CrossMark

Human fall detection using normalized shape aspect ratio

Weidong Min[1,2] · Song Zou[1] · Jing Li[1,3]

Abstract
In video surveillance, automatic human fall detection is important to protect vulnerable groups such as the elderly. When the camera layout varies, the shape aspect ratio (SAR) of a human body may change substantially. In order to rectify these changes, in this paper, we propose an automatic human fall detection method using the normalized shape aspect ratio (NSAR). A calibration process and bicubic interpolation are implemented to generate the NSAR table for each camera. Compared with some representative fall detection methods using the SAR, the proposed method integrates the NSAR with the moving speed and direction information to robustly detect human fall, as well as being able to detect falls toward eight different directions for multiple humans. Moreover, while most of the existing fall detection methods were designed only for indoor environment, experimental results demonstrate that this newly proposed method can effectively detect human fall in both indoor and outdoor environments.

Keywords Human fall detection · Falling toward different directions · Indoor and outdoor environments · Normalized shape aspect ratio

1 Introduction

Real-time detection of unpredictable events and abnormal human behavior can prevent further damage from happening. As an example of abnormal behavior detection, automatic human fall detection in video surveillance plays an important role in protecting vulnerable groups from falling, Especially, with the rapid growth of the aging society, fall injuries have become the second leading cause of accidental death, and studies have shown that medical outcomes are largely dependent on response and rescue time [2, 14].

✉ Jing Li
   jingli@ncu.edu.cn

1   School of Information Engineering, Nanchang University, Nanchang 330031, China

2   School of Software Engineering, Nanchang University, Nanchang 330029, China

3   State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

A few of fall detection methods have been developed in recent years, some of which were summarized in a survey together with principles and approaches [23]. They can be categorized into the following three major types: i) wearable sensor-based; ii) ambience sensing-based; and iii) vision-based. The methods based on wearable sensors equipped on the human body acquire acceleration and position information of human motion in order to find out whether it is in accord with the features of human fall [3, 9, 16, 18]. These sensors include speed sensors, accelerometers, position sensors and three-axis accelerometers [3, 9, 21], which were also adopted in smart phones for fall detection [21]. However, this kind of methods are poor in terms of sensitivity, and thus are prone to give false positives. Moreover, they require each user to wear sensors that may be uncomfortable, inconvenient and uneconomical. In this way, it is unpractical to detect humans wearing no sensors in public places. In addition, it is difficult to judge holistic movement of a human only by a limited number of sensors for collecting the movement information, which leads to some false positives in falls. As an initial attempt, data from multiple sensors were used to reconstruct a 3D human body to detect falls [16]. However, the method could not detect falls toward different directions.

Ambience sensing-based methods arrange a number of sensors in the scenes to determine whether a human is going to fall by detecting the floor vibration, the electric current generated near the fall site, the fall sound, and so on [26]. In [30], multiple sensors were deployed in the home environment, while accelerometer offered an inexpensive way to detect falls. After the user's approximate position was calculated by Received Signal Strength Indicator, image processing was used to determine the user posture when alerts happened. In [37], a Gaussian mixture model (GMM) supervector was created to model each fall as a noise segment by using the audio signal from a single far-field microphone. A support vector machine built on a kernel between GMM supervectors was employed to classify audio segments into falls and various types of noise. Being poor in anti-interference and environmental factors, this method may have a high false positive rate, and thus can only be used for fall detection in some special circumstances.

Due to the limitations of the above-mentioned methods, vision-based fall detection using the video data captured from a single camera or multiple cameras has gained more and more attention in recent years and become one of the most widely used fall detection technologies, such as 2D video analysis. Human fall detection based on a single camera is a problem of detecting human fall behavior analyzing three-dimensional motion from a two-dimensional real-time dynamic video streams. 2D images of the human's shape have been widely used to detect falls [1, 6, 8, 12, 17, 25, 28, 32, 34, 37]. In [6, 18, 32], the ratio of the height to the width of a rectangle was used to determine whether a human falls. In [32], only the falls in two directions were detected by the spatial-temporal analysis of the SAR. In [6], the falls in the indoor environment were detected according to different human postures. In [17], three features were used, i.e. the human shape aspect ratio, the effective area ratio and the center variation rate, to detect human fall, but this method is complex and would fail since the human shape aspect ratio changes with the relative position of the camera and the target. In [28], the Canny detector was used to detect the human profile edge and the body shape based on shape deformation. Chua et al. [8] presented a new video-based fall detection technique based on human shape variation, where only three points are used to represent different regions of a human body, namely the head, body, and legs. Other works used external ellipses that closely encapsulate the contour of the human body to represent its shape [12, 25, 34, 37]. The ellipse method calculated the inclination of the human body and the ground to determine whether the human body began to fall or has already fallen down. It was also capable of removing the

humans carrying elongated objects such as a long-handled vacuum cleaner in order to obtain more accurate enclosing ellipse.

Some human detection methods in fall detection were discussed in literature, including the Gaussian Mixture Model (GMM) [27, 31, 32], the CodeBook background subtraction method [6, 11, 34], the improved background subtraction method [12] and the simple prospect detection method [22]. In an complex environment, especially in outdoor scenes, there are many factors affecting human detection and continuous tracking, such as human reflections and shadows, lighting changes, shaking branches, and partial occlusions. In [7], region-based ConvNet detector was applied to detecting the target, which achieved high accuracy with substantial training data. However, deep learning-based detection methods need GPU to accelerate the computation, which is not always available in the laboratories. Human tracking obtains the human's continuous moving trajectory and related information such as the speed and the shape. MeanShift [10] finds the most likely target location in the current video frame, and then tracks the humans. The temporal analysis of human motion feature is important to judge human fall. In [19], an approach was presented for complex activity recognition comprising of temporal pattern mining (TPM) and adaptive Multi-Task Learning (MTL). TPM captures the intrinsic properties of activities, while adaptive MTL uncovers the relatedness among activities and selects discriminative features. In [20], an efficient algorithm was proposed to identify temporal patterns among actions and utilize the identified patterns to represent activities for automated recognition.

In recent years, deep learning has greatly improved the accuracy of human action recognition [33, 36]. Although deep learning-based fall detection methods [1, 24, 35] show superior performance in terms of accuracy, it suffers from inferior performance in regard to speed. In order to achieve high accuracy, a number of videos containing fall actions need to be collected to train the neural network, but collecting such specific videos is not an easy task.

In this paper, we propose an automatic human fall detection method based on human motion tracking and the normalized shape aspect ratio (NSAR) in real-time videos. The human shape information and the aspect ratio has already been used in several previous works for fall detection [5, 13, 15, 27–29]. Compared with the existing works, our proposed method has the following advantages:

(1) It introduces the normalization since the change of the shape aspect ratio is caused by the relative postures and the distance between the human body and the camera.
(2) It employs the normalized shape aspect ratio together with the moving speed and directions to better detect human fall towards eight different directions.
(3) It smoothes the curve of the NSAR over time to eliminate the influence of hands and legs swinging back and forth.
(4) It is designed to be suitable for both indoor and outdoor environments.

The remainder of this paper are organized as follows. Section 2 overviews the major steps of our proposed method. Section 3 discusses human object detection and motion tracking. Section 4 explain the shape aspect ratio and its application to fall detection. The concept, algorithms and advantages of the normalized shape aspect ratio for fall detection are described in details in Section 5. Section 6 illustrates the experimental setup and results followed by Section 7 offering conclusions and future work.

2 Method overview

The human fall detection method proposed in this paper mainly includes the following six components: i) calibration; ii) bicubic interpolation; iii) table of calibrated shape aspect ratio(CSAR); iv) index table; v) analysis of motion characteristics; and vi) fall detection. These steps are either conducted offline or online, i.e., the calibration and bicubic interpolation of calibrated shape aspect ratio are processed offline, while the other steps are implemented online. The method overview is given in Fig. 1.

To calibrate the camera in a specific scene, CSAR is gathered uniformly via human walking through the whole video sequence. Then, the CSAR of each pixel in the image is calculated by bicubic interpolation of the gathered CSAR, which would be utilized to normalize the shape aspect ratio (SAR) of the tracked human. The results of bicubic interpolation form a table of CSAR of each pixel in the scene image stored in memory. The human shape is segmented from background and SAR is computed based on the bounding box enclosing it. The table of CSAR is indexed by the centric of the bounding box, and then the SAR is divided by the index value. Afterwards, we calculate and record the motion characteristics of each tracked human, including the moving trajectory of the human centroid, the moving speed history and the bounding boxes enclosing the human shape. If the NSAR of human changes substantially, the human body is judged as fall. The calculation of NSAR together with the moving speed and directions for detecting human fall will be discussed in details in Section 5.

3 Human object detection and motion tracking

3.1 Foreground detection

We adopt the foreground detection algorithm in [38], where only the body positive movement of the human is detected and to be judged whether it is consistent with human fall conditions. The algorithm uses a Gaussian mixture model to model the color distribution observed in any particular pixel in order to segment the foreground from background. The morphological
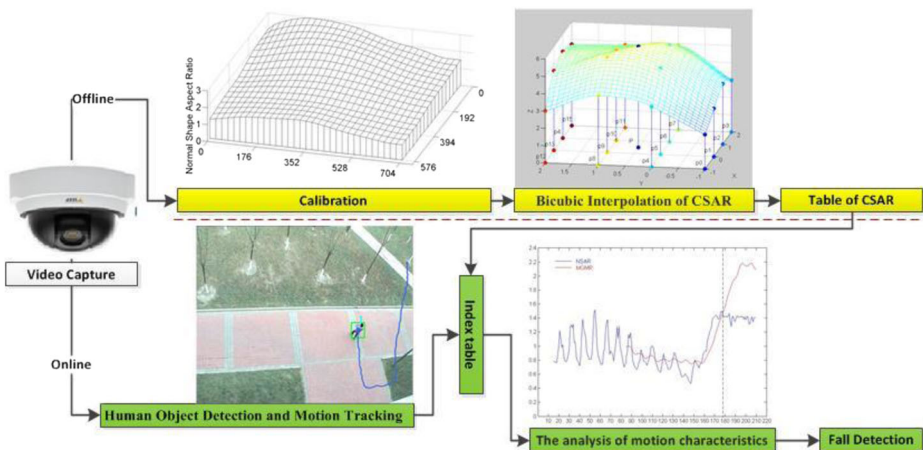

Fig. 1 The overview of the proposed human fall detection method

transformations such as erosion and dilation are then utilized to eliminate small patches and connect big patches. After that, the image patches whose area is bigger than a threshold are selected as humans.

This method can efficiently detect humans in a general environment. As an example, Fig. 2 shows a video frame and the human subject detected by the algorithm. When the human is walking from left to right throughout the video scene, s/he can be correctly detected across the whole walking process.

3.2 Human motion tracking

In this paper, we adopt the object tracking algorithm in [10] to find the most probable target position in the current video frame by MeanShift. The dissimilarity of the color distribution between the target model and the target candidate is expressed by a metric derived from the Bhattacharyya coefficient [4]. In the initial video frame, we firstly track a rectangle window defined for the area of the human. Then, MeanShift is used in the color space to track its area. The search for the new target location in the current frame starts at its estimated location in the previous frame. When the human moves, the algorithm estimates his/her most probable location in the current frame by maximizing the Bhattacharyya coefficient. The algorithm is fast and efficient for tracking non-rigid objects, and thus is quite suitable for human fall detection in real-time video monitoring.

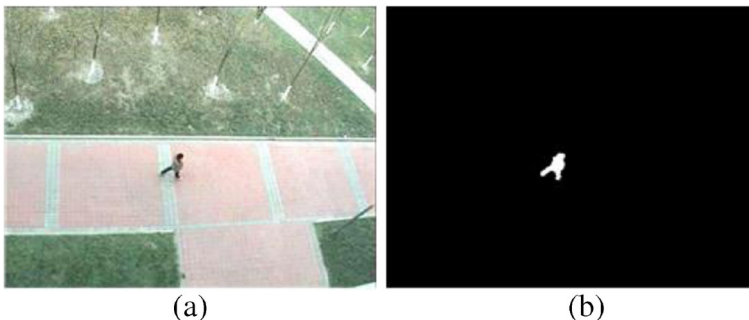The Bhattacharyya coefficient is defined by:

$$\rho = \int \sqrt{p(x)q(x)dx} \tag{1}$$

where p(x) and q(x) are the relative density functions for the discrete location of the target model, respectively. The derivation of the Bhattacharyya coefficient involves the estimation of the densities p and q, for which the histogram formulation is employed. The discrete density q is estimated from the m-bin histogram of the target model, as defined by:

$$q = \{q_u\}, u = 1, 2, \cdots, m \sum_{u=1}^{m} q_u = 1 \tag{2}$$

Similarly, the discrete density p is estimated at a given location y from the m-bin histogram of the target candidate, as defined by:

$$p(y) = \{p_u(y)\}, u = 12, \cdots, m \sum_{i=1}^{m} p_u = 1 \tag{3}$$

(a)　　　　　　　　　　　　　　　(b)

Fig. 2 Human detection: (a) is a video frame and (b) shows the human detection result using the foreground detection algorithm in [18]

Applying Eq. (2) and Eq. (3) to Eq. (1), the Bhattacharyya coefficient is recomputed by:

$$\rho(y) = \rho[p(y), q] = \sum_{u=1}^{m} \sqrt{p_u(y)q_u} \tag{4}$$

Let $\{x_1, x_2, \cdots, x_n\}$ be the pixel locations of the target model obtained from the color density function f(x). The target kernel density is given by:

$$q_u = \sum_{i=1}^{n} K(X_i)\delta[b(X_i)-u] \tag{5}$$

where K is a kernel weight function. Based on color density estimation, the MeanShift algorithm detects various candidate locations to maximize the Bhattacharyya coefficient as defined in Eq. (4) in order to obtain the most probable location y of the target in the current frame. Using Taylor expansion around the values $p_u(y_0)$, the Bhattacharyya coefficient in Eq. (1) is approximated by:

$$\rho[p(y), q] \approx \frac{1}{2}\sum\sqrt{p_u(y_0)q_u} + \frac{1}{2}\sum_{i=1}^{n} w_i K(X_i) \tag{6}$$

where

$$w_i \sum_{u=1}^{n} \delta[b(X_i)-u]\sqrt{\frac{q_u}{p_u(y_0)}} \tag{7}$$

In Eq. (6), since the first item does not change dramatically with different y, the Bhattacharyya coefficient is maximized when the second item reaches its maximum value. The maximum Bhattacharyya coefficient is achieved through the iterations in the MeanShift algorithm. This is done by deriving the new location y_1 from the current location y_0, given by:

$$y_i = \sum_{u=1}^{n} X_i w_i G(y_0-x_i) / \sum_{u=1}^{n} w_i G(y_0-x_i) \tag{8}$$

In this way, a human body can be continuously tracked based on the most probable location from the previous frame to the current frame when s/he walks. Figure 3 is an example of human tracking results when a human walks in a video scene, in which the green rectangle represents the detected human subject and the blue line records the walking path of the body centroid.

(a)   (b)
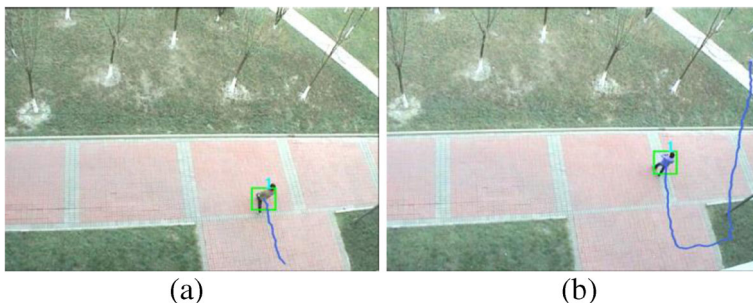
Fig. 3 Human tracking results, in which the green rectangle represents the human subject and the blue records the walking path of the body centroid

4 Shape aspect ratio for fall detection

4.1 Shape aspect ratio

In tracking, a human body is detected and encapsulated in a rectangle. The difference of telling the fall from the normal action depends on the human shape aspect ratio (SAR), which is defined by the height of the rectangle divided by the width of the rectangle. When a human falls, it is clear that the SAR changes dramatically, which is quite different from the small change in normal walking. In other words, the human SAR can be either larger than a high threshold or less than a low threshold. When a human falls, the speed usually decreases, the body centroid moves toward the ground, and s/he may fall toward different falling directions as shown in Fig. 4, which are: Upward, Downward, Leftward, Rightward, Up-Left, Up-Right, Down-Left and Down-Right. The falling toward four major falling directions, i.e., Upward, Downward, Leftward and Rightward will be discussed first. When a human walks from left to right in a video scene, falling rightward is also called as falling forward because the right direction in the video scene is the same as the walking direction; and falling leftward is also called as falling backward for the same reason.

4.2 Fall detection based on shape aspect ratio

Figure 5 shows a human walking normally from left to right and then falling forward, backward, upward and downward. In Fig. 5, the human body was judged to fall because the SARs change substantially. The process how SARs change is described in Fig. 6.

As we can see from the figure, four solid lines of different colors show the changes of the human SAR when s/he walks from left to right and falls forward, backward, upward and downward as in Fig. 5. The four dash-dot lines of different colors correspond to the falling time of the solid line with the same color. For example, the red solid line shows the walking involving falling backward; correspondingly, the red dash-dot line marks the time of falling backward.
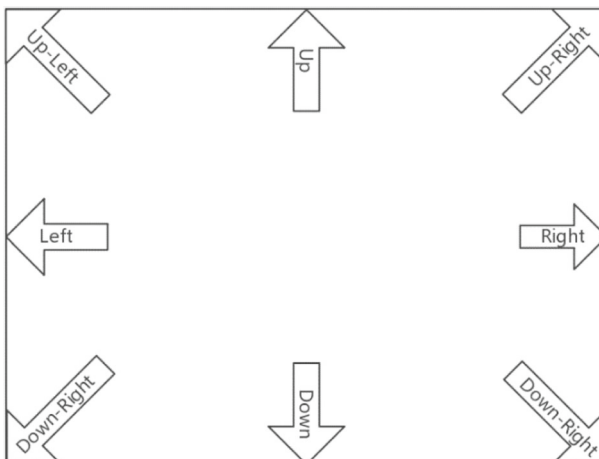

Fig. 4 The definition of eight falling directions in a video scene
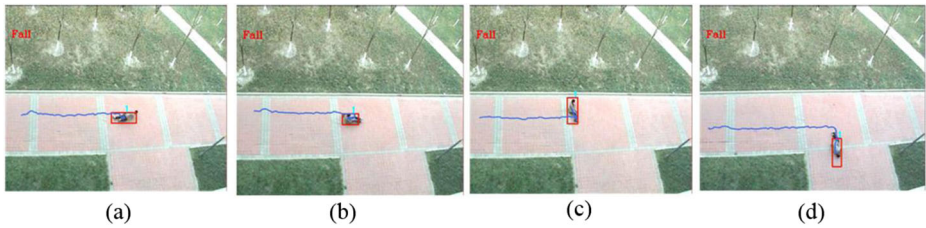
(a)   (b)   (c)   (d)

Fig. 5 A human walks normally and then falls toward four different directions

Apparently from the falling time, the SAR starts to change substantially. The SAR of falling upward and downward increases dramatically while the SAR of falling forward and backward decreases dramatically. In a word, the proposed method can successfully detecte the human fall according to the SAR changes, and it further distinguishes four falling directions using the moving directions of the human body centroid. The SAR decreases in both falling forward and falling backward directions, while the body centroid moves forward and backward, respectively.

5 Fall detection based on normalized shape aspect ratio

5.1 Normalized shape aspect ratio

As described above, the SAR can be used to successfully detect human fall when the ratio changes substantially. However, the SAR of the human walking normally may be quite different regarding the camera layouts and human locations. Therefore, we introduce the normalized shape aspect ratio (NSAR) to tackle this problem.
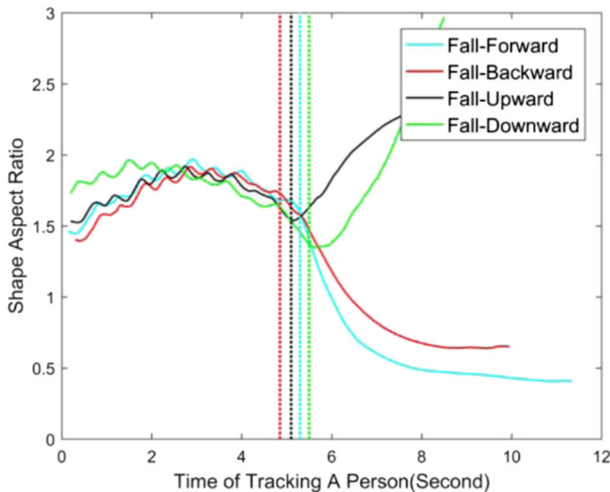

Fig. 6 Four solid lines of different colors show the changes of the height, width and the human SAR when s/he walks from left to right and falls to different directions. Four dash-dot lines of different colors correspond to the falling time of the solid line with the same color

For each position in the video scene, the SAR of a human walking normally is measured for each initially-positioned camera, which is called the calibrated shape aspect ratio (CSAR) at that position. Here, we define the NSAR as the actual SAR divided by the CSAR. Apparently, the NSAR should be close to 1 when the human walks normally. If the NSAR is largely different from 1, the human body is judged to be falling.

5.2 Generation of calibrated shape aspect ratio

The CSAR depends on each specific video scene, i.e. the location relationship between the video camera and the ground. It is gathered and calculated through a calibration process which detects the CSARs for a human walking through the whole video scene. The process has the following four steps:

(1) Divide the video scene into N*M equal rectangles.
(2) A human walks normally along the N + 1 horizontal lines from left to right in the video.
(3) Apply foreground detection to detecting the human by a bounding box, and then record the SAR of the box for every position.
(4) For each horizontal line, the SAR of the position, where the horizontal line intersects with M + 1 vertical lines, is selected. Thus, the sampled CSARs are generated by the SAR of (N + 1)*(M + 1) position which constitutes N*M equal rectangles.

Figure 7 shows sampled CSARs generated from the collected data during the experiments. The video scene is a rectangle area of 704-by-576 which corresponds to the camera resolution. As we can see, the sampled CASRs apparently are quite different at different positions. When the human walks throughout the video scene, the CSAR table is calculated by bicubic interpolation of the sampled CSARs around the pixels.

5.3 Bicubic interpolation of the CSAR

The bicubic interpolation is one of the most commonly used interpolation methods in three-dimensional space. The algorithm interpolates the value of 16 points around the
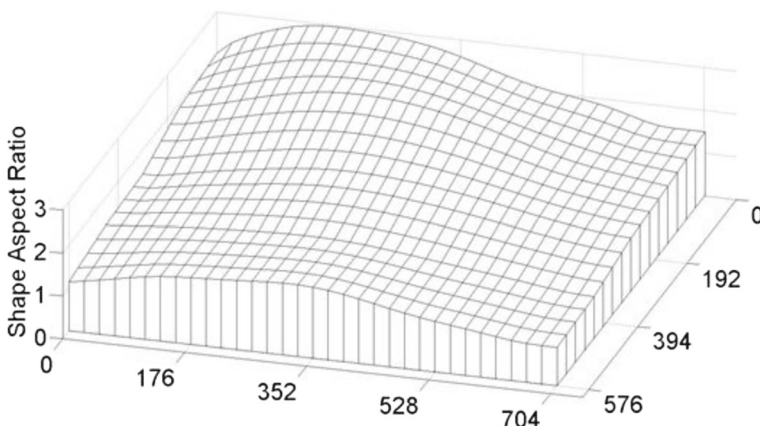

Fig. 7 The sampled CSAR

sample point, as shown in Fig. 8, which means it not only considers the influence of 4 directly adjacent points but also the changing rate of values between neighboring points.

In Fig. 8, $P(i) \mid i = 0, 1, \ldots, 15\}$ are known points where the coordinates of the point $P(5)$ are $(0, 0)$. For the sake of convenience, four adjacent points are considered as a unit square, e.g. $P(5):(0,0)$, $P(6):(1,0)$, $P(9):(0,1)$ and $P(10):(1,1)$. Suppose the values $f(x, y)$ and the derivatives $f_x(x, y)$, $f_y(x, y)$, $f_{xy}(x, y)$ are known at the four corners of this unit square. The interpolated table can be represented by

$$f(x,y) = \sum_{i=1}^{3} \sum_{j=0}^{3} a_{ij} x^i y^i \tag{9}$$

The first-order partial derivatives are computed by

$$f_x(x,y) = \sum_{i=0}^{3} \sum_{j=0}^{3} i a_{ij} x^{i-1} y^j \tag{10}$$

$$f_y(x,y) = \sum_{i=0}^{3} \sum_{j=0}^{3} j a_{ij} x^i y^{j-1} \tag{11}$$

The second-order partial derivative is computed by

$$f_{xy}(x,y) = \sum_{i=0}^{3} \sum_{j=0}^{3} i j a_{ij} x^{i-1} y^{j-1} \tag{12}$$
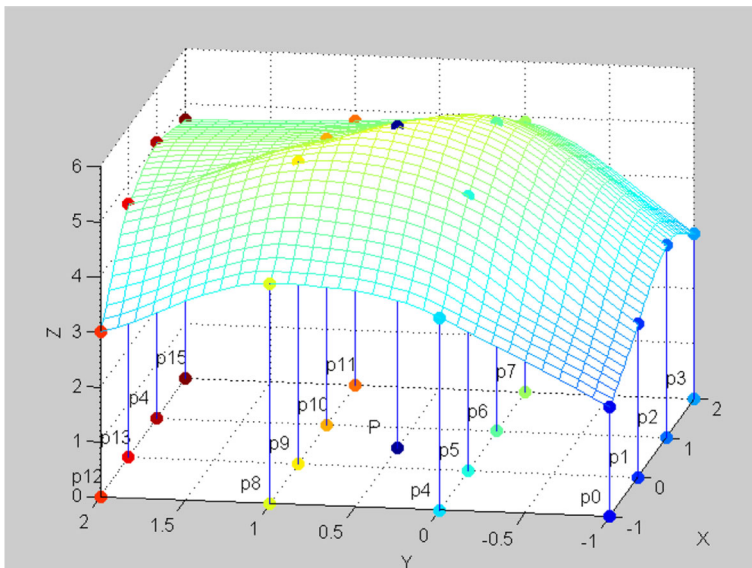

Fig. 8 Bicubic interpolation of CSAR

Applying the $f(x,y)$, $f_x(x,y)$, $f_y(x,y)$, $f_{xy}(x,y)$ of P(5), P(6), P(9) and P(10) to Eq. (9), Eq. (10), Eq. (11) and Eq. (12), respectively, there are 16 linear equations. In Eq. (9), there are 16 unknown coefficients which can be determined by

$$
\begin{bmatrix}
a_{00} & a_{01} & a_{02} & a_{03} \\
a_{10} & a_{11} & a_{12} & a_{13} \\
a_{20} & a_{12} & a_{22} & a_{23} \\
a_{30} & a_{31} & a_{32} & a_{33}
\end{bmatrix}
$$

$$
=
\begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 \\
-3 & 3 & -2 & -1 \\
2 & -2 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
f(0,0) & f(0,1) & f_y(0,0) & f_y(0,1) \\
f(1,0) & f(1,1) & f_y(1,0) & f_y(1,1) \\
f_x(0,0) & f_x(0,1) & f_{xy}(0,0) & f_{xy}(0,1) \\
f_x(1,0) & f_x(1,1) & f_{xy}(1,0) & f_{xy}(1,1)
\end{bmatrix}
\begin{bmatrix}
1 & 0 & -3 & 2 \\
0 & 0 & 3 & -2 \\
0 & 1 & -2 & 1 \\
0 & 0 & -1 & 1
\end{bmatrix}
$$

(13)

Finite difference is applied to computing the derivatives $f_x(x,y)$, $f_y(x,y)$, $f_{xy}(x,y)$ by Eq. (14) to (16).

$$
f_x(x,y) = \frac{f(x+1,y) - f(x-1,y)}{2}
$$

(14)

$$
f_y(x,y) = \frac{f(x,y+1) - f(x,y-1)}{2}
$$

(15)

$$
f_{xy}(x,y) = \frac{1}{8}(f(x+1,y+1) - f(x-1,y+1) - f(x+1,y-1) + f(x-1,y-1))
$$

(16)

5.4 Fall detection

The process of our proposed fall detection method is given in Fig. 9. After the human body is detected and tracked, the actual SAR is computed. The calibrated shape aspect ratio
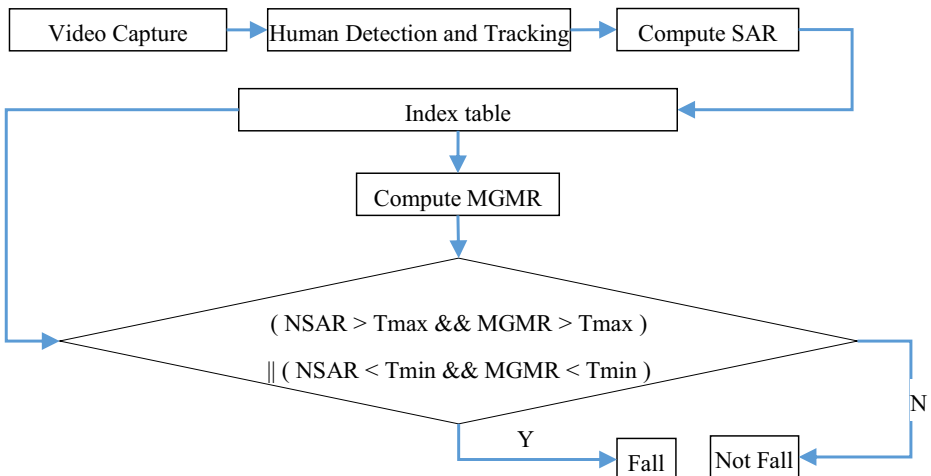

Fig. 9 The process of fall detection

(CSAR) is indexed by the position of tracked human body. Then, the NSAR is calculated as the actual SAR divided by the CSAR.

$$NSAR = \frac{SAR}{CSAR} \qquad (17)$$

Another parameter multi-frame geometric mean ratio (MGMR) is introduced to smooth the curve of the NSAR over time. If both NSAR and MGMR are bigger than a threshold Tmax or smaller than a threshold Tmin, the human body will be judged as fall.

The method proposed in this paper can successfully detect human fall toward four major directions according to NSAR, as shown in Fig. 10. As we can see, the NSAR remained close to 1 when the human walks normally. From the falling time, NSAR starts to change substantially and is quite different from 1. The NSARs of falling upward and falling downward become much larger than 1 while the NSARs of falling forward and falling backward become much smaller than 1.

However, even if a human moves normally rather than falling, the moving parts of the human body such as hands and legs also change the shape rectangle. The changing process in the rectangle is different between walking and falling. When a human walks normally with the hands and legs swinging back and forth regularly, the rectangle displays cyclic changes over time. Here, we divide the fall process into three stages: i) before falling; ii) falling; and iii) after falling. When a human is in the process of falling, the SAR changes gradually and eventually reaches a stable value. In our captured video, the sampling rate is 24 fps and the process from standing to falling lasts less than 24 frames, which takes about 1 s. Aside from the NSAR, we use MGMR in Eq. (18) to determine whether there is falling, which is the geometric mean of the ratio of the multiplication of the NSARs of multi-frames before falling to the multiplication of the NSARs of multi-frames after falling. In this paper, we choose 24 frames to compute the geometric mean. Let $\{BF(i)| i = 1, 2, 3\cdots, 24\}$ be the NSAR of the first 24 frames, $\{F(i)| i = 1, 2, 3\cdots, 24\}$ be the NSAR
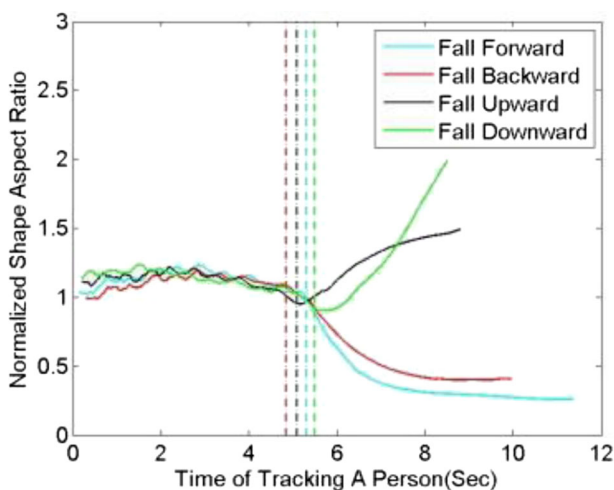
Fig. 10 Four solid lines of different colors show the changes of the human NSAR when a s/he walks from left to right and falls to different directions; four dash-dot lines of different colors correspond to the falling time of the solid line with the same color

of the middle 24 frames and $\{AF(i)|\,i=1,2,3\cdots,24\}$ be the NSAR of the third 24 frames. MGMR is defined by

$$MGMR = \left(\prod_{i=1}^{24} \left(AF(i)/BF(i) \right) \right)^{1/24} \tag{18}$$

Let T be the threshold of determining whether the human is falling based on the NSAR. From Fig. 10, the NSAR remained close to 1 when the human is walking normally, and we can see that $\{BF(i)|\,i=1,2,3\cdots,24\}$ is close to 1 and $\{AF(i)|\,i=1,2,3\cdots,24\}$ is approximately equal to T. $AF(i)/BF(i)$ with $i=1,2\cdots,24$ can be approximately computed by

$$AF(i)/BF(i) \approx T\, with\, i = 1,2\cdots,24 \tag{19}$$

Applying Eq. (20) to Eq. (19), MGMR is given by Eq. (20).

$$MGMR = \left(\prod_{i=1}^{24} \left(AF(i)/BF(i) \right) \right)^{1/24} \approx \left(\prod_{i=1}^{24} T \right)^{1/24} = T \tag{20}$$

It is proved in theory that the MGMR curve is similar to the NSAR curve. We set the threshold of MGMR as the same as in the NSAR. When a human is walking with hands swaying, the NSAR appears to be periodically varied. After the human falling down, the NSAR substantially varies which may last for a few seconds. By using the MGMR, the algorithm can reject the cyclic changes and a sudden change of the NSAR as fall, which reduces the false judgment. Because T is a limited value, the MGMR can be converged.

Figure 11 is an example of the curves of the NSAR and MGMR when a human walks from left to right and then falls downward. As we can see, the MGMR curve is smoother than the NSAR curve, where the falling time is at Frame 176 when the NSAR is 1.493 and the MGMR is 1.402. In fall detection, positive true rate should be as high as possible, so we set the smaller threshold as Tmax to be 1.4 and the bigger threshold as
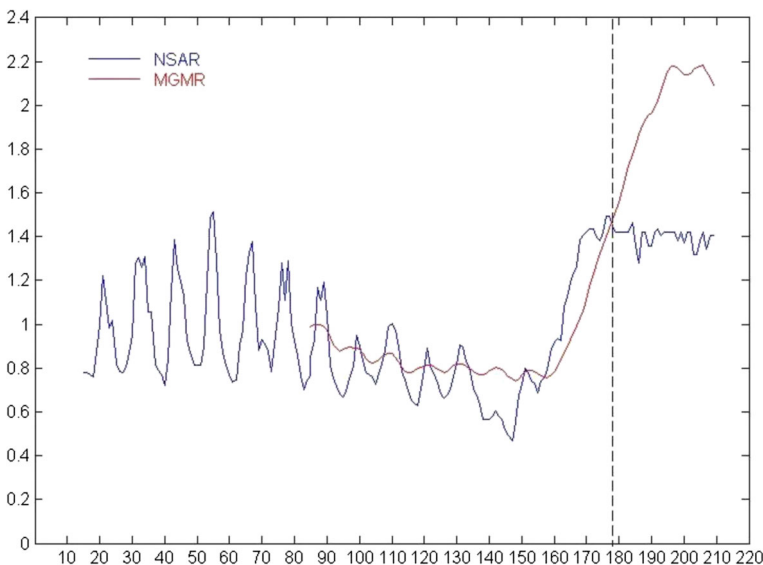

Fig. 11 The curves of the NSAR and MGMR

Tmin to be 0.5. The NSAR of Frame 57 is 1.758, if we do not use the MGMR, the human would be judged as fall by mistakes. The MGMR can effectively reduce the false positive rate. The experiment discussed below shows that the algorithm can successfully eliminate the effects of swinging hands and legs.

6 Experimental results and discussion

In this section, we test the performance of our proposed fall detection algorithm in an indoor and two outdoor environments.

6.1 The experimental results of indoor environments

In the indoor scene, a Hikvision camera was placed at about 3 m high and 45 degrees angle with the vertical line. Figure 12(a) is a snapshot of the scene. Before fall detection, the algorithm needs to measure the calibrated shape aspect ratio for each position of the scene. Figure 12(b) is the corresponding NSAR of the scene, where the red region represents relative high value of the NSAR, and the blue region represent relative low value of the NSAR. As we can see, most of the red region is located on the up-right corner, and most of the blue region is located on the down-left corner.

In the indoor experiment, two videos were captured with the camera resolution 960*720 at a 24-fps to validate the effectiveness of the proposed algorithm. Video I lasts 7 min and 14 s, in which a human walks normally and then falls for thirty-two times. Video II lasts 3 min, in which a chair was placed in the scene center, and the human's behavior includes walking, running, falling and taking another chair with him.

Figure 13 shows three snapshots from videoI. In Fig. 13(a) and (b), a human who walks normally then falls forward and backward respectively is correctly detected as fall by our algorithm. In Fig. 13(c), when a human who walks normally and then falls toward the camera, the NSAR does not change so much that the falling human is not judged as fall by mistake.
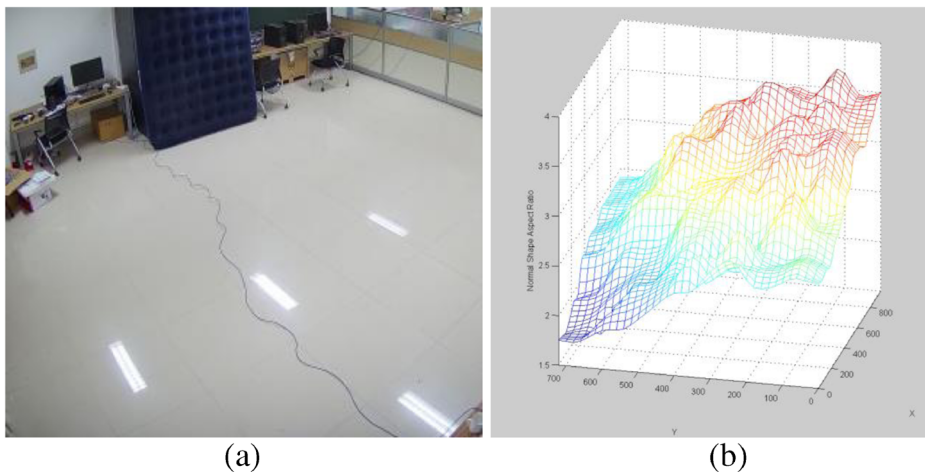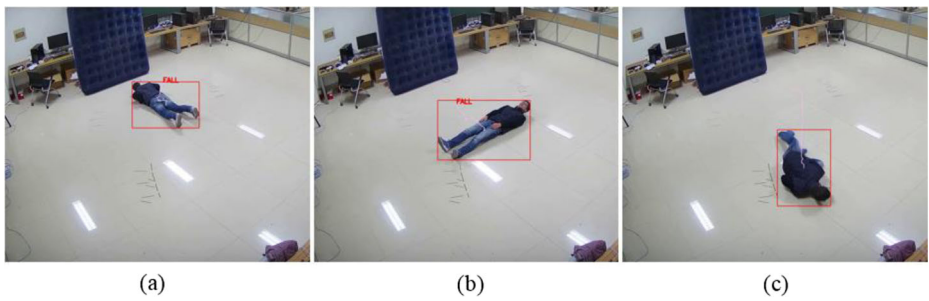

(a)   (b)

Fig. 12 The scene in the indoor environment and the corresponding NSAR

Fig. 13 The fall detection results using the NSAR in VideoIunder the indoor environment

Figure 14 shows six snapshots from VideoII. In Fig. 14(a), a falling human occluded by the chair is still correctly detected. Figure 14(b) shows that a human taking a chair is detected by mistake as falling because the chair extends the width of the bounding box of the human, and thus the NSAR becomes smaller. Figure 14(c) shows that a human is running. Figure 14(d), (e), (f) show that a falling human is successfully detected as falling at different scene locations.

We use precision P and recall R defined in Eq. (21) and Eq. (22) to evaluate the experiments.

$$P = \frac{TP}{TP + FP} \tag{21}$$

$$R = \frac{TP}{TP + FN} \tag{22}$$

Here, TP, FP and FN are the number of true positives, false positives and false negatives respectively.
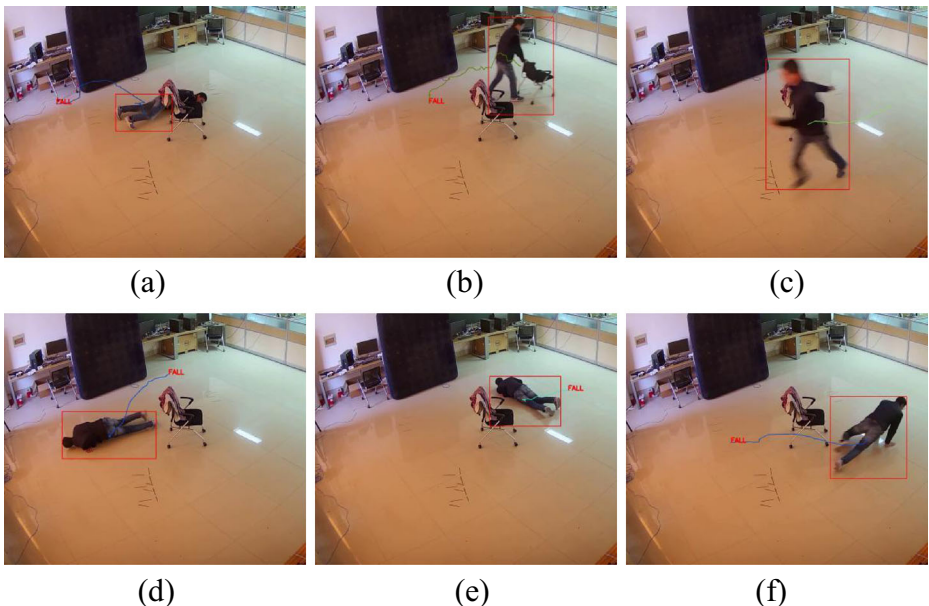
Fig. 14 The fall detection results using the NSAR in Video II under the indoor environment

Table 1 Shows the comparison of our NSAR-based fall detection method with SAR-based fall detection method in terms of precision and recall for VideoIand VideoII

		TP	FP	FN	P	R
NSAR	VideoI	29	0	3	100.00%	93.50%
	VideoII	5	1	0	83.33%	100.00%
SAR	VideoI	27	2	5	93.10%	84.38%
	VideoII	4	1	1	80.00%	80.00%

As seen in Table 1, the comparative experiment shows the performance of NSAR outperforms the SAR, for example, in VideoI, the precision of NSAR is 6.90% higher than that of SAR and the recall of NSAR is 9.12% higher than that based on SAR.

In Fig. 15, the experimental results of more walking directions and different falling directions are presented. In Fig. 15(a), (b), (c) and (d), a human walks upward from down side and falls toward four directions: forward, backward, left, right. Fig. 15(e), (f), (g) and (h) show that a human walks from left to right and falls toward the following four directions: Up-Left, Up-Right, Down-Left and Down-Right.

Figure 16 shows the changes of the SAR and the NSAR when a human walks upward from down side and falls in Fig. 15(a), (b), (c) and (d), respectively. Both the SAR and the NSAR can successfully detect human fall in four major falling directions, and they starts to change substantially from the falling time. In Fig. 16(a), the threshold for determining the human fall is related to every position in the walking process, which means the threshold may deviate with the true threshold when the SAR is not normalized. Therefore, the algorithm using the SAR could not detect fall because it has already increased substantially before the fall happens.

Figure 17 shows the changes of the SAR and the NSAR when a human walks from left to right and falls to another four directions, which are Up-Left, Up-Right, Down-Left and Down-Right in Fig. 15(e), (f), (g), (h), respectively. In Fig. 17(b), the threshold is independent of each position by using the NSAR which is close to 1 when the human walks normally. From the falling time, the NSAR starts to change substantially and is much less than 1. The method proposed in this paper can successfully detect the human falls toward these four directions according to the changes of the NSAR.

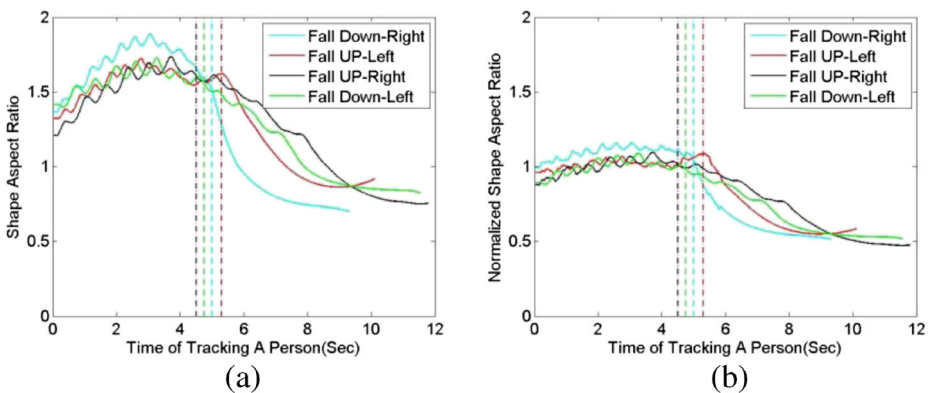| (a)	(b)	(c)	(d)
(e)	(f)	(g)	(h)
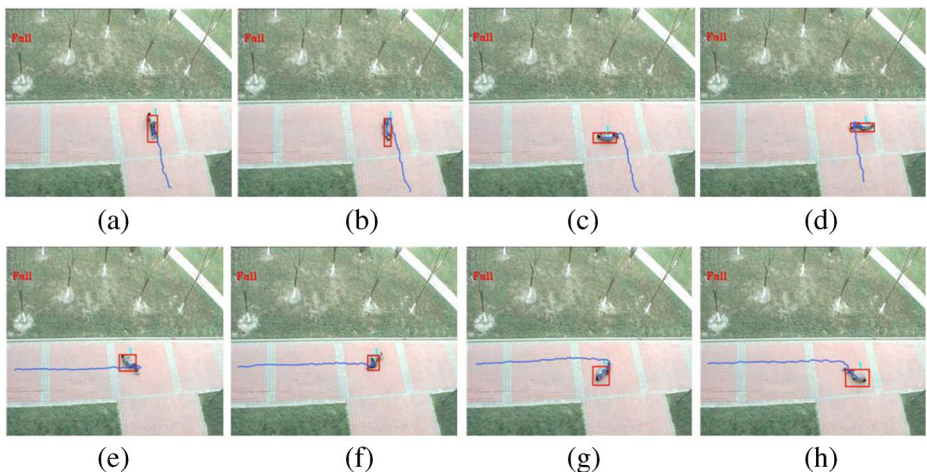
**Fig. 15** A human falls toward eight different directions
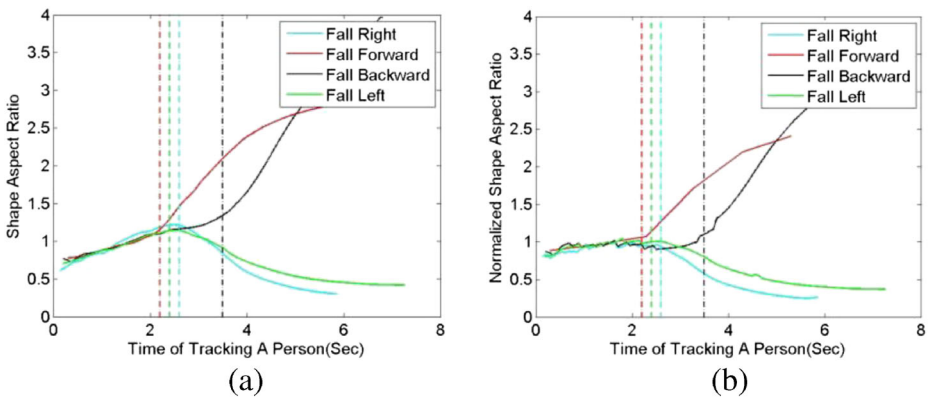
Fig. 16 Four solid lines of different colors show the changes of the human SAR and NSAR when s/he walks upward from down side and falls forward, backward, left and right. Our dash-dot lines of different colors correspond to the falling time of the solid line with the same color

6.2 The experimental results of outdoor environments

To test the algorithm's robustness, we captured a video in an outdoor scene using a Hikvision camera placed ten-meter high from the ground on a road junction in the campus of Nanchang University in China, which is a more realistic scenario with shaking branches, shadows, cars, or humans walking by. The video lasts 4 min and 28 s, consisting of 6432 frames with resolution 768*567 at a 24-fps sampling rate. Five snapshots are given in the first column in Fig. 18. The second column represents the corresponding foreground detection results that only the target with its size within a threshold is recognized as a human subject and encapsulated in a rectangle. The third column represents the corresponding tracking results. In this situation, Fig. 18(a) shows the scene background with wavering tree branches. In Fig. 18(b), there are two persons walking cross each other. In Fig. 18(c), a person riding on a bicycle and a person riding on an electric bicycle appear in the scene at the same time. In Fig. 18(d), a car is moving from down-left to up-right and the person is walking in the sunshine. In Fig. 18(e), there are up to five persons in the scene simultaneously.


Fig. 17 Four solid lines of different colors show the changes of the human SAR and NSAR when s/he walks and falls in the following four directions: Up-Left, Up-Right, Down-Left and Down-Right. Four dash-dot lines of different colors correspond to the falling time of the solid line with the same color
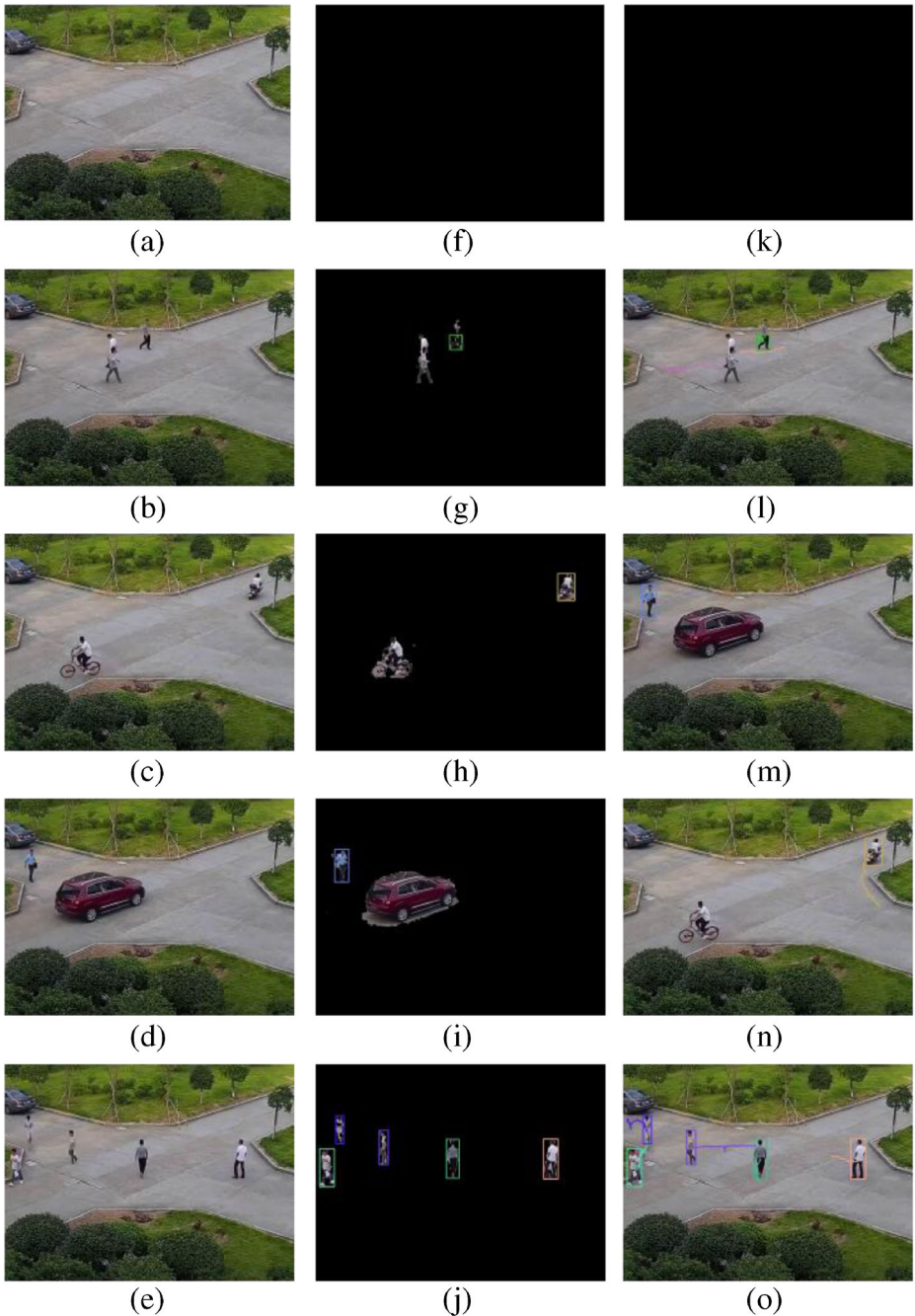
Fig. 18 The results of foreground detection and tracking in our experiment

Fig. 18(f) is the foreground detection result of Fig. 18(a). In Fig. 18(g), when two persons are walking cross each other, the area of the cross section is larger than that when a single person is normally walking by. In this case, these two persons would not
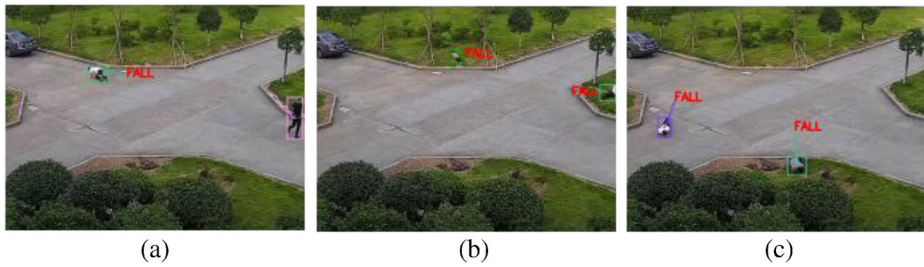
Fig. 19 Human fall detection: (**a**), (**b**), (**c**) show five falls with different poses detected by our algorithm in real-world scenarios. In (**a**), the left person falls toward Up-Left; in (**b**), the left person falls toward Up-Left and the right person falls rightwards; in (**c**), the left person and the right person fall downwards at the same time

be detected by our foreground detection method. After the cross, the algorithm can successfully detect these two persons again. The moving car in Fig. 18(h) and the person riding on a bicycle in Fig. 18(i) are also rejected because of the same reason as in Fig. 18(b). Figure 18(i) shows that the human shadow is removed by shadow detection. Figure 18(j) shows that our algorithm can detect five human subjects successfully in the complex scene.

Figure 18(k) is the tracking result of Fig. 18(a). In Fig. 18(l), when two persons are walking cross each other, they would be lost by our tracking method because they could not be detected by foreground detection. After the cross, the algorithm restarts and can successfully track them. When they fall, the NSAR presents the characteristics of temporarily substantial changes, so the algorithm does not have to track the whole trajectories of humans and this process will not affect the final fall detection results. In Fig. 18(m) and (n), the algorithm can successfully track the motion trajectories of the moving person and the moving electric bicycle, respectively. Fig. 18(o) shows that five persons are tracked successfully.

In this experiment, volunteers fall five times and all falls are detected by our algorithm. Figure 19 shows the detection results. In Fig. 19(a), the fall of the left person is detected successfully and the upright person who is close to the right scene edge is not detected. In Fig. 19(b), the fall of two persons located on the meadow are all detected successfully. Furthermore, the fall of the right person who was partially out of the scene after his falling is detected by our proposed algorithm. In Fig. 19(c), another example demonstrates that two persons falling simultaneously can be detected successfully by the proposed algorithm.

Table 2 shows the fall detection results for this outdoor experiment by our proposed method. In this complex scene, the algorithm reports 100% true positive rate, in which only 2 frames are falsely detected among the 6432 frames.

Our algorithm is real-time. The speed of our method can reach 20 fps on Lenovo E42 (Inter(R) Core(TM) i7-6500 U CPU @2.5G). The experimental results demonstrate that our proposed method performs very well in the aspect of the positive rate, the false positive rate, and the running speed.

Table 2 The fall detection results for this outdoor experiment

	TP	FP	FN	P	R
Outdoor	5	2	0	71.43%	100%

7 Conclusions and future work

In this paper, a human fall detection method based on real-time human motion tracking and the normalized shape aspect ratio is proposed. While most of the existing methods detect falls at home, this method is suitable for both indoor and outdoor environments and also capable of detecting falls toward eight different directions. The methods using the shape aspect ratio which may fail to detect human fall when the relative position and the distance between the human and the camera changes. The proposed method rectifies the changes to detect human fall based on normalized shape aspect ratio, which is defined as the actual SAR divided by the CSAR. Furthermore, we propose the multi-frame geometric mean ratio to smooth the curve of the normalized shape aspect ratio to eliminate the influence of hands and legs swinging back and forth. Last but not the least, our algorithm is real-time on an ordinary laptop without GPU. However, the algorithm has a limitation, i.e., an offline calibration process is needed to gather the sampled calibrated shape aspect ratio. In the future, we will explore the photographic geometry theory for calibrating the camera automatically. What is more, we will combine deep learning technologies to carry out the research on human activity recognition to distinguish human fall from normal daily behaviors such as bending, crouching, sitting down, etc.

References

1. Alhimale L, Zedan H, Al-Bayatti A (2014) The implementation of an intelligent and video-based fall detection system using a neural network. Appl Soft Comput 18:59–69
2. Alwan M, Rajendran PJ, Kell S, et al (2006) A Smart and Passive Floor-Vibration Based Fall Detector for Elderly. In: 2nd International Conference on Information & Communication Technologies, Damascus, pp. 1003–1007
3. Augustyniak P, Smoleń M, Mikrut Z, Kańtoch E (2014) Seamless tracing of human behavior using complementary wearable and house-embedded sensors. Sensors 14:7831–7856
4. Bhattacharyya A (1942) On a measure of divergence between two statistical populations defined by their probability distributions. Bull Calcutta Math Soc 35:99–109
5. Chen M-C (2016) A video surveillance system designed to detect multiple falls. Adv Mech Eng 8(4):1–11
6. Chen M-C, Liu Y-M (2013) An indoor video surveillance system with intelligent fall detection capability. Math Probl Eng:1–8
7. Chu J, Guo Z, Leng L (2018) Object Detection Based on Multi-Layer Convolution Feature Fusion and Online Hard Example Mining. IEEE Access PP:1–1
8. Chua J-L, Chang YC, Lim WK (2015) A simple vision-based fall detection technique for indoor video surveillance. SIViP 9:623–633
9. Cola G, Avvenuti M, Vecchio A et al (2015) An on-node processing approach for anomaly detection in gait. IEEE Sensors J 15:6640–6649
10. Comaniciu D, Ramesh V, Meer P (2000) Real-time tracking of non-rigid objects using mean shift. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 142–149
11. Feng P, Yu M, Naqvi SM, Chambers JA (2014) Deep learning for posture analysis in fall detection. In: 19th International Conference on Digital Signal Processing, 12–17

12. Feng W, Liu R, Zhu M (2014) Fall detection for elderly person care in a vision-based home surveillance environment using a monocular camera. SIViP 8:1129–1138
13. Hazelhoff L, Han J, de With PHN (2008) Video-Based Fall Detection in the Home Using Principal Component Analysis. Advanced Concepts for Intelligent Vision Systems. 298–309
14. Igual R, Medrano C, Plaza I (2013) Challenges, issues and trends in fall detection systems. Biomed Eng Online 12:66
15. Ji T, Turjo M, Wong M-F, et al (2005) Fall Incidents Detection for Intelligent Video Surveillance. In: 5th International Conference on Information Communications & Signal Processing. 1590–1594
16. Lai C-F, Chen M, Pan J-S et al (2014) A collaborative computing framework of cloud network and WBSN applied to fall detection and 3-D motion reconstruction. IEEE J Biomed Health Inform 18:457–466
17. Li L, Huang W, Gu IYH, Tian Q (2003) Foreground Object Detection from Videos Containing Complex Background. In: Proceedings of the 11th ACM International Conference on Multimedia. 2–10
18. Liu H, Zuo C (2012) An improved algorithm of automatic fall detection. AASRI Procedia 1:353–358
19. Liu Y, Nie L, Han L, et al. (2015) Action2Activity: Recognizing Complex Activities from Sensor Data. In: Proceedings of the 24th International Joint Conference on Artificial Intelligence. 1617–1623
20. Liu Y, Nie L, Liu L, Rosenblum DS (2016) From action to activity: sensor-based activity recognition. Neurocomputing 181:108–115
21. Medrano C, Plaza I, Igual R et al (2016) The effect of personalization on smartphone-based fall detectors. Sensors 16(1):117
22. Mirmahboub B, Samavi S, Karimi N, Shirani S (2013) Automatic monocular system for human fall detection based on variations in Silhouette area. IEEE Trans Biomed Eng 60:427–436
23. Mubashir M, Shao L, Seed L (2013) A survey on fall detection: principles and approaches. Neurocomputing 100:144–152
24. Núñez-Marcos A, Azkune G, Arganda-Carreras I (2017) Vision-Based Fall Detection with Convolutional Neural Networks. Wireless Communications and Mobile Computing Vol. 2017, Article ID 9474806, 16
25. Rhuma A, Yu M, Chambers J (2013) Posture Recognition Based Fall Detection System. Lecture Notes on Software Engineering. 350–355
26. Rimminen H, Lindstrom J, Linnavuo M, Sepponen R (2010) Detection of falls among the elderly by a floor sensor using the electric near field. IEEE Trans Inf Technol Biomed 14:1475–1476
27. Rougier C, Meunier J, St-Arnaud A, Rousseau J (2007) Fall Detection from Human Shape and Motion History Using Video Surveillance. In: 21st International Conference on Advanced Information Networking and Applications Workshops. 875–880
28. Rougier C, Meunier J, St-Arnaud A, Rousseau J (2011) Robust video surveillance for fall detection based on human shape deformation. IEEE Trans Circuits Syst Video Technol 21:611–622
29. Rougier C, Meunier J, St-Arnaud A, Rousseau J (2013) 3D head tracking for fall detection using a single calibrated camera. Image Vis Comput 31:246–254
30. Tabar AM, Keshavarz A, Aghajan H (2006) Smart home care network using sensor fusion and distributed vision-based reasoning. In: 4th ACM International Workshop on Video Surveillance and Sensor Networks Pages 145–154
31. Tra K, Pham TV (2013) Human fall detection based on adaptive background mixture model and HMM. international conference on advanced technologies for. Communications:95–100
32. Xiao H-E, Wang Z-L, Li Q, Wang N-M (2013) Gaussian mixture model for background based automatic fall detection. Institution of Engineering and Technology, 234–237
33. Xu H, Das A, Saenko K (2017) R-C3D: Region Convolutional 3D Network for Temporal Activity Detection. In: 2017 IEEE International Conference on Computer Vision (ICCV). 5794–5803
34. Yu M, Naqvi SM, Rhuma A, Chambers J (2012) One class boundary method classifiers for application in a video-based fall detection system. IET Comput Vis 90–100
35. Yu M, Gong L, Kollias S (2017) Computer vision based fall detection by a convolutional neural network. In: proceedings of the 19th ACM international conference on multimodal. Interaction:416–420
36. Zhu H, Vial R, Lu S (2017) TORNADO: A Spatio-Temporal Convolutional Regression Network for Video Action Proposal. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 5813–5821
37. Zhuang X, Huang J, Potamianos G, Hasegawa-Johnson M (2009) Acoustic fall detection using Gaussian mixture models and GMM supervectors. In: IEEE International Conference on Acoustics, Speech and Signal Processing. 69–72
38. Zivkovic Z (2004) Improved adaptive Gaussian mixture model for background subtraction. In: Proceedings of the 17th IEEE International Conference on Pattern Recognition. 2; 28–31

Weidong Min obtained his BE, ME and PhD of computer application at Tsinghua University in China in 1989, 1991 and 1995, respectively, on the research subjects of computer graphics, image processing and computer aided geometric design. He was an assistant professor of Tsinghua University from 1994 to 1995. From 1995 to 1997 he was a postdoctoral researcher at University of Alberta, Canada. From 1998 to 2014 he worked as a senior researcher and senior project manager at Corel and other companies in Canada. In recent years, he cooperated with School of Computer Science & Software Engineering, Tianjin Polytechnic University, China. From 2015 he is a professor at School of Information Engineering, Nanchang University, China. He is a Member of "The Recruitment Program of Global Expert" of Chinese government. He is an executive director of China Society of Image and Graphics. His current research interests include computer graphics, image and video processing, distributed system, software engineering and network management.
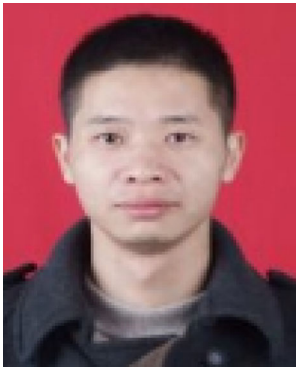
Song Zou obtained his master degree of Computer software and theory in Zhejiang University in China in 2011. He is a PhD student in the School of Information Engineering in Nanchang University. His research interests include vision-based human action understanding and abnormal behavior detection.

Jing Li obtained her PhD degree in Electronic and Electrical Engineering from the University of Sheffield, UK, in 2012. Before joining Nanchang University as an Associate Professor, she was a Research Associate at the University of Sheffield. Her research interests include human behavior analysis, object recognition, visual tracking and scene understanding in complex environments. She has authored or coauthored in various journals, such as IEEE Transactions on Industrial Informatics, Information Sciences (Elsevier), etc.