Fall Detection using Human Skeleton Features

Heilym Ramirez[1], Sergio A. Velastin[2,3], Ernesto Fabregas[4], Ignacio Meza[1], Dimitrios Makris[5], and Gonzalo Farias[1]

[1]Escuela de Ingeniería Eléctrica, Pontificia Universidad Católica de Valparaíso, Av. Brasil 2147. 2362804 Valparaíso, Chile.
[2]School of Electronic Engineering and Computer Science, Queen Mary University of London, London E1 4NS, UK
[3]Department of Computer Science and Engineering, Universidad Carlos III de Madrid, 28911 Leganés, Madrid, Spain
[4]Departamento de Informática y Automática, Universidad Nacional de Educación a Distancia, Juan del Rosal 16. 28040 Madrid, Spain.
[5]Faculty of Science, Engineering and Computing, Kingston University, London SW15 3DW, UK.

Keywords: fall detection, human skeleton, pose estimation, computer vision.

Abstract

A leading cause of death and serious injury in people, especially for the older people, are falls. In addition, fall accidents have a direct economic cost to healthcare systems and have an indirect impact, to the society's productivity. Among the most significant problems in fall detection systems is privacy, limitations of operating devices, and the comparison of machine learning techniques for detection. This article presents a system of fall detection by means of a k-Nearest Neighbor (KNN) classifier based on camera-vision using pose detection of the human skeleton for the features extraction. The proposed method is evaluated with UP-FALL dataset, surpassing the results of other fall detection systems that use the same database. This method achieves a 98.84% accuracy and an F_1-Score of 97.41%.

1 Introduction

The World Health Organization (WHO) suggests that each year, approximately 37.3 million falls are serious [1] sufficient to require immediate medical attention and that one of the leading causes of fatal lesions is falls, making falls a major global public health.

Although the older people are most at risk of serious and fatal injuries by falls, children and infants are also a group at high-risk when they suffer falls injuries [1]. Therefore, detect falls has become a field of interest for many researchers. Falls involve a significant direct economic cost to healthcare systems, in both terms of hospital and in long-term care costs, besides, of the resulting indirect costs [2]. This field of research has given rise to the intelligence environments concept that help to create assistance and monitoring systems in relevant environments.

Wearable sensors have been the main focuses in systems for fall detection, environmental sensors or vision devices [3]. Context-sensitive systems include all the systems that use sensor technology implemented on the environment, such as cameras, motion capture devices and Kinect, which have the advantage of not being invasive devices allowing a more real context, taking into account that it is easier to implement a system with cameras to identify falls in a certain population, than putting a sensor or body device on each individual. In addition, the use of cameras is generally cheaper.

The important challenges and issues identified by most authors include the concerns about privacy, intrusion and operating device limitations, as well as the difficulties in comparing between the techniques.

Nowadays, the development and studies of computer vision focused on detect falls have been a popular research topic [4–8] where issues such as accuracy and decrease have been addressed and with computational complexity as one of the main challenges. Vision-based systems use image processing techniques over video frames or captured images from cameras. Machine learning (ML) models can apply on top of image processing techniques to permit a more precise fall detection. Fall detection systems use the best recognized techniques of supervised learning, these are: MLP (multilayer perceptron) [9], SVM (support vector machines) [10], HMM (hidden Markov models), decision trees, KNN (random forest, k-Nearest Neighbors) [11] and CNN (Convolutional Neural Networks) [12]. Zerrouki et al. [13] present a study comparing ML-models for the falls detection, selecting as input video sequences during different daily falls and activities. Yanfei et al. [14] study the signals from a Kinect camera and processes point cloud images to detect drops and reduce false positives. Recently, fall detection applying deep learning techniques for has become an active area of research. Lu et al. [7] aplica CNN y LSTM para la extracción de características utilizando secuencias de video de datos ambientales. In [15], CNN (convolutional neural networks) are trained on different sets of optical flow image data which helps the network detect different actions.

This article shows a method that can detect falls by only using images from a standard video camera without the need to use environmental or depth sensors, that significantly outperforms the results obtained with the best ML model (KNN) in [16]. Fall detection is carried out using pose estimation of the human skeleton for feature extraction. The main contributions of this method is that the detection of the human skeleton can be used by any suitable pose estimation algorithm. The proposed approach has been validated with the multi-modal public data set presented in [16]. A machine learning algorithm has been tested. The results exceed those obtained by other authors with the same data set by a significant margin [16, 17]. The article is organized as follows. In section 2, a description of the UP-FALL fall detection data set is presented. Section 3 details the proposed fall detection approach and explains AlphaPose, the human skeleton proposes detection as a feature extraction method to recognize activities. Section 4 shows the experimental results and a comparison with the previous results on the same data set. And then Section 5 summarizes the conclusions and the future research work.

2 Dataset

Work presented here uses UP-FALL as an experimental dataset to evaluate results and comparing it to the original work [16] that also uses the same data set.

The dataset included 12 activities, which were performed by 17 healthy young human subjects. Subjects performed 5 different types of falls (forward falls using the hands, forward falls using the knees, backward falls, falls sitting on an empty chair, and falls sideways) and 7 daily human activities (walk, stand, picking up an object, sit, jump, lay and kneel). Table 1 shows the 12 activities.

Activity ID	Description
1	Forward falls using the hands
2	Forward falls using the knees
3	Backward falls
4	Sideways falls
5	Sitting falls on an empty chair
6	Walk
7	Stand
8	Sit
9	Picking up an object
10	Jum
11	Lay down
20	Unknown activity

Table 1. UP-FALL Activities.

The data distribution based on daily activities vs. falls are 77.55% and 22.45%, respectively. The instances of non-fall activities outnumber falls so as to make it more representative of the sporadic nature of fall events. The dataset is multi modal with data captured using five Mbientlab Meta Sensor wearable sensors (IMU), one electroencephalograph (EEG) NeuroSky MindWave headset and six infrared sensors (IR). It also includes data obtained from two Microsoft LifeCam Cinema cameras (CAM) placed at 1.82 m above the ground, one for a side view and the other for a front view.

One of the challenges presented by UP-FALL is that of fall detection as a binary classification problem, i.e. to distinguish between a fall (any of classes 1 to 5) and a non-fall (any of the remaining classes).

The work in [16] evaluates four different machine learning (ML) methods for the fall detection and activity recognition problems: Random Forest (RF), Support Vector Machine (SVM), Multi-Layer Perceptron (MLP) and k-Nearest Neighbors (kNN). KNN model delivered the best performance using video-only (CAM) data with a F_1-Score of 15.19%. Nevertheless, as shown in Table 3, it gave poor results. These were improved significantly when using a CNN (convolutional neural network), getting an F_1-Score of 71.20%.

3 Method

This work focuses on improving the performance of fall detection using only the video data, as in practical applications such as assisted living and public space monitoring, use wearable and other sensor modalities is not realistic. The main hypothesis is that the originally poor results can be improved significantly by using articulated bodies (skeletons) extracted from the video, even when using the same ML methods used in [16]. So, the aim is to implement a fall detection method using information from camera 1 in the dataset, and compare its performance to that using KNN on optical flow, as initially reported.

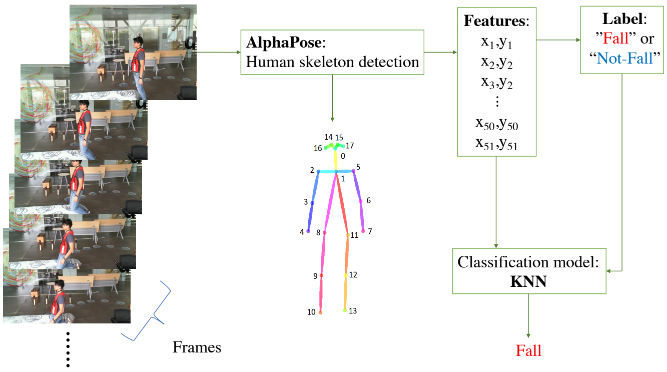


Figure 1. Workflow for fall detection.

The method that was developed for this study is illustrated in Fig. 1. Consists of collecting data, feature extraction using human skeleton estimation, skeleton filtering, and at last a classification model for fall recognition. All the steps have been implemented using Python 3.6.

3.1 Feature extraction and selection

Images are located at `https://sites.google.com/up.edu.mx/har-up/`. The files are organized in 17 folders, one for each subject. Within each folder, there are 11 subfolders, one for each activity. Within these subfolders, there are three other subfolders, one for each trial. In every subfolder, there is a CSV file that points to a ZIP-file with the images recorded on camera 1.

3.1.1 Human skeleton detection

By using AlphaPose [18], human pose detection is possible, obtaining 17 keypoints or joints, with coordinates (x,y), which when joined form a skeleton with the human's pose, also indicating the score of the detection for each keypoint. With these 51 (17*(2+1)) attributes, the characteristics are obtained to train a classifier to detect falls.

AlphaPose is an open source method that allows an accurate multi-person pose estimator [18], available in `https://www.mvig.org/research/alphapose.html`. It uses RGB images as input, then performs the pose detection with a pre-trained model (COCO dataset), outputting a JSON-file with the location (x,y) of 17 keypoints, which together form a skeleton with the pose(s) of one or more people.

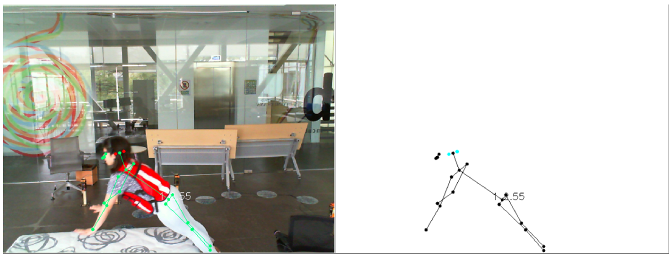


Figure 2. Falling woman skeleton detection using AlphaPose.

The process to generate the feature extraction begins by defining all the image sequences from UP-FALL. These images are then processed with AlphaPose to generate a dataset of frames of skeleton joints coordinates which can be visualised in images such as the one shown in Figure 2.

3.1.2 Skeletons filtering

After converting RGB images to skeletons, a clean up process is needed. First, the detection of more than one skeleton (person) in the same frame. Second, the images from the UP-FALL database supplied by camera 1 are used, because the detection of the fall is more accurate with a side view vs. a front view. Finally, is important noted that some frames in the database do not contain people, so these images are eliminated to generate greater consistency between the skeletonization and the labels.

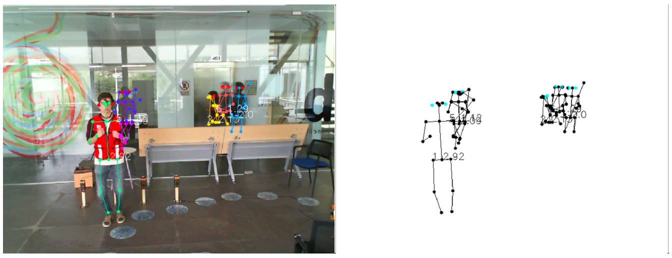


Figure 3. Image with more than one skeleton.

Detection of more than one skeleton for the same frame presents a problem for the system, generating an incorrect classification. Figure 3 shows an example of the detection of several skeletons in one image. It is observed that only one skeleton is associated with the label, while the other skeletons are not part of the action (label).

Different metrics were designed to find the correct skeleton associated with the label and the AlphaPose skeletonization confidence score was identified as the best indicator. Finally, a filter was designed that selects the skeleton with the highest score in each frame eliminating the rest of the skeletons in such frames.

3.2 Classification model

This work seeks to improve upon the results of the best ML-model (KNN) in [16] with the CAM modality. Therefore, k-Nearest Neighbors (KNN) has been chosen for direct comparison. This is a method based on instances that compares an input with the training points of the k nearest neighbors and determines the output response which is based on the most frequent type observed in the k neighbors.

Model	Parameters

KNN	neighbors = 5
	leaf size = 30
	metric = Euclidean

Table 2. Parameter settings for KNN model.

Table 2 summarizes the parameter settings for the classifier model. These are the same as those used in [16].

4 Evaluation

The fall detection method is evaluated by means of the classifier model with the features extracted from the 2D skeleton coordinates, obtained with AlphaPose, for each frame of the dataset processed independently. Experiments were performed using 70% (154,462 samples) of the training dataset was used for performance the experiment and the remaining 30% (66,198 samples) for testing. The dataset contains 220,660 images of which 49,544 correspond to falls and 171,116 to non-falls.

For the detection of falls, a binary classifier system based on the twelve UP-FALL activities is implemented, for which a re-labeling process is carried out. The 5 falls activities (forward falls using the hands, forward falls using the knees, backward falls, falls sitting on an empty chair, and falls sideways) are labeled ”Fall” and the and 7 daily human activities (walk, stand, picking up an object, sit, jump, lay and knees) are labeled ”Not Fall”.

Using the same experimental methodology described in [16], performance evaluation is carried out using ten rounds (k-fold = 10) of cross-validation using random 90:10 partitions of the whole dataset for each classification method. In UP-FALL, actors involved in training are the same actors for testing but their training and test actions are not mixed.

4.1 Evaluation metrics

For a direct comparison, this work uses the same performance metrics used in [16]: Accuracy, precision, sensitivity, specificity and F_1-Score. Where:

- True positives (TP): ”Fall” detected as ”Fall”.

- False positives (FP): ”Not Fall” detected as ”Fall”.
- True negatives (TN): ”Not Fall” detected as ”Not Fall”.
- False negatives (FN): ”Fall” detected as ”Not Fall”.

To calculate the accuracy, divide the average number of TP and TN by the total number of cases examined (Equation 1).

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{1}$$

As per Equation (2) precision, is the average number of the number of TP divided by the sum of TP and FP. Recall is the average number of TP across all activities and falls divided by the sum of TP and FN (Equation (3)).

$$precision = \frac{TP}{TP + FP} \tag{2}$$

$$recall = \frac{TP}{TP + FN} \tag{3}$$

To calculate the Specificity, divide the average number of TN by the sum of TN and FP, as Equation (4) shows.

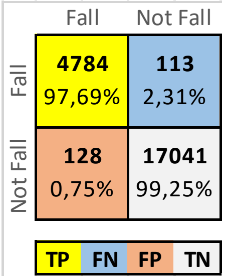
$$specificity = \frac{TN}{TN + FP} \tag{4}$$

Finally, F_1-Score is calculated as shown in Equation (5), and is used to evaluate the proposed model, as a single figure of merit that considers both precision and recall.

$$F_1 - Score = 2 \times \frac{precision \times recall}{precision + recall} \tag{5}$$

4.2 Results

The presented approach successfully recognized falls using a KNN classifier with skeleton human poses. The classifier delivered a high accuracy of 98.84%, a precision and recall of 97.53% and 97.30% respectively, a specificity of 99.29% and an F_1-Score of 97.41%.



	Fall	Not Fall
Fall	4784 97,69%	113 2,31%
Not Fall	128 0,75%	17041 99,25%

Figure 4. Confusion matrix for fall detection method.

Figure 4 shows the best confusion matrix from cross-validation for KNN classifier based on accuracy. It is observed in the confusion matrix of model that of all the fall data only 2.31% data are not recognized as falls.

Therefore, it can be concluded that the original hypothesis is demonstrated, namely that it is possible to detect falls with the k-nearest neighbor classifier proposed in [16] when using only a single modality (vision from a camera) through the use of human skeleton pose estimate features, obtained via deep neural models, which considerably improves the model performance.

Model	**KNN** in [16]	**CNN** in [16]	**KNN** in [17]	**CNN** in [17]	**This** **Work**
Accuracy	34.03	95.10	27.30	82.26	**98.84**
Precision	15.32	71.80	16.32	74.25	**97.53**
Recall	15.54	71.30	14.35	71.67	**97.30**
Specificity	93.09	**99.50**	90.96	77.48	99.29
F_1-Score	15.19	71.20	15.27	72.94	**97.41**

Table 3. Comparison between our proposal and the best models of other camera vision based fall detection systems, that use UP-FALL dataset.

Table 3 compares the models with the best performance of camera vision-based systems for fall detection that use the UP-FALL dataset.

The method proposed here achieves the detection of falls using only one camera, unlike other works that use eight cameras for vision-based systems for fall detection, as from [19] and [15], which makes our method simpler and cheaper to implement. Also, the network structure in this method (Figure 1) is very simple compared to other methods. As an example, in [19] PCA is employed for feature extraction and SVM is employed for classification and in [15] present a VGG-16 architecture modified to receive inputs and CNN for classification. On the other hand, Espinosa et al. [17] use a multi-camera approach using UP-FALL dataset, which gets results similar to [19] and [15] using only two cameras. As Table 3 shows, they gets their best performance with a CNN.

In Table 3 it can be seen that the proposed human skeleton features method meets the main objective of this work, exceeding by more than 80% the performance of the models (RF, SVM, MLP and KNN) given in [16] with its windowing method. It is shown that the use of human detection features, in addition to making possible the detection of falls with a high performance, makes our KNN model overcome the best falls detection models based on camera vision with UP-FALL reported in the state-of-the-art.

5 Conclusion

This paper presents a system for fall detection based on camera vision with a KNN classification model. A method for features extraction with pose estimation based on human skeletonization is proposed. The method is evaluated with the UP-FALL pubklic access dataset and uses the KNN model in [16] for direct comparison. The method outperforms [16] and other fall detection systems that use the same dataset.

The proposed method demonstrated good results using human skeletonization for features extraction, despite the detection of more than one skeleton in a single frame. It is possible that other algorithms use, like an LSTM, could eliminate the problem of confusion when carrying out an analysis over time, managing to identify the skeleton of interest based on a frames sequence.

For future work and as part of our ongoing project, a system of fall detection and activity recognition is being developed, pairing four classifier models (RF, SVM, MLP and KNN).

Acknowledgements

This work was supported by the Chilean Research and Development Agency (ANID) under the Project FONDECYT 1191188.

References

[1] W. H. Organization, W. H. O. Ageing, and L. C. Unit, *WHO global report on falls prevention in older age*. World Health Organization, 2008.

[2] S. R. de Assis Neto, G. L. Santos, E. da Silva Rocha, M. Bendechache, P. Rosati, T. Lynn, and P. T. Endo, "Detecting human activities based on a multimodal sensor data set using a bidirectional long short-term memory model: A case study," in *Challenges and Trends in Multimodal Fall Detection for Healthcare*, pp. 31–51, Springer, 2020.

[3] M. Mubashir, L. Shao, and L. Seed, "A survey on fall detection: Principles and approaches," *Neurocomputing*, vol. 100, pp. 144–152, 2013.

[4] K. Adhikari, H. Bouchachia, and H. Nait-Charif, "Activity recognition for indoor fall detection using convolutional neural network," in *2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA)*, pp. 81–84, IEEE, 2017.

[5] C.-Y. Lin, S.-M. Wang, J.-W. Hong, L.-W. Kang, and C.-L. Huang, "Vision-based fall detection through shape features," in *2016 IEEE Second International Conference on Multimedia Big Data (BigMM)*, pp. 237–240, IEEE, 2016.

[6] H. Liu and Y. Guo, "A vision-based fall detection algorithm of human in indoor environment," in *Second International Conference on Photonics and Optical Engineering*, vol. 10256, p. 1025644, International Society for Optics and Photonics, 2017.

[7] N. Lu, Y. Wu, L. Feng, and J. Song, "Deep learning for fall detection: Three-dimensional cnn combined with lstm on video kinematic data," *IEEE journal of biomedical and health informatics*, vol. 23, no. 1, pp. 314–323, 2018.

[8] Y. Kong, J. Huang, S. Huang, Z. Wei, and S. Wang, "Learning spatiotemporal representations for human fall detection in surveillance video," *Journal of Visual Communication and Image Representation*, vol. 59, pp. 215–230, 2019.

[9] A. Jalal, S. Kamal, and D. Kim, "A depth video sensor-based life-logging human activity recognition system for elderly care in smart indoor environments," *Sensors*, vol. 14, no. 7, pp. 11735–11759, 2014.

[10] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "Training computationally efficient smartphone–based human activity recognition models," in *International Conference on Artificial Neural Networks*, pp. 426–433, Springer, 2013.

[11] L. C. Jatoba, U. Grossmann, C. Kunze, J. Ottenbacher, and W. Stork, "Context-aware mobile health monitoring: Evaluation of different pattern recognition methods for classification of physical activity," in *2008 30th annual international conference of the ieee engineering in medicine and biology society*, pp. 5250–5253, IEEE, 2008.

[12] S. Münzner, P. Schmidt, A. Reiss, M. Hanselmann, R. Stiefelhagen, and R. Dürichen, "Cnn-based sensor fusion techniques for multimodal human activity recognition," in *Proceedings of the 2017 ACM International Symposium on Wearable Computers*, pp. 158–165, 2017.

[13] N. Zerrouki, F. Harrou, A. Houacine, and Y. Sun, "Fall detection using supervised machine learning algorithms: A comparative study," in *2016 8th International Conference on Modelling, Identification and Control (ICMIC)*, pp. 665–670, IEEE, 2016.

[14] Y. Peng, J. Peng, J. Li, P. Yan, and B. Hu, "Design and development of the fall detection system based on point cloud," *Procedia computer science*, vol. 147, pp. 271–275, 2019.

[15] A. Núñez-Marcos, G. Azkune, and I. Arganda-Carreras, "Vision-based fall detection with convolutional neural networks," *Wireless communications and mobile computing*, vol. 2017, 2017.

[16] L. Martínez-Villaseñor, H. Ponce, J. Brieva, E. Moya-Albor, J. Núñez-Martínez, and C. Peñafort-Asturiano, "Up-fall detection dataset: A multimodal approach," *Sensors*, vol. 19, no. 9, p. 1988, 2019.

[17] R. Espinosa, H. Ponce, S. Gutiérrez, L. Martínez-Villaseñor, J. Brieva, and E. Moya-Albor, "Application of convolutional neural networks for fall detection using multiple cameras," in *Challenges and Trends in Multimodal Fall Detection for Healthcare*, pp. 97–120, Springer, 2020.

[18] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, "Rmpe: Regional multi-person pose estimation," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2334–2343, 2017.

[19] S. Wang, L. Chen, Z. Zhou, X. Sun, and J. Dong, "Human fall detection in surveillance video based on pcanet,"

Multimedia tools and applications, vol. 75, no. 19, pp. 11603–11613, 2016.