Eigenwalk: a Novel Feature for Walk Classification and Fall Prediction

Masoud Hemmatpour, Renato Ferrero, Bartolomeo Montrucchio, Maurizio Rebaudengo
Dipartimento di Automatica e Informatica, Politecnico di Torino
{ masoud.hemmatpour, renato.ferrero, bartolomeo.montrucchio, maurizio.rebaudengo } @polito.it

ABSTRACT

Predicting a fall is one of the most promising approach to avoid it. Different studies strive to classify abnormal and normal walks in order to predict a fall before its occurrence. This study introduces *eigenwalk*, a novel feature based on the principal components of the accelerometer and gyroscope signals. This feature, in conjunction with a random forest classifier, is able to distinguish walk patterns and to estimate a fall risk. As the accelerometer and the gyroscope embedded in a smartphone are recognized to be precise enough for fall avoidance systems, they have been exploited in an experimental analysis in order to compare the proposed approach with the most recent ones. The results have shown that the new feature in combination with the random forest classification outperforms state-of-the-art approaches, by improving the accuracy up to 98.6%.

1. INTRODUCTION

Falls are a serious problem in both medical and social aspects. Unintentional falls are one of the causes of hospitalization that mostly increases health service costs [1]. Falls are particularly critical among elderly people. So, international organizations such as the United Nations advises improving health care systems and creating a supportive environment in order to decrease and control falls in the elderly community. Consequently, different fall avoidance systems have been developed. Generally speaking, fall avoidance systems can be categorized into two different types: fall prediction and detection systems. Fall prediction systems (FPSs) recognize abnormal walks and predict a fall occurrence [2]. Fall detection systems only notify an acquaintance of the user in case of fall occurrence [3, 4, 5, 6, 7]. Since the latter systems do not avoid a fall, they are less effective than FPSs. FPSs strive to avoid a fall with common sensors, such as accelerometer (for measuring the acceleration, i.e, rate of change of the velocity of an object) and gyroscope (for measuring the angular rate around one or more axes of the space). FPSs usually follow a unique procedure: firstly, data are collected from sensors, then, data are analyzed to compute an appropriate feature set. Finally, a classification algorithm recognizes abnormal walks based on the feature set. Most of FPSs usually analyze user's posture or gait variables to investigate the characteristics of the movement. This study investigates FPS considering in particular data acquired with gyroscope and accelerometer. In this study, a new feature, called *eigenwalk*, is proposed in combination with a random forest classifier. The eigenwalk feature is based on the analysis of the principal component of the data from accelerometer and gyroscope. Random forest is a classification method that creates a multitude of decision trees at training time, and then it makes predictions according to the majority rule [8].

The remainder of the paper is organized as follows. Relevant work in literature is reviewed in Section 2. The proposed feature is described in Section 3. Exploited classification methods in FPS are explained in Section 4. Evaluation criteria are explained in Section 5. Then, the experimental setup and the results are presented in Section 6. Finally, some conclusions are written in Section 7.

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

2. RELATED WORK

Several types of FPS have been developed over the past decade. FPSs usually regard tilt and acceleration changes, which are obtained by means of gyroscope and accelerometer.

When a user significantly tilts in a direction, he/she assumes an abnormal posture, which can lead to a fall. So, the user tilt can be a factor to assess the risk of a fall. User tilt can be estimated with the combination of gyroscope and accelerometer [9]. Data collected from the accelerometer and gyroscope of a smartphone are processed to classify the user's gait pattern. If the system monitors an abnormal gait pattern, it warns the user of a potential fall. The user's tilt is computed to predict the fall, as described in the following. In the formulas, $A(t)$ indicates the acceleration and $A_x(t)$, $A_y(t)$, $A_z(t)$ refer to its components along the 3 axes. Furthermore, A_{xi}, A_{yi} and A_{zi} are the discrete i-th acceleration samples in the 3 axes. Similarly, $G(t)$ indicates the gyroscope measure and $G_p(t)$, $G_r(t)$, $G_y(t)$ refer to the measure on pitch, roll, and yaw.

First, the average value of N instances in the acceleration vector is computed as:

$$\overrightarrow{B}(t) = \frac{1}{N}\Sigma_{t=1}^{N}\overrightarrow{A}_0(t) . \quad (1)$$

The tilt angles are defined as:

$$\theta_1 = \arctan\left(\frac{B_y}{B_z}\right) \quad (2)$$

$$\theta_2 = \arctan\left(\frac{B_x}{B_y\sin(\theta_1) + B_z\cos(\theta_1)}\right) . \quad (3)$$

The tilt acceleration is computed as follows:

$$\overrightarrow{A}_1(t) = \begin{bmatrix} \cos(\theta_2) & \sin(\theta_1)\sin(\theta_2) & -\cos(\theta_1)\sin(\theta_2) \\ 0 & \cos(\theta_1) & -\sin(\theta_1) \\ \sin(\theta_2) & \sin(\theta_1)\cos(\theta_2) & \cos(\theta_1)\cos(\theta_2) \end{bmatrix} \times \overrightarrow{A}_0(t)$$

BODYNETS 2016, December 15-16, Turin, Italy

DOI 10.4108/eai.15-12-2016.2267645

The tilt gyroscope $\overrightarrow{G}_1(t)$ is computed similarly to $\overrightarrow{A}_1(t)$. Subsequently, the gravity vector is removed from the accelerometer data:

$$\overrightarrow{A}_2(t) = \left[\overrightarrow{A}_1x(t), \overrightarrow{A}_1y(t), \overrightarrow{A}_1z(t) - \frac{1}{N}\Sigma_{t=1}^{N}\overrightarrow{A}_1z(t)\right] \quad (4)$$

$$\overrightarrow{A_h}(t) = \sqrt{A_{2x}(t)^2 + A_{2y}(t)^2} \quad (5)$$

$$\overrightarrow{G_t}(t) = \sqrt{G_{1p}(t)^2 + G_{1r}(t)^2} \quad (6)$$

Finally, a general tilt vector is created by computing the magnitude of the horizontal acceleration and combining it with the measurement of the gyroscope. Afterwards, *Hjorth parameters* and *energy* are computed on the tilt vector.

Hjorth parameters are statistical measures of the signal, based on its variance $var(A(t))$ in the time domain [10]:

- $Hjorth\ activity = var(A(t))$ indicates the signal power
- $Hjorth\ mobility = \sqrt{\frac{var(A'(t))}{var(A(t))}}$ shows the smoothness of the signal curve
- $Hjorth\ complexity = \frac{mobility(A'(t))}{mobility(A(t))}$ measures the irregularities in the frequency domain.

The $energy$ of the acceleration signal specifies the amount of activity in the vertical and horizontal directions. It estimates the strength of the contact with the floor, so it can be used to recognize abnormal walk pattern such as stumbling. The energy of the signal is computed as follows:

$$E_x = \int_{-\infty}^{\infty} |A(t)|^2 dt, \quad (7)$$

Other acceleration-derived parameters that can be good fall indicator are the *signal magnitude area* (SMA), *signal magnitude vector* (SMV), *derivative of the acceleration* and *peak* analysis [11].

SMA is used to classify the user activities. It is computed as:

$$SMA = \frac{1}{T}\left(\int_0^T |A_x(t)|dt + \int_0^T |A_y(t)|dt + \int_0^T |A_z(t)|dt\right) \quad (8)$$

where T is the monitored interval.

SMV specifies the degree of the movement intensity and the resultant of the acceleration signal:

$$SMV = \frac{1}{n}\Sigma_{i=1}^{n}\sqrt{A_{xi}^2 + A_{yi}^2 + A_{zi}^2} \quad (9)$$

where n is the number of samples.

The derivative $A'(t)$ of the acceleration indicates the vibration of the movement.

The peak and the peak-to-peak are simple and useful measurements of data changing over time. The peak is the maximum value of the signal over the period of time, and the peak-to-peak is the difference between minimum and maximum value of the signal.

Another features used in FPS relies on the relative frequency in the data distribution [12]. Firstly, the collected data are ordered in different intervals from the minimum to the maximum value. Then, a histogram based on the repetition of the values is computed as shown in Algorithm 1, where n specifies the number of intervals, Y is the set of input values and X is the sequence of endpoints of the intervals. After computing the relative frequency, the standard deviation of the relative frequency distribution is exploited to classify normal and abnormal walks.

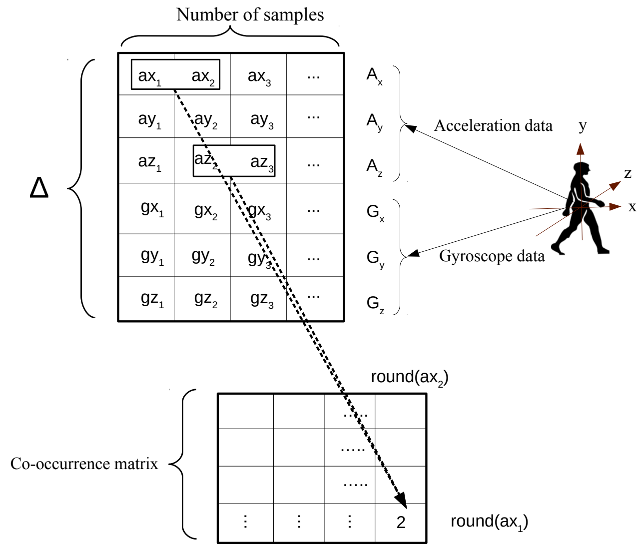

Figure 1: Computation of the co-occurrence matrix.

Algorithm 1: Computation of relative frequency

```
numIntervals = n;
intervalWidth = (max(Y) − min(Y)) / numIntervals;
x0 = min(Y);
for (i = 1; i < numIntervals; i ++)
    xi = xi−1 + intervalWidth;
ncount = histc(Y, X);
relativefreq = ncount / length(Y);
```

3. EIGENWALK

The principal component analysis is a mathematical procedure based on an orthogonal transformation of data. In this context, the input is the set of signals obtained from accelerometer and gyroscope, and the output of this transformation is a set of vectors called eigenvectors. Since the signals are obtained from users during walking, the eigenvectors are called $eigenwalks$. In this transformation, the first eigenwalk element presents the most dominant feature. To reduce the computation effort, the dimensionality of the original training set is usually reduced. In this study, the dimensionality is reduced by computing the eigenwalks on a characteristic list of the training set, based on a co-occurrence matrix.

A co-occurrence matrix shows the similarity of adjacent values at a given offset. It is usually exploited to process images [13], but its application in fall prediction has been recently investigated [12]. Fig. 1 shows an example of a co-occurrence matrix obtained from the accelerometer and gyroscope signals. Firstly, the acceleration and gyroscope data are stored in a matrix Δ: each column of the matrix presents a sample per instant. This matrix has 6 rows: at every instant, 3 values are obtained from the accelerometer and 3 from the gyroscope. The number of columns depends on the length of the monitoring interval, as one column is added at every instant. Firstly, each cell of the co-occurrence matrix is initialized to zero. Then, each couple of consecutive cells in Δ is analyzed to find similar patterns. Afterwards, the number of similar patterns is reported in the co-occurrence matrix at the coordinates given by the cell content. For example, if pairs (ax_1, ax_2) and (az_2, az_3) in Δ are equal, i.e., $ax_1 = az_2$ and $ax_2 = az_3$, the cell (round(ax_1), round(ax_2)) in the co-occurrence matrix is set to 2. Since the data obtained from accelerometer and gyroscope are float numbers and the indices of the co-occurrence matrix are integer, float numbers are rounded to the closest integer value. Moreover, the number of rows and columns of the co-occurrence matrix is equal to the max-

$$\Gamma = \begin{pmatrix} \text{Contrast}^1 & \cdots & \text{Contrast}^n \\ \text{Homogeneity}^1 & \cdots & \text{Homogeneity}^n \\ \text{Correlation}^1 & \cdots & \text{Correlation}^n \\ \text{Uniformity}^1 & \cdots & \text{Uniformity}^n \\ \text{Maximum probability}^1 & \cdots & \text{Maximum probability}^n \end{pmatrix}$$

Figure 2: Eigenwalk computation procedure.

imum value in Δ. For example, if round(ax_1) and round(ax_2) are the maximum values among the cells in Δ, the dimension of the co-occurrence matrix is equal to round(ax_1)$\times$round(ax_2).

Let p_{ij} be the ratio of the $(i, j)th$ element of the co-occurrence matrix to the sum of all the elements. The co-occurrence matrix can be characterized as follows:

- *contrast* shows the intensity between a cell and its neighbors:

$$contrast = \sum_{i=1}^{K}\sum_{j=1}^{K}(i-j)^2 p_{ij} \tag{10}$$

- *homogeneity* measures the spatial closeness of the distribution of elements to the diagonal of the co-occurrence matrix:

$$homogeneity = \sum_{i=1}^{K}\sum_{j=1}^{K}\frac{p_{ij}}{1+|i-j|} \tag{11}$$

- *correlation* shows how a value is related to its neighbors:

$$correlation = \sum_{i=1}^{K}\sum_{j=1}^{K}\frac{(i-m_r)(j-m_c)p_{ij}}{\sigma_r\ \sigma_c} \tag{12}$$

  where m_r, m_c, σ_r and σ_s are computed as follows:

$$m_r = \sum_{i=1}^{K}\sum_{j=1}^{K} ip_{ij},\ m_c = \sum_{j=1}^{K}\sum_{i=1}^{K} jp_{ij} \tag{13}$$

$$\sigma_r = \sum_{i=1}^{K}\sum_{j=1}^{K}(i-m_r)^2 p_{ij},\ \sigma_c = \sum_{j=1}^{K}\sum_{i=1}^{K}(i-m_c)^2 p_{ij}$$

- *uniformity* computes the sum of the squares of the elements:

$$uniformity = \sum_{i=1}^{K}\sum_{j=1}^{K} p_{ij}^2 \tag{14}$$

- *maximum probability* measures the highest value of the co-occurrence matrix. Simply, it can be computed by max(p_{ij}).

In order to compute the eigenwalks, firstly a new matrix Γ is created with the contrast, homogeneity, correlation, uniformity, and maximum probability of all samples, as shown in Fig. 2. For every row in Γ, the average value of the row is subtracted from all elements in the row, in order to build a new matrix Φ. Finally, the covariance matrix C is computed as $C = \Phi\Phi^T$. Then, the eigenvectors of C are computed. Generally speaking, given an $m \times m$ matrix K, the number λ is an eigenvalue of the matrix if there exists a non-zero vector v such that $Kv = \lambda v$. The vector v is called eigenvectors of K corresponding to the eigenvalue λ. Finally, the eigenvectors of the covariance matrix C are defined as eigenwalks because they present walking features.

4. CLASSIFICATION METHOD

In order to classify abnormal walks and normal daily activities, a classification algorithm should be performed on selected features. FPSs usually use the following classification methods:

1. Threshold-based algorithm
2. Decision tree (DT)
3. Support vector machine (SVM)
4. Multi-layer perceptron (MLP)
5. Random forest (RF)

A threshold-based algorithm utilizes a threshold to classify the feature set of the user gait. After being extracted from the input signals, the features are compared with pre-defined thresholds. Since the thresholds have an important effect on the performance of the algorithm, the biggest challenge of a threshold-based algorithm is determining the thresholds.

The decision tree is a directed tree with a root node without incoming edges, decision nodes with one incoming edge and possible outcoming edge; leaf nodes without outcoming edges. At the training stage, each internal node splits the instance space into two or more parts with the goal to optimize the performance of the classifier. Then, every path from the root to a leaf node forms a decision rule to determine which class a new instance belongs to [14].

A support vector machine is used for classifying separable classes of the observations. The SVM finds the best hyperplane with the maximum margin to separate two classes. The SVM can be defined as linear or non-linear according to the hyperplane function used for the classification.

The multi-layer perceptron is a neural network with different layers between input and output layers. The data flows in one direction from input to output layer.

Random forest is a large collection of decision trees. It consists of an arbitrary number of simple trees, which are used to determine the final outcome. For classification problems, the set of trees vote for the most popular class. The main idea in random forests is to reduce the correlation between the trees, without increasing the variance [8].

5. EVALUATION CRITERIA

Given a set with P positive instances and N negative instances, True Positive (TP) and True Negative (TN) are defined as correct identification of a true classification of positive and negative instance, respectively, while False Positive (FP) and False Negative (FN) misidentify a positive and negative instances. The most common criteria for evaluating classification algorithms in FPSs are:

1. *specificity* or *true negative rate (TNR)* measures the rate of negative instances that are correctly identified as negative:

$$specificity = \frac{\#TN}{\#TN + \#FP} \tag{15}$$

   Moreover, $generality$ is computed as 1 - *specificity*.

2. *sensitivity* or *true positive rate (TPR)* measures the rate of positive instances that are correctly identified as positive:

$$sensitivity = \frac{\#TP}{\#TP + \#FN} \tag{16}$$

3. *accuracy* counts the number of samples correctly classified:

$$accuracy = \frac{\#TP\ + \#TN}{\#P + \#N} \quad (17)$$

4. *error rate* is the number of wrong classifications:

$$error\ rate = \frac{\#FP\ + \#FN}{\#P + \#N} \quad (18)$$

5. *precision* is the ratio of correctly classified samples as positive to the number of all samples classified as positive:

$$precision = \frac{\#TP}{\#TP + \#FP} \quad (19)$$

6. *recall* is the ratio of correctly classified samples as positive to all real positive samples:

$$recall = \frac{\#TP}{\#TP + \#FN} \quad (20)$$

In addition, the ROC curve is a graphical plot that illustrates the performance of a classifier [15]. The curve is drawn by sensitivity and generality. The generality and sensitivity are plotted on the x and y axes of the ROC plot, respectively. The best classifier is located at the top left corner of the ROC graph, which represents 100% sensitivity and 100% specificity. For example, Fig. 3 shows the ROC of four prediction instances. The further the result is from the diagonal in the above space, the better the accuracy is, so A has the best prediction. C is the worst among the four instances, because it is below and far from the diagonal line. B is a good classifier but not as much as A, because it is above but not far from the diagonal line. Moreover, since D is closer to the diagonal line, it is preferable than C.

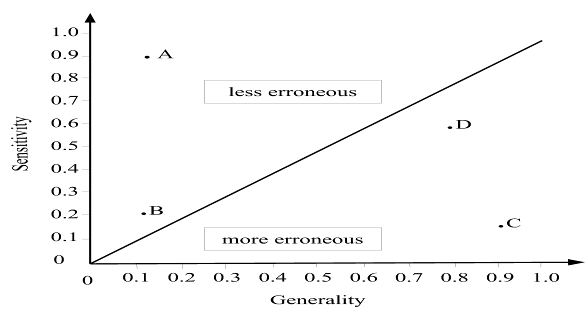

Figure 3: ROC plot.

6. EXPERIMENTAL ANALYSIS

A fall may be due to weakness, balance deficit, gait deficit, visual deficit and mobility limitation. Based on these fall factors, the methods used most frequently to simulate an abnormal walks are:

- walking with a straightened knee [9, 2, 16]
- walking with a leg length discrepancy [9, 2, 16]
- walking on a rough surface [17]
- walking through obstacles [11].

In the experiments, a flat area with different types of obstacle are prepared. Users are asked to walk through obstacles without looking them for 10 seconds, so they can lose their balance. This can

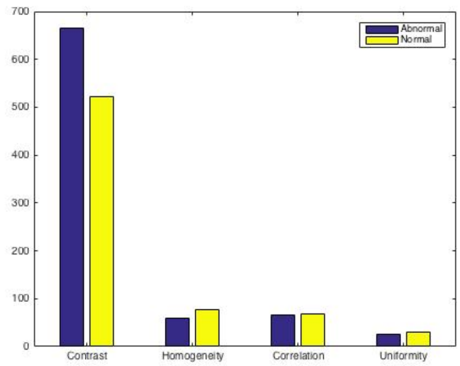

Figure 4: Normal VS abnormal walks.

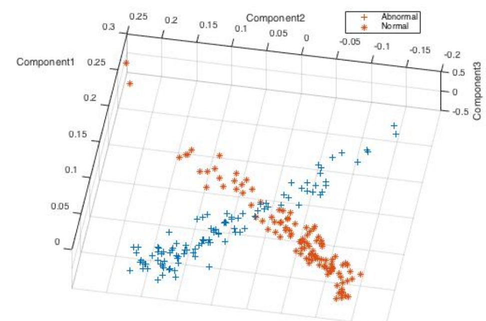

Figure 5: Normal VS abnormal eigenwalks.

be a good simulation of abnormal walks in real cases. The users in the experiments are 22 men with a weight in the range of 65-110 kg and height in the range of 160-185 cm, and 3 women with a weight in the range of 50-60 kg and height in the range of 157-165 cm. They are in the range of 18-35 years old. Since exploiting a middleware is a common approach to collect the user sensor data [18], MATLAB R2015b is used in this study. WEKA version 3.6.13[1], an open source tool for data mining, has been used to classify the data.

An iPhone 4S is adopted in the experiments, equipped with the STMicro STM33DH 3-axis accelerometer and the STIMicro AGDI 3-axis gyroscope. The smartphone is placed on the back trunk of the user to be closed to the center of the body.

Fig. 4 shows the average characterization of the co-occurrence matrix. Homogeneity, uniformity, maximum, and correlation in normal walks are higher than in abnormal walks. On the contrary, the contrast in abnormal walks is higher. However, these characterization need to be more precise to be effectively used for the walking classification. Therefore, the principal component analysis is computed. Fig. 5 shows the first three eigenwalks of the co-occurrence matrix. These eigenwalks are representative of normal and abnormal walks because they are located along two different directions in the plot.

Different state-of-the-art approaches are implemented [9, 11, 12]. Table 1 compares their performance with respect to the proposed method, by applying different classification methods. The accuracy is increased by more than 7%.

[1]http://www.cs.waikato.ac.nz/ ml/weka/

Measures	[9]	[11]	[12]	Proposed method
	DT	DT	MLP	RF
Accuracy	83.88	81.42	90.82	98.623
Error Rate	16.11	18.57	9.17	1.376
Sensitivity	0.88	0.78	0.87	0.991
Generality	0.21	0.15	0.06	0.018
Precision	0.81	0.83	0.93	0.981
Recall	0.88	0.78	0.87	0.991
F-Measure	0.84	0.82	0.90	0.986
ROC Area	0.86	0.82	0.96	0.999

Table 1: Result comparison.

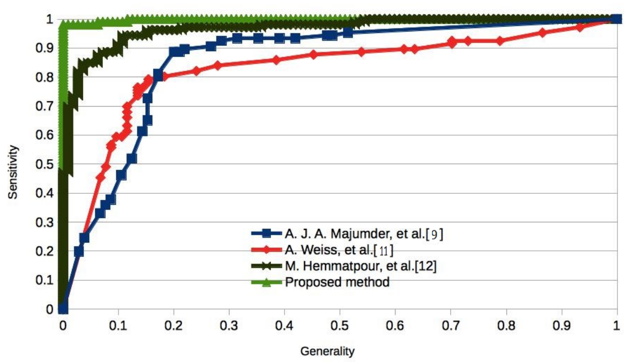

Figure 6: ROC curves of different approaches.

The best performance of the proposed method is confirmed by the ROC curves, as shown in Fig. 6: the area under the ROC curve of the proposed method is higher than for the other techniques.

7. CONCLUSIONS

This paper has analyzed various gait features based on data obtained from accelerometer and gyroscope. A new gait feature, called eigenwalk, has been proposed based on a co-occurrence matrix. Furthermore, the state-of-the-art fall prediction algorithms have been experimentally evaluated. Based on the presented results, the proposed feature in combination with a random forest classifier presents the best performance among the other fall prediction algorithms.

8. ACKNOWLEDGMENTS

This work was partially supported by the grant "Bando Smart Cities and Communities", OPLON project (OPportunities for active and healthy LONgevity) funded by the Italian Ministry for University.

9. REFERENCES

[1] Alexander, B. H., Rivara, F. P., and Wolf, M. E., "The cost and frequency of hospitalization for fall-related injuries in older adults." *American journal of public health*, vol. 82, no. 7, pp. 1020-1023, 1992.

[2] Majumder, A. J. A., Zerin, I., Uddin, M., Ahamed, S. I., and Smith, R. O., "Smartprediction: A real-time smartphone-based fall risk prediction and prevention system," in *Proceedings of the Research in Adaptive and Convergent Systems*, pp. 434-439, 2013.

[3] Shen, V. R., Lai, H., and Lai, A., "The implementation of a smartphone-based fall detection system using a high-level fuzzy Petri net." *Applied Soft Computing*, pp. 390-400, 2015.

[4] Li, Y. K., Ho, K. C., and Popescu, M., "A Microphone Array System for Automatic Fall Detection," in *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 5, pp. 1291-1301, 2012.

[5] Zigel, Y., Litvak, D., and Gannot, I., "A Method for Automatic Fall Detection of Elderly People Using Floor Vibrations and Sound-Proof of Concept on Human Mimicking Doll Falls," in *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 12, pp. 2858-2867, 2009.

[6] Bian, Z., Hou, J., Chau, L., and Magnenat-Thalmann, N., "Fall detection based on body part tracking using a depth camera" *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 2, pp. 430-439, 2015.

[7] Fortino, G., and Gravina, R., "Fall-MobileGuard: a smart real-time fall detection system." in *Proceedings of the 10th EAI International Conference on Body Area Networks*, pp. 44-50, 2015.

[8] Friedman, J., Hastie, T., and Tibshirani, R.. "The elements of statistical learning second edition", Springer, 2001.

[9] Majumder, A. J. A., Rahman, F., Zerin, I., William, E. J., and Ahamed, S. I., "iPrevention: Towards a novel real-time smartphone-based fall prevention system," in *28th Proceedings of the ACM Symposium on Applied Computing*, pp. 513-518, 2013.

[10] Hjorth, B., "EEG analysis based on time domain properties," *Electroencephalography and clinical neurophysiology*, vol. 29, no. 3, pp. 306-310, 1970.

[11] Weiss, A., Shimkin I., Giladi N., and Hausdorff, J. M., "Automated detection of near falls: algorithm development and preliminary results," *BMC research notes*, vol. 3, no. 1, p. 1-8, 2010.

[12] Hemmatpour, M., Ferrero, R., Montrucchio, B., and Rebaudengo, M., "A neural network model based on co-occurrence matrix for fall prediction" In *Proceedings of the 6th EAI International Conference on Wireless Mobile Communication and Healthcare*, 2016.

[13] Gonzalez, R.C., and Richard, E. W., "Digital image processing", 2008.

[14] Dai, W., and Ji, W., "A mapreduce implementation of C4. 5 decision tree algorithm." In *International Journal of Database Theory and Application*, vol. 7, no. 1, pp. 49-60, 2014.

[15] Fawcett, T., "An introduction to ROC analysis." Pattern recognition letters, vol. 27, no. 8, pp. 861-874, 2006.

[16] Majumder, A. J. A., Zerin, I., Ahamed, S. I., and Smith, R. O., "A multi-sensor approach for fall risk prediction and prevention in elderly," in *ACM SIGAPP Applied Computing Review*, vol. 14, no. 1, pp. 41-52, 2014.

[17] Otis, M. J-D., and Menelas, B. A. J., "Toward an augmented shoe for preventing falls related to physical conditions of the soil." In *IEEE international conference on systems, man, and cybernetics*, pp. 3281-3285, 2012.

[18] Fortino, G., Giannantonio, R., Gravina, R., Kuryloski, P., and Jafari, R., "Enabling effective programming and flexible management of efficient body sensor network applications." in *IEEE Transactions on Human-Machine Systems*, vol. 43, no. 1, pp. 115-133, 2013.