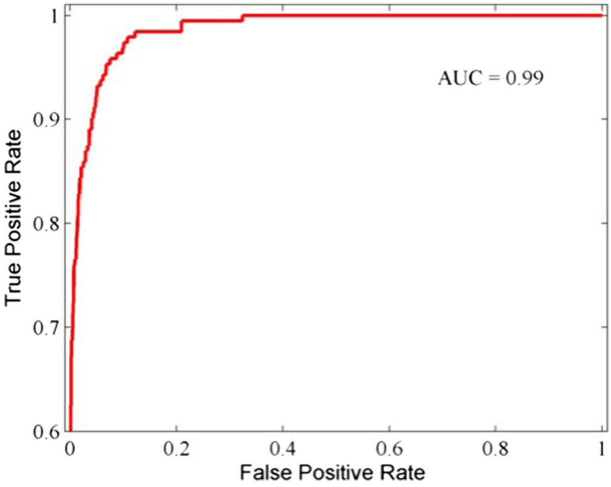


**Fig. 16** The receiver operating characteristic curve of the our method on dataset-II

**Table 6** Overall accuracy rates for six actions classification on dataset-II

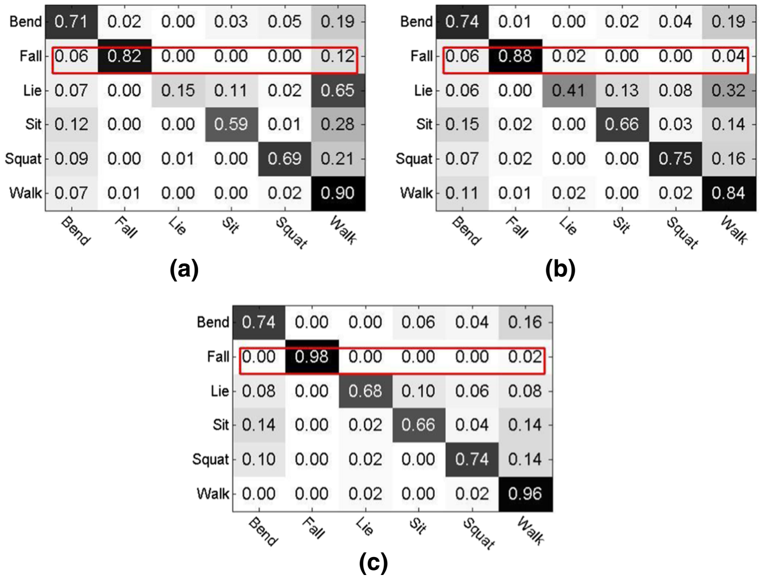
| Method | Accuracy |
|---|---|
| SVM | 64.30% |
| ELM | 71.34% |
| Our method | **81.33%** |

lying, sitting and squatting. The motions of the three actions look like the same in the first stage. Lying actions are the most difficult to classify. The lowest classification accuracy rate (15.00%) is obtained for lying. About 65.00% of lying is misclassified as walking. Sitting and squatting are also easy to be misclassified as falling with 28.00% and 21.00%, respectively. The accuracy for falling is only 82.00%, while the highest classification accuracy rate exceeds 90.00%.

The ELM method gets better classification results than the SVM classifier. From Fig. 17b, we can observe that the classification accuracy for lying is improved. However, there are still 32.00% and 13.00% of lying misclassified as walking and sitting, respectively. The worst classification accuracy is still obtained for lying, and the highest correct classification rate (88.00%) is recoded for falling. Lying and sitting are the most confounding activities for the ELM approach. 32.00% of lying are misclassified as walking and 15.00% of sitting are also misclassified as bending.

Finally, Fig. 17c shows the achievements of our proposed DAGSVM. Except squatting, all human actions are classified more accurate than the compared two methods. The classification accuracy of lying is much better. Falling gets the highest classification accuracy (98.00%) and the worst classification result (66.00%) is obtained for sitting. 14.00% of sitting are misclassified as bending and 10.00% of lying are still misclassified as a sitting.

To sum up the experimental results, with the help of slow feature analysis, the DAGSVM is proved to be effective on dataset-II, showing much better performance. The overall



| | Bend | Fall | Lie | Sit | Squat | Walk |
|---|---|---|---|---|---|---|
| Bend | 0.71 | 0.02 | 0.00 | 0.03 | 0.05 | 0.19 |
| Fall | 0.06 | 0.82 | 0.00 | 0.00 | 0.00 | 0.12 |
| Lie | 0.07 | 0.00 | 0.15 | 0.11 | 0.02 | 0.65 |
| Sit | 0.12 | 0.00 | 0.00 | 0.59 | 0.01 | 0.28 |
| Squat | 0.09 | 0.00 | 0.01 | 0.00 | 0.69 | 0.21 |
| Walk | 0.07 | 0.01 | 0.00 | 0.00 | 0.02 | 0.90 |

**(a)**

| | Bend | Fall | Lie | Sit | Squat | Walk |
|---|---|---|---|---|---|---|
| Bend | 0.74 | 0.01 | 0.00 | 0.02 | 0.04 | 0.19 |
| Fall | 0.06 | 0.88 | 0.02 | 0.00 | 0.00 | 0.04 |
| Lie | 0.06 | 0.00 | 0.41 | 0.13 | 0.08 | 0.32 |
| Sit | 0.15 | 0.02 | 0.00 | 0.66 | 0.03 | 0.14 |
| Squat | 0.07 | 0.02 | 0.00 | 0.00 | 0.75 | 0.16 |
| Walk | 0.11 | 0.01 | 0.02 | 0.00 | 0.02 | 0.84 |

**(b)**

| | Bend | Fall | Lie | Sit | Squat | Walk |
|---|---|---|---|---|---|---|
| Bend | 0.74 | 0.00 | 0.00 | 0.06 | 0.04 | 0.16 |
| Fall | 0.00 | 0.98 | 0.00 | 0.00 | 0.00 | 0.02 |
| Lie | 0.08 | 0.00 | 0.68 | 0.10 | 0.06 | 0.08 |
| Sit | 0.14 | 0.00 | 0.02 | 0.66 | 0.04 | 0.14 |
| Squat | 0.10 | 0.00 | 0.02 | 0.00 | 0.74 | 0.14 |
| Walk | 0.00 | 0.00 | 0.02 | 0.00 | 0.02 | 0.96 |

**(c)**

**Fig. 17** Confusion matrices of the three methods on dataset II. **a** SVM. **b** ELM. **c** Our method

accuracy reaches 81.33% for six actions recognition. It is about 10.00% higher than that of the FV-SVM method [3]. As for the most confounding activity (lying), our approach achieves the best performance with a much better classification rate (68.00%) than that of ELM (41.00%) and SVM (15.00%), respectively.

## 5 Conclusions

In this paper, we proposed a novel slow feature analysis based framework for fall detection with one-dimensional feature vector sequences. Firstly, a color distortion background subtraction model was adopted to extract the human silhouettes. The morphological operations were applied to refine the extraction results and to remove noise and non-human blobs. The region of interest was covered by an ellipse as a human shape model. Six shape deformation features were quantified from the covered silhouette, which was used for representing different human postures at a given moment of time. Secondly, the basic slow feature analysis algorithm with a supervised learning strategy was developed to analyze the shape feature sequences. The learned slow feature functions could encode discriminative information of the fall incident. Finally, the transformed slow features of the fall sequences were aggregated into a classification vector by the accumulated squared derivative scheme. The falls were classified after the accumulated squared derivative features had been fed into the DAGSVM. Our experiments also verified that, compared with other state-of-the-art methods, the proposed method could achieve a satisfying accuracy on the two public datasets.

Nevertheless, the performance of our method still largely depends on the accuracy of the background subtraction technique, even though we utilize only six features extracted from a silhouette. Besides, the two datasets used here just provide simulated falls and other few daily activities, which constitute only a small part of real world interactions. Hence, our further work is to investigate the applicability of our method to the detection of real falls along with complex daily activities.

**Acknowledgements** This work was supported by Department of Science and Technology in Hebei Province China with Grant No.12213519D1. The authors also would like to thank the anonymous editors and reviewers for their insightful comments and suggestions which improved this work.

## References

1. Ageing WHO (2008) Who global report on falls prevention in old age. Technical Report, World Health Organization
2. Amin MG, Zang YD, Ahmad F et al (2016) Radar signal process for elderly fall detection: the future for in-home monitoring. IEEE Signal Proc Mag 33(2):71–80
3. Aslan M, Sengur A, Xiao Y et al (2015) Shape feature encoding via fisher vector for efficient fall detection in depth-videos. Appl Soft Comput 37:1023–1028
4. Auvinet E, Rougier C, Meunier J et al (2010) Multiple cameras fall dataset. Technical Report, DIRO-Université de Montréal
5. Barnich O, Van D (2011) Vibe: a universal background subtraction algorithm for video sequences. IEEE T Image Process 20(6):1709–1724
6. Berkes P, Wiskott L (2005) Slow feature analysis yields a rich repertoire of complex cell properties. J Vision 5(6):579–602
7. Bosch-Jorge M, Sanchez-Salmeron AJ, Valera A (2014) Fall detection based on the gravity vector using a wide-angle camera. Expert Syst Appl 41(17):7980–7986
8. Chaudhuri S, Thompson H, Demiris G (2014) Fall detection devices and their use with older adults: a systematic review. J Geriatr Phys Ther 37(4):178–196

9. Chua J, Chang Y, Lim W (2013) A simple vision-based fall detection technique for indoor video surveillance. Signal Image Video P 9(3):623–633
10. Crispim-Junior C, Buso V, Avgerinakis K et al (2016) Semantic event fusion of different visual modality concepts for activity recognition. IEEE Trans Patt Anal Mach Intell 38(8):1598–1611
11. Duda R, Hart P, Stork D (2000) Pattern classification, 2nd edn. Wiley, New Jersey
12. Erden F, Velipasalar S, Alkar AZ et al (2016) Sensors in assisted living: a survey of signal and image processing methods. IEEE Signal Proc Mag 33(2):36–44
13. Hamm J, Money A, Atwal A et al (2016) Fall prevention intervention technologies: a conceptual framework and survey of the state of the art. J Biomed Inform 59:319–335
14. Hassan MM, Lin K, Yue X et al (2017) A multimedia healthcare data sharing approach through cloud-based body area network. Future Gener Comp Sy 66:48–58
15. Heikkil M, Pietikinen M (2006) A texture-based method for modeling the background and detecting moving objects. IEEE Trans Patt Anal Mach Intell 28(4):657–662
16. Horprasert T, Harwood D, Davis L (1999) A statistical approach for real-time robust background subtraction and shadow detection. In: Proceedings of international conference on computer vision, pp 1-19
17. Hossain MS, Hossain SA, Alamri A et al (2013) Ant-based service selection framework for a smart home monitoring environment. Multimed Tools Appl 67(2):433–453
18. Igual R, Medrano C, Plaza I (2013) Challenges issues and trends in fall detection systems. Biomed Eng Online 12(66):1–66
19. Islam SMR, Kwak D, Kabir MDH et al (2015) The internet of things for health care: a comprehensive survey. IEEE Access 3:678–708
20. Khan MS, Yu M, Feng P et al (2015) An unsupervised acoustic fall detection system using source separation for sound interference suppression. Signal Process 110(61):199–210
21. Khan S, Hoey J (2017) Review of fall detection techniques: a data availability perspective. Med Eng Phys 39:12–22
22. Koshmak G, Loutfi A, Linden M (2015) Challenges and issues in multisensor fusion approach for fall detection: review paper. J Sensors 2016:1–16
23. Liu CL, Lee CH, Lin PM (2010) A fall detection system using $k$-nearest neighbor classifier. Expert Syst Appl 37(10):7174–7178
24. Ma X, Wang H, Xue B (2014) Depth-based human fall detection via shape features and improved extreme learning machine. IEEE J Biomed Health 18(6):1915–1922
25. Madarshahian R, Caicedo J, Zambrana DA (2016) Benchmark problem for human activity identification using floor vibrations. Expert Syst Appl 62:263–272
26. Meng L, Miao C, Leung C (2017) Towards online and personalized daily activity recognition, habit modeling, and anomaly detection for the solitary elderly through unobtrusive sensing. Multimed Tools Appl 76(8):10779–10799
27. Mirmahboub B, Samavi S, Karimi N et al (2013) Automatic monocular system for human fall detection based on variations in silhouette area. IEEE T Biomed Eng 60(2):427–436
28. Mubashir M, Shao L, Seed L (2013) A survey on fall detection: principles and approaches. Neurocomputing 100(16):144–152
29. Noury N, Fleury A, Rumeau P et al (2007) Fall detection-principles and methods. In: Proceedings of 29th annual international conference of the engineering in medicine and biology society, pp 1663–1666
30. Olivieri DN, Conde IG, Sobrino XA (2012) Eigenspace-based fall detection and activity recognition from motion templates and machine learning. Expert Syst Appl 39(5):5935–5945
31. Platt JC, Cristianini N, Shawe-Taylor J (1999) Large margin dags for multiclass classification. In: Proceedings of Conference on Neural Information Processing Systems, pp 547–553
32. Poppe R (2010) A survey on vision-based human action recognition. Image Vision Comput 28(6):976–990
33. Pratt WK, Adams JE (2007) Digital image processing, 4th edn. Prentice Hall, New Jersey
34. Salem O, Guerassimov A, Mehaoua A et al (2013) Anomaly detection scheme for medical wireless sensor networks. Springer, New York
35. Shin I, Son J, Ahn S et al (2015) A novel short-time fourier transform-based fall detection algorithm using 3-axis accelerations. Math Probl Eng 2015(2015):1–8
36. Su S, Wu SS, Chen SY (2016) Multi-view fall detection based on spatio-temporal interest points. Multimed Tools Appl 75(14):8469–8492
37. Vapnik V (2013) The nature of statistical learning theory. Springer Science and Business Media, Berlin
38. Wang S, Chen L, Zhou Z et al (2016) Human fall detection in surveillance video based on pacnet. Multimed Tools Appl 75(19):11603–11613

39. Weinland D, Ronfard R, Boyer E (2011) A survey of vision-based methods for action representation segmentation and recognition. Comput Vis Image Und 115(2):224–241
40. Wickramasinghe A, Torres RLS, Ranasinghe DC (2017) Recognition of falls using dense sensing in an ambient assisted living environment. Pervasive Mob Comput 34:14–24
41. Wiskott L, Sejnowski TJ (2002) Slow feature analysis: unsupervised learning of invariances. Neural Comput 14(4):715–770
42. Yoon HJ, Ra HK, Park T et al (2015) Fades: behavioral detection of falls using body shapes from 3D joint data. J Amb Intel Smart En 7(6):861–877
43. Yu M, Rhuma A, Naqvi S et al (2012) A posture recognition based fall detection system for monitoring an elderly person in a smart home environment. IEEE T Inf Technol B 16(6):1274–1286
44. Yun Y, Gu YH (2016) Human fall detection in videos by fusing statistical features of shape and motion dynamics on riemannian manifolds. Neurocomputing 207:726–734
45. Zhang Z, Conly C, Athitsos V (2015) A survey on vision-based fall detection. In: Proceedings of 8th ACM international conferences on pervasive technologies related to assistive environments, pp 1-7
46. Zhang Z, Tao D (2012) Slow feature analysis for human action recognition. IEEE Trans Patt Anal Mach Intell 34(3):436–450

**Kaibo Fan** was born in China in 1985.He obtained his B.E. degree in Automation Engineering from Yantai Nanshan University in 2009,M.S. degree in Control Theory and Control Engineering from Tianjin University of Technology in 2012,China. He is currently a Ph.D. student in School of Electrical and Information Engineering at Tianjin University, China. His research interests are computer vision and image processing.

**Ping Wang** is a professor at Department of Electrical Engineering and Automation, Tianjin University, China. She is a PhD supervisor in control science and engineering. Her research interests include pattern recognition and its application, image understanding and moving objects tracking.

**Shuo Zhuang** is now a Ph.D. Candidate at School of Electrical and Information Engineering, Tianjin University, China. His current research interests include machine learning, computer vision and image processing, as well as various computer vision applications in agriculture.