

Automatic Detection of Human Fall in Video

Vinay Vishwakarma, Chittaranjan Mandal, and Shamik Sural

School of Information Technology
Indian Institute of Technology, Kharagpur, India
{vvinay,chitta,shamik}@sit.iitkgp.ernet.in

Abstract. In this paper, we present an approach for human fall detection, which has important applications in the field of safety and security. The proposed approach consists of two parts: object detection and the use of a fall model. We use an adaptive background subtraction method to detect a moving object and mark it with its minimum-bounding box. The fall model uses a set of extracted features to analyze, detect and confirm a fall. We implement a two-state finite state machine (FSM) to continuously monitor people and their activities. Experimental results show that our method can detect most of the possible types of single human falls quite accurately.

1 Introduction

Human fall is one of the major health problems for elderly people. Falls are dangerous and often cause serious injuries that may even lead to death. Fall related injuries have been among the five most common causes of death amongst the elderly population. Falls represent 38% of all home accidents and cause 70% of death in the 75+ age group. It is shown in [1] that the number of reported human falls per year was around 60,000 with an associated cost of at least £400 million in the UK.

Early detection of a fall is an important step in avoiding any serious injuries. An automatic fall detection system can help to address this problem by reducing the time between the fall and arrival of required assistance. Here, we present an approach for human fall detection using a single camera video sequence. Our approach consists of two steps: object detection and the use of a fall model. We apply an adaptive background subtraction method to detect a moving object and mark it with its minimum-bounding box. The fall model consists of two parts: fall detection and fall confirmation. It uses a set of extracted features to analyze, detect and confirm a fall. In the fall model, the first two features (aspect ratio, horizontal and vertical gradient values of an object) are responsible for fall detection and the third feature (fall angle) is used for fall confirmation. We also implement a two-state finite state machine to continuously monitor people and their activities.

The organization of the paper is as follows. Section 2 explains the related work on fall detection. Section 3 describes the object detection method. Section 4 elaborates the fall model. In Section 5, we present experimental results followed by conclusion in section 6.

2 Related Work

Primarily, there are three methods of fall detection, classified in the following categories:

1. Acoustics based Fall Detection
2. Wearable Sensor based Fall Detection
3. Video based Fall Detection

In video based fall detection, human activity is captured in a video that is further analyzed using image processing techniques. Since video cameras have been widely used for surveillance as well as home and health care applications, we use this approach for our fall detection method.

Due to the advancements in vision technologies, many individuals and organizations are concentrating on fall detection using video based approaches. In [2], authors have used background modeling and subtraction of video frames in HSV color space. An on-line hypothesis-testing algorithm is employed in conjunction with a finite state machine to infer fall incident detection. However, they only use aspect ratio of a person as an observation feature based on which fall incident is detected. Lue and Hu [3] have presented a fall detection algorithm using dynamic motion pattern analysis. They assume that a fall can only start when the subject is in an upright position and characterizes a big change in either X or Y direction when a fall starts. In [5], Toreyinet al. have used a background estimation method to detect moving regions. Using connected component analysis, they obtain the minimum bounding rectangles (blob) and calculate the aspect ratio. They also use audio channel data based decisions and fuse it with video data based decisions to reach a final decision. In [6], authors subtract the current image from the background image to extract the foreground of interest. To obtain the associated threshold, they consider a subject's personal information such as weight and height. Each extracted aspect ratio is validated with the user's personal information to detect the fall. In [7], McKenna and Nait-Charif [7] have used a particle filtering method to track a person and extract his trajectories using 5-D ellipse parameter space in each sequence. An associated threshold on a person's speed is used to label the inactivity zone and human fall. In [8], authors use 3-D velocity as a feature parameter to detect human fall from a single camera video sequence. At first, 3-D trajectory is extracted by tracking a person's head with the help of a particle filter as it has a large movement during a fall. Next, 3-D velocity is computed from 3-D trajectory of the head.

Most of the existing vision based fall detection systems use either motion information or a background subtraction method for object detection. An abrupt change in the aspect ratio is analyzed in different ways such as Hidden Markov Model (HMM), adaptive threshold and the user's personal information to detect falls in video. A person's velocity is often used to classify a human either as walking or falling in a video. There are also some other existing models, but they work well only in restricted environment.

In our approach, we use an adaptive background subtraction method using a Gaussian Mixture Model (GMM) in YCbCr color space for object detection. We propose a fall model that consists of two steps, first fall detection and then fall confirmation. We extract three features from an object and use the first two features for fall detection and the last for fall confirmation. We implement a simple two-state finite state machine (FSM) to continuously monitor people and their activities.

3 Object Detection

The first and the most important task of human fall detection is to detect humans accurately so we apply an adaptive background subtraction method using a Gaussian mixture model (GMM) and then extract a set of features for fall modeling.

3.1 Background Modeling and Subtraction

A recorded video is used as an input which is stored as a sequence of frames using the Berkeley MPEG Decoder. For every frame, we convert its pixel from the RGB color space to the YCbCr color space. We use mean values of image pixels for further processing.

GMM considers each background pixel as a mixture of Gaussians. The Gaussians are evaluated using a simple heuristic to hypothesize pixels which are most likely to be part of the background process. The probability that an observed pixel has intensity value X_t at time t is modeled by a mixture of K Gaussians as

$$P(X_t) = \sum_{i=1}^k w_{i,t} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t}). \quad (1)$$

where

$$\eta(X_t, \mu_{i,t}, \Sigma_{i,t}) = \frac{1}{((2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{n}{2}})} * e^{\frac{1}{2}(X_t - \mu_t)^T \Sigma^{-1}(X_t - \mu_t)}. \quad (2)$$

$$\omega_{i,t} = (1 - \alpha)\omega_{i,t-1} + \alpha(M_{k,t}). \quad (3)$$

Here m is the mean, α is the learning rate and $M_{k,t}$ is 1 for the model which matches and 0 for the rest.

The background estimation problem is addressed by specifying the Gaussian distributions which have the most supporting evidence and the least variance. Since a moving object has larger variance than a background pixel, in order to represent background process, first the Gaussians are ordered by the value of $\frac{\omega}{\sigma^2}$ in decreasing order. The background process stays on top with the lowest variance by applying a threshold T , where

$$B = \operatorname{argmin}_b \left(\sum_{k=1}^b \omega_k \geq T \right). \quad (4)$$

All pixels X_t which do not match any of these components will be marked as foreground.

Pixels are partitioned as being either in the foreground or in the background and marked appropriately. We apply connected component analysis that can identify and analyze each connected set of pixels to mark the rectangular bounding box over an object.

3.2 Feature Extraction

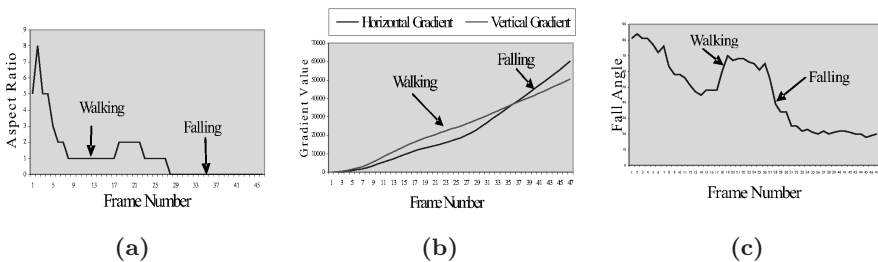
We extract a set of features from each object and its bounding box such as aspect ratio, horizontal (G_x) and vertical (G_y) gradient values and fall angle, which we use further in the fall model.

Aspect Ratio. The aspect ratio of a person is a simple yet effective feature for differentiating normal standing pose from other abnormal pose. In Table 1(a), we compare the aspect ratio of an object in different human pose.

Horizontal and Vertical Gradients of an Object. When a fall starts, a major change occurs in either X or Y direction. For each pixel, we calculate its horizontal (G_x) and vertical (G_y) gradient values. In Table 1(b), we compare horizontal and vertical gradient value of an object's pixel in different human pose.

Fall Angle. Fall angle (θ) is the angle of a vertical line through the centroid of object with respect to the horizontal axis of the bounding box. Centroid (C_x, C_y) is the center of mass co-ordinates of an object. In Table 1(c), we compare fall angles of an object in different pose such as walking and falling.

Table 1. Comparison of features distribution of object in different pose



4 Fall Model

Building a fall Model consists of two steps: fall detection and fall confirmation. For the fall detection step, we use aspect ratio and object's horizontal and vertical gradient values. For the fall confirmation, we use fall angle with respect to the horizontal axis of its bounding box. We use rule-based decisions to detect and confirm the fall.

4.1 Fall Detection

1. Aspect ratio of human body changes during fall. When a person falls, the height and width of his bounding box changes drastically.
2. When a person is walking, the horizontal gradient value is less than the vertical gradient value ($G_x < G_y$) and when a person is falling, the horizontal gradient value is greater than the vertical gradient value ($G_x > G_y$).
3. For every feature, we assign a binary value. If the extracted feature satisfies the rules, we assign binary value 1 otherwise 0.
4. We apply OR operation on the feature values. If we get the resultant binary value as 1 then we detect the person as falling, otherwise not.

4.2 Fall Confirmation

When a person is standing, we assume that he is in an upright position and the angle of a vertical line through the centroid with respect to horizontal axis of the bounding box should be approximately 90 degree. When a person is walking, the θ value varies from 45 degree to 90 degree (depending on their style and speed of walking) and when a person is falling, the angle is always less than 45 degrees.

For every frame where a fall has been detected, we apply the fall confirmation step. We calculate the fall angle (θ) and if θ value is less than 45 degree, we confirm that the person is falling. Similarly, we take next few (in our approach, it is seven) frames and analyze their features using the fall model to confirm a fall situation.

4.3 State Transition

To continuously monitor human behavior, which changes from time to time, we implement a simple two-state finite state machine (FSM). As shown in Fig. 1, the two states are 'Walk' and 'Fall' respectively.

Rule 1: Feature values should satisfy fall detection model.

Rule 2: Feature values should satisfy fall confirmation model.

When current state is 'Walk', the system begins to perform Rule 1 testing. If Rule 1 is not satisfied, the state remains unchanged; otherwise the state transits to 'Fall'. When current state is 'Fall', the system begins to perform Rule 2 testing. If Rule 2 is satisfied, the state remains unchanged; otherwise it transits back to 'Walk' state, which is the case when a person has fallen and again started to walk. Alarms will be triggered once a person remains in the state of 'Fall' for a period longer than a pre-set duration.

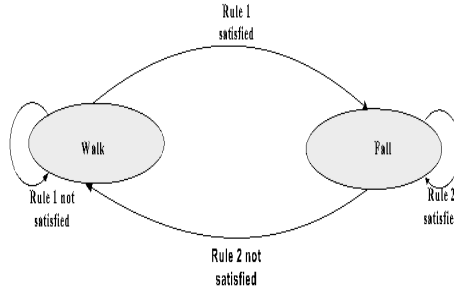


Fig. 1. A finite state machine for human fall detection

5 Experimental Results

We have implemented our proposed approach using C as the programming language in linux and tested it intensively in a laboratory environment. To verify the feasibility of our proposed approach, we have taken 45 video clips (indoor, outdoor and omni-video) as our test target. A handycam (SONY DCR-HC40E MiniDV PAL Handycam Camcorder) was used to capture indoor and outdoor video clips. Video clips contain a number of different possible types of human fall (sideway, forward and backward) and no fall condition. In every video clip, one or more moving object exists in the scene. In this paper, we use a set of criteria to evaluate our system including accuracy, sensitivity and specificity [6].

Table 2. Recognition results

Video Types	Scene Types	Total Frames	Fall Types	TP	FP	FN	TN
Indoor	Single	93	Forward	20	0	0	73
Indoor	Single	216	Backward	56	0	0	150
Indoor	Single	286	Sideway	76	0	0	210
Indoor	Single	100	No Fall	0	0	0	100
Outdoor	Single	87	Forward	47	0	0	40
Outdoor	Single	141	Backward	14	0	0	127
Indoor	Multiple	175	Sideway	80	30	15	50
Outdoor	Multiple	624	Sideway	144	10	120	350
Omni-video	single	376	Forward	100	10	10	256
Omni-video	Multiple	1007	Forward	257	50	440	260

In Table 2, we show results of indoor, outdoor and omni-video clips containing different types of possible human fall. In our experiment, a fixed threshold is set for every feature. For aspect ratio, we set the threshold between 0 and 1. For horizontal (G_x) and vertical (G_y) gradient values of an object, G_x is less than G_y in the case of a walking person and G_x is greater than G_y in the case of a falling person.

For fall angle (θ), θ is between 45 degree to 90 degree in the case of a walking person and θ is less than 45 degree in the case of a falling person. We have selected these thresholds empirically. In our experiments, we first evaluate the system performance (accuracy, sensitivity and specificity) using aspect ratio as a feature. Next, we evaluate system performance using our proposed fall model as shown in table 3. Results of aspect ratio as a feature parameter are shown in parenthesis. The experimental results show that the proposed method can accurately detect most of the possible types of fall in video. Some successful and unsuccessful image frames of human fall detected by our approach are shown in Table 4.

Table 3. System performance of proposed fall model and aspect ratio approach

Video Content	Accuracy(%)	Specificity(%)	Sensitivity(%)
Indoor + Single	100 (95)	100 (97)	100 (90)
Outdoor + Single	100 (93)	100 (90)	100 (95)
Indoor + Multiple	74 (62)	84 (50)	62 (73)
Outdoor + Multiple	79 (64)	97 (85)	54 (37)
Omni-video + Single	94 (89)	96 (92)	90 (81)
Omni-video + Multiple	51 (40)	83 (49)	36 (29)

Table 4. Image frames of fall detected by our proposed approach



Our approach is able to achieve promising results only when there is a single person in the scene. For multiple people in the scene or in a crowd, this approach is not able to detect the fall accurately. For all video, we consider that first few frames contain only background scene.

6 Conclusion

We have presented a method for automatic detection of human fall in video. The proposed approach contains two main components, object detection and the use of a fall model. For object detection, we use an adaptive background subtraction method using a Gaussian Mixture Model in YCbCr color space. For the fall model, we extract a set of features such as aspect ratio, horizontal and vertical gradient values of an object as well as fall angle. In our experiments, we have taken three types of video clips (indoor, outdoor and omni-video) for both single and multiple people in the scene. Our experimental results show that the proposed method can accurately detect a single falling person. In future work, we plan to improve the fall model and apply it for multiple people in the scene.

References

1. Marquis-Faulkes, F., McKenna, S.J., Newell, A.F., Gregor, P.: Gathering the requirements for a fall monitor using drama and video with older people. *Technology and Disability* 17, 227–236 (2005)
2. Tao, J., Turjo, M., Wong, M., Wang, M., Tan, Y.: Fall incidents detection for intelligent video surveillance. In: *Fifth International Conference on Information, Communication and Signal Processing*, pp. 1590–1594 (2005)
3. Luo, S., Hu, Q.: A dynamic motion pattern analysis approach to fall detection. In: *IEEE International workshop in Biomedical Circuit and Systems* (2004)
4. Alwan, M., Rajendran, P.J.: A smart and passive floor-vibration based fall detector for elderly. *Information and Communication Technologies* 1, 1003–1007 (2006)
5. Ugur Toreyin, B., Dedeoglu, Y., Enis Cetin, A.: HMM based falling person detection using both audio and video. In: *Proc. IEEE Conf. Signal Processing and Communication Application (14th)* (2006)
6. Miaou, S., Sung, P., Huang, C.: A Customized Human Fall Detection System using omni-camera images and personal information. In: *Proc. of the 1st Distributed Diagnosis and Home Healthcare (D2H2) Conference, Arlington, Virginia, USA* (2006)
7. McKenna, S.J., Nait-Charif, H.: Summarising contextual activity and detecting unusual inactivity in a supportive home environment. *Pattern Analysis Application (14th)* 7, 386–401 (2005)
8. Rougier, C., Meunier, J.: Demo: Fall detection using 3D head trajectory extracted from a single camera video sequence. *Journal of Telemedicine and Telecare* 11(4) (2005)
9. Stauffer, C., Grimson, W.E.L.: Adaptive background mixture model for real time tracking. In: *Proc. IEEE Conf. Computer Vision and Pattern Recognition CVPR 1999*, pp. 246–252 (1999)