2021 43rd Annual International Conference of the
IEEE Engineering in Medicine & Biology Society (EMBC)
Oct 31 - Nov 4, 2021. Virtual Conference

Confidence-based Fall Detection Using Multiple Surveillance Cameras

Dara Ros and Rui Dai
Department of Electrical Engineering and Computer Science
University of Cincinnati, Cincinnati, OH-45220, USA
rosda@mail.uc.edu; rui.dai@uc.edu

Abstract— **The major cause of serious or even fatal injury for the elderly is a fall. Among various technologies developed for detecting falls, the camera-based approach provides a non-invasive and reliable solution for fall detection. This paper introduces a confidence-based fall detection system using multiple surveillance cameras. First, a model for predicting the confidence of fall detection on a single camera is constructed using a set of simple yet useful features. Then, the detection results from multiple cameras are fused based on their confidence levels. The proposed confidence prediction model can be easily implemented and integrated with single-camera fall detectors, and the proposed system improves the accuracy of fall detection through effective data fusion.**

I. INTRODUCTION

One of the most concerning injury for the elderly is falling that could happen when there is not anyone around. Many individuals aged 65 and over are less resilient and more vulnerable to accidental falls [1]. In most cases, an injured person has difficulty calling for help, or he/she could not get up by themselves and is exposed to a prolonged stay on the ground, which possibly results in serious or fatal injury. To protect the elderly against harmful falling events, fall detection solutions have been developed using wearable devices, ambient sensors, and video cameras. Although wearable devices with accelerometers are very capable of detecting inactivity and postural movement, it is onerous for the elderly to always wear a physical device [2]. Ambient sensors, such as audio and vibration sensors, have difficulty in differentiating human fall from an inanimate object fall due to their less visibility nature [2]. Unlike the other ambient sensors, video cameras provide rich and useful information for fall detection. Moreover, video cameras have already been deployed ubiquitously for surveillance purposes. Therefore, the vision-based approach is a promising non-invasive and low-cost solution for practical fall detection systems.

The video sequences are usually examined for body shape changes [3] to detect falls. In [3], the shape deformation of a person's silhouette is analyzed, and then falls are detected from normal activities using a Gaussian mixture model (GMM). A simple detection solution is designed based on human shape variation analysis and presenting a person's silhouette by only three points instead of an ellipse or a bounding box [4]. Another study in [5] analyzes body parts movements. It devises a detected bounding box into a ratio of 30:40:30. Falls are detected based on the vertical motion velocity, the ratio of head's center and its variance, and the body parts motion velocity.

Recently, Convolutional Neural Networks (CNN) have been applied in fall detection systems. The algorithm in [6] feeds RGB images to an optical flow image generator, based on which a CNN learns features and then classifies if a sequence of frames includes fall events. Similarly, the fall detection approach in [7] utilizes a combination of optical flow images obtained from two cameras and performs feature extraction and classification based on CNNs. A drawback of these algorithms is that the models may need to be re-trained if the number of cameras or the relative positions of cameras changes. The system, moreover, would require a big data set for training, which is challenging considering the limited number of public data sets.

The presence of multiple cameras in a surveillance system may capture the same human object from different views, providing more enriched observation. The information from multiple cameras could be jointly considered to improve detection performance. In [3], each camera has a vote resulting from the GMM classifier, and a simple majority vote method is used to make a final decision on all the cameras.

This paper studies how to improve the overall performance of a multi-camera system based on the fall detection results from individual cameras. The confidence of detection on each camera depends on specific scenes. From a particular camera view, one might look like in a standing posture while it possibly is a falling one on a different angle view. This study analyzes a set of features to determine the confidence level of each camera's detection result. These features are change rates of detected silhouette's ratio, orientation, centroid's height, optical x-axis projection, brightness, blind quality score, and silhouette's size. The detection results from multiple cameras are then fused taking into consideration their confidence levels.

Our major contribution in this work is an efficient model for predicting the confidence of fall detection on individual cameras in a system. Unlike other works that are designed on a specific number of cameras [6] [7] or for a particular single-camera detection algorithm [3] [4] [5], we provide a general solution that works for any number of cameras and with different single-camera fall detection algorithms. Besides, the proposed confidence prediction solution involves very low overhead, allowing the real-time fusion of detection results from multiple cameras.

978-1-7281-1178-0/21/$31.00 ©2021 IEEE

(a) Camera 1 (b) Camera 2 (c) Camera 3 (d) Camera 4 (e) Camera 5 (f) Camera 6 (g) Camera 7 (h) Camera 8

Fig. 1. Sample snapshots of video data set of scene 1.

II. CONFIDENCE-BASED FALL DETECTION

In this research, we study a public fall detection data set introduced by [8]. This data set consists of 24 scenarios of different activities and various ways of falling, and each scene is captured by eight inexpensive IP cameras from different views. The video sequences have an average length of 13 seconds with a frame rate of 120 frames per second, and a resolution of 720×480. It includes activities performed by one person comprise walking, housekeeping, sitting down, standing up, forward fall, backward fall, and loss of balance fall. For example, Fig.1 shows labeled fall frames from different views in Scene 1. In our study, the data set is fragmented into small window segments of sequential frames. A 15-frame window is investigated since it has been found in [8] that a fall event most likely happens within half a second.

We apply the open-source fall detection algorithm in [9] on the videos by each individual camera. We evaluate the sensitivity, specificity, and accuracy of the detection results on each camera, which are given by the following equations:

$$Sensitivity = \frac{TP}{TP+FN}; Specificity = \frac{TN}{TN+FP} \quad (1)$$

$$Accuracy = \frac{TN+TP}{TN+FP+TP+FN} \quad (2)$$

where TP, TN, FN, FP stand for the numbers of true positive, true negative, false negative, and false positive events.

The sensitivity indicates how well an algorithm detects a fall, and specificity shows performance on a no-fall detection. As accuracy is related to both sensitivity and specificity, we propose to evaluate the accuracy on a window of frames as the confidence of fall detection (CoF).

A. Confidence of Detection

We present a model for predicting the accuracy of fall detection or CoF on a single camera. This model utilizes a set of features that could be easily obtained through analysis of the video frames in a window. Two categories of features are extracted: features reflecting the overall quality of the video frames, and features that indicate the characteristics of a human object in the videos.

First, the overall quality of the video frames could have some effects on the accuracy of detecting an object, which is an important step in activity recognition. We propose two features in this category: the brightness level and the perceptual quality of a frame. The average brightness level is calculated for each frame, and then the average brightness of all the frames in the 15-frame window (*meanLuma*) is calculated as one feature for CoF prediction. Regarding the perceptual quality, a no-reference quality model is needed here since usually there are no reference images for quality evaluation in practical settings. The commonly used no-reference BRISQUE model [10] is applied to get a quality score on each frame in the window, and the average of the quality scores (*meanQs*) is calculated as another feature for CoF prediction.

The most challenging problem in detecting a fall is differentiating actual fall events from other daily activities that have similar characteristics to fall. In addition to video quality, the detection accuracy is also related to how a human object is captured by the surveillance cameras. The following features are extracted to describe the characteristics of a human object in a video. All of the features could be easily obtained during the execution of commonly-used fall detection algorithms.

- *Change rate of Silhouette's ratio*. A Silhouette's ratio is determined by a bounding box that encapsulates the detected object. As shown in Fig. 2, $r = a/b$ where a and b are the horizontal and vertical lengths of the bounding box, respectively. This ratio could indicate whether the object is laying, standing, or else. The change rate of the ratio δ_r implies a change in movement or body posture, given by
$$\delta_r = \frac{r(t) - r(t-tw)}{tw} \quad (3)$$
where t denotes time; tw is the length of time window. The values of δ_r for a scene are plotted in Fig. 3(a), where δ_r of no-fall frames are approximately zeros, whereas the one of fall frames are not.
- *Change rate of orientation*. Another property to consider is the object's orientation, as shown in Fig. 2. The change rate of orientation δ_{Ort} is given by
$$\delta_{Ort} = \frac{\theta(t) - \theta(t-tw)}{tw} \quad (4)$$
where $orientation$ represents angle between the ellipse's major axes and the x-axis; t denotes time; tw is the length of time window. Fig. 3(b) signals a difference between fall frames and no-fall frames in term of orientation change rate.
- *Average and standard deviation of a detected centroid's height ($meanCH$ and $stdCH$)*. From observation, the height of the detected centroid CH indicates a position of a person from the floor. For example, if a person lays on the ground, a centroid of the bounding box would be a lot smaller than when it stands. This property could be reversed if it is viewed from a different angle. As shown in Fig. 4 (a) and (b), both $meanCH$ and $stdCH$ of fall

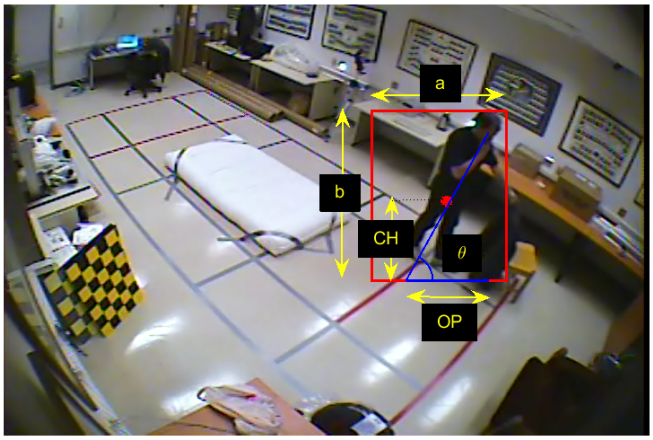

Fig. 2. Features Based on A Detected Object

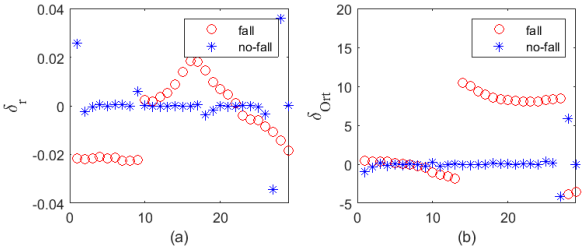

Fig. 3. Change rate of silhouette's ratio and orientation of Scene 1

frames are separated from the no-fall frames, and the values of $stdCH$ for fall-frames are consistently higher than that of no-fall-frames.

- *Average and standard deviation of the silhouette's projection on optical x-axis ($meanOP$ and $stdOP$).* These features can indicate whether or not it is easy to detect a person falling. Intuitively, a projection on the x-axis would rather big if the person is laying on the ground.
- *Average size of the detected silhouette* $meanSize$. This feature partially contributes to how well a posture detection is. As stated in [11], it would be hard to detect the object from the background if its size is either too small or too large. We calculate the average size of the detected silhouette in the time window.

After obtaining all the selected features, we use the Bagged tree of ensemble classifier to train a classification model [12]. On an independently drawn bootstrap copy of input data, every tree in the ensemble is grown. This model is to predict confidence of a fall detection based on the aforementioned 9 features from a 15-frame window. The tuning parameters of the ensemble are the maximum number of splits and the number of learners. The outputs of the prediction model (predicted CoF) are quantized into 11 levels or classes: 0.001, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, and 1, where 1 indicates the most confident in detection, and 0.001 serves the very less likely.

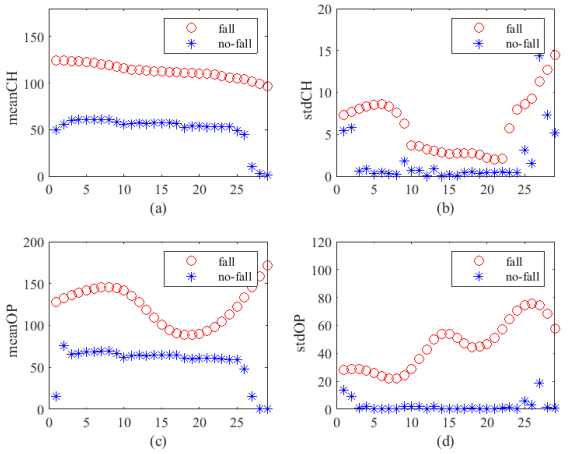

Fig. 4. Center of Height and Optical x-axis Projection of Scene 1

B. Confidence-based Data Fusion

Once the confidence of detection CoF is determined on each camera, we combine them by associating CoF with x, the detection result of the camera. We then fuse the detection results using the following equation:

$$CbFD = \sum(CoF * x) \tag{5}$$

where x takes values of 1 (fall) and -1 (no fall). Finally, the system output is fall if $CbFD > 0$ and no fall if $CbFD \leq 0$.

III. PERFORMANCE EVALUATION

To evaluate the performance of the proposed confidence prediction model, the entire data set is divided into a training set and a testing set, which are listed in Table I. Each video scene comprises 8 camera views from different angles. The training set includes 18 scenes that are randomly selected from the whole data set, and the remaining scenes are for testing. For training, 5-fold cross validation, 30 learners, and a maximum splits of 119854 are set to build the ensemble trees.

The classification performance of the proposed confidence prediction model is presented in Fig. 5. On a large number of observations (119855 data points), the proposed features are calculated over a time window of 15 frames per observation. The majority of data points are in either class 0.001 or class 1, and most of them are correctly classified. The classification performance for in-between classes (class 0.2 - 0.8) is fair, but there are only 2.5% data points from these classes. The

TABLE I
LEARNING SETTING

Category	Video Name	Data Points	Percentage
Training set	chute02, chute03, chute04, chute05, chute07, chute08, chute10, chute11, chute12, chute13, chute14, chute16, chute17, chute18, chute19, chute20, chute21, chute22	119855	78.26%
Testing set	chute01, chute06, chute09, chute15, chute23, chute24	93673	21.74%

overall accuracy of classification on the entire training set is 95.3%.

Among 24-scene of the experimental data set, there are 25 fall events and 49 no-fall events, and a fall event has 40 frames on average. For example, in scene 1, a fall event starts from 1080th to 1108th frame. If a fall is detected within the fall event window, a TP is incremented toward the system evaluation. Otherwise, it is considered for a FP.

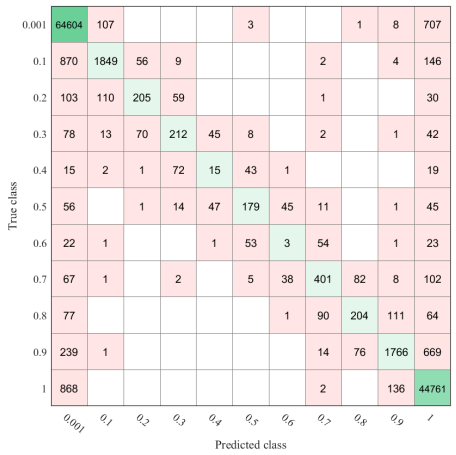

Fig. 5. Confusion matrix of Confidence Prediction model

Next, the overall performance of the confidence-based fusion approach is evaluated on the entire data set. The fall detection algorithm for single cameras in [9] is implemented. Table II describes a comparison of performance in terms of accuracy, sensitivity, and specificity, and Table III depicts the confusion matrices of our proposed algorithm and the majority vote approach in [3]. Our proposed model improves the system's performance in all terms. As can be observed in Table II, the proposed algorithm greatly increases the sensitivity in comparison to the majority vote algorithm and the average performance on single camera based detection.

During the experiment, we notice that there are some frames where the detector does not detect a whole silhouette of the object of interest. Moreover, there are some instances where the person moves close to a still-object, and both areas are detected as one object. These result in some incorrect feature values. Another issue, that causes a lower rate of sensitivity, is related to how the fall frames are labeled. These observations suggest some rooms for improvement of the proposed work: a better object detection algorithm could help improve our prediction model by better differentiating human objects from other objects.

TABLE II

FALL DETECTION PERFORMANCE OF A SYSTEM INTEGRATED WITH [9]

Algorithm	**Accuracy**	***Sensitivity***	***Specificity***
CbFD (Proposed)	72.00%	30.56%	64.10%
Majority Vote [3]	66.67%	0.00 %	66.67%
Average Single-Camera [9]	44.00%	16.00%	58.00%

TABLE III

DETECTION PERFORMANCE COMPARISON

		CbFD Detected		Majority Vote Detected	
		Fall	***no-Fall***	***Fall***	***no-Fall***
Labelled	***Fall***	11	14	0	25
	no-Fall	24	25	0	49

IV. CONCLUSION

In this paper, a confidence-based fall detection solution for multiple-camera systems has been proposed. The solution integrates a new model for predicting the confidence of fall detection on each camera using easily obtained features. The detection results from multiple cameras are then fused based on their confidence levels. The proposed solution achieves better performance than the majority vote algorithm and single camera detection. With low-complexity, the proposed method can be adapted to a system with any number of cameras and different single-camera detection algorithms. In the future, we hope to verify our solution using more data in practical surveillance scenarios.

REFERENCES

[1] Briana Moreland, Ramakrishna Kakara, and Ankita Henry. Trends in nonfatal falls and fall-related injuries among adults aged $\geq$ 65 years—united states, 2012–2018. *Morbidity and Mortality Weekly Report*, 69(27):875, 2020.

[2] Muhammad Mubashir, Ling Shao, and Luke Seed. A survey on fall detection: Principles and approaches. *Neurocomputing*, 100:144–152, 2013.

[3] Caroline Rougier, Jean Meunier, Alain St-Arnaud, and Jacqueline Rousseau. Robust video surveillance for fall detection based on human shape deformation. *IEEE Transactions on circuits and systems for video Technology*, 21(5):611–622, 2011.

[4] Jia-Luen Chua, Yoong Choon Chang, and Wee Keong Lim. A simple vision-based fall detection technique for indoor video surveillance. *Signal, Image and Video Processing*, 9(3):623–633, 2015.

[5] Chadia Khraief, Faouzi Benzarti, and Hamid Amiri. Vision-based fall detection for elderly people using body parts movement and shape analysis. In *Eleventh International Conference on Machine Vision (ICMV 2018)*, volume 11041, page 110410K. International Society for Optics and Photonics, 2019.

[6] Adrián Núñez-Marcos, Gorka Azkune, and Ignacio Arganda-Carreras. Vision-based fall detection with convolutional neural networks. *Wireless communications and mobile computing*, 2017, 2017.

[7] Ricardo Espinosa, Hiram Ponce, Sebastián Gutiérrez, Lourdes Martínez-Villaseñor, Jorge Brieva, and Ernesto Moya-Albor. A vision-based approach for fall detection using multiple cameras and convolutional neural networks: A case study using the up-fall detection dataset. *Computers in biology and medicine*, 115:103520, 2019.

[8] Edouard Auvinet, Caroline Rougier, Jean Meunier, Alain St-Arnaud, and Jacqueline Rousseau. Multiple cameras fall dataset. *DIRO-Université de Montréal, Tech. Rep*, 1350, 2010.

[9] Caroline Rougier, Jean Meunier, Alain St-Arnaud, and Jacqueline Rousseau. Fall detection from human shape and motion history using video surveillance. In *21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07)*, volume 2, pages 875–880. IEEE, 2007.

[10] Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on image processing*, 21(12):4695–4708, 2012.

[11] Lingchao Kong, Ademola Ikusan, Rui Dai, and Jingyi Zhu. Blind image quality prediction for object detection. In *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 216–221. IEEE, 2019.

[12] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.