Improving Automatic Fall Detection with Spectral and Temporal Features

Dhanush Karthikeyan[1], Nicholas Coyle[2], Dr. Kaibao Nie[3], Dr. Tadesse Ghirmai[3]
[1]California Polytechnic University Pomona, [2]Western Washington University, [3]University of Washington Bothell

Background

The independent life of the elderly can drastically change after a fall, with both their physical and emotional health becoming compromised depending on the severity of the incident. Automatic fall detection can alert first responders to administer prompt treatment.

Research in fall detection from the SmartFall [1] team uses temporal analysis of accelerometer signals to detect falls with accuracy ranging from 65-99%. There is extensive overlap in temporal analysis between falls and the activities of daily living which led to our exploration of different techniques to mitigate this issue. We propose combining spectral and temporal analysis to improve the performance of fall detection algorithms.

Motivation

- Falling is the leading cause of injury and injury-related deaths among the elderly.
- Improving reliability of fall detection so wearable medical devices may automatically detect falls and alert first responders.

Problem Statement

Improving fall detection by optimizing sliding window size and feature extraction.

Proposed Methods

- Utilize both spectral and temporal features
- Variance of Power Spectral Density (PSD)
- Resultant Acceleration, min and max acceleration [1]

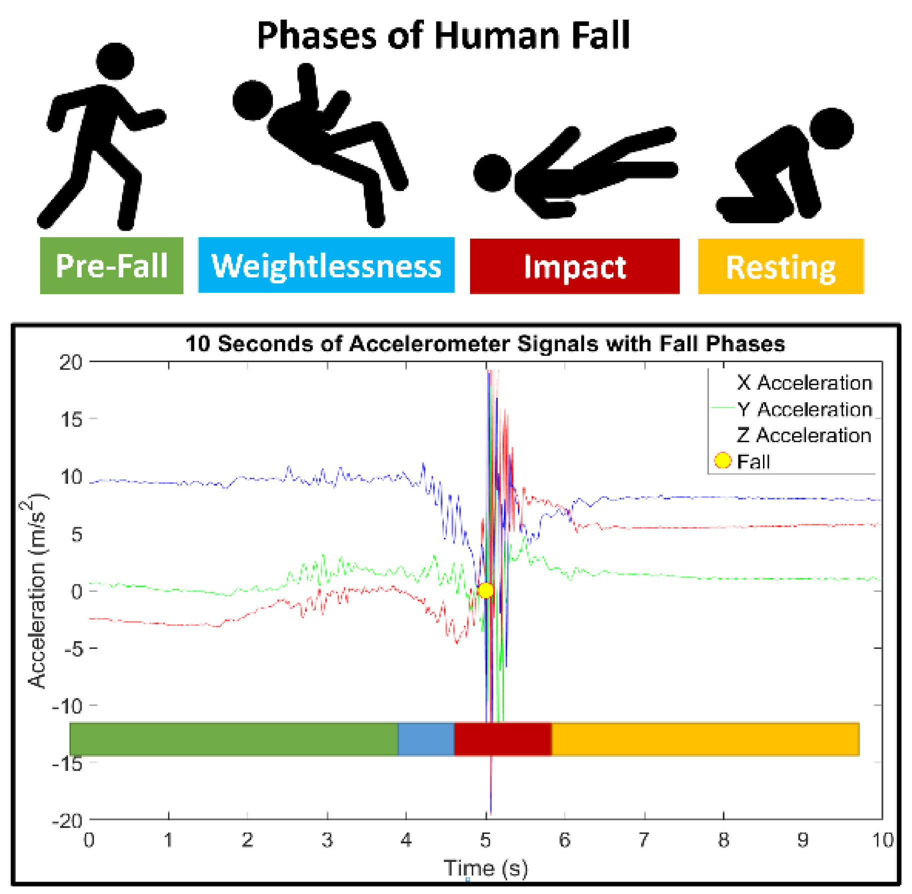


Fig 1. Triaxial accelerometer sensor signals with the phases of the fall.

Data Analysis

FARSEEING Dataset

Waist-mounted triaxial accelerometer data from 23 real and unplanned falls in elderly subjects (age 56+) was utilized from the FARSEEING Dataset [3]. Each fall was prelabeled at a singular sample point by an expert in fall signal processing. This data was processed with an Elliptical IIR high pass filter.

Smartwatch Dataset

Another dataset used has 240 simulated falls in 10 young healthy male subjects wearing a wrist-mounted device. Each fall was prelabeled with all sample points in a 750ms window. This triaxial accelerometer data was not filtered.

Optimizing Window Size

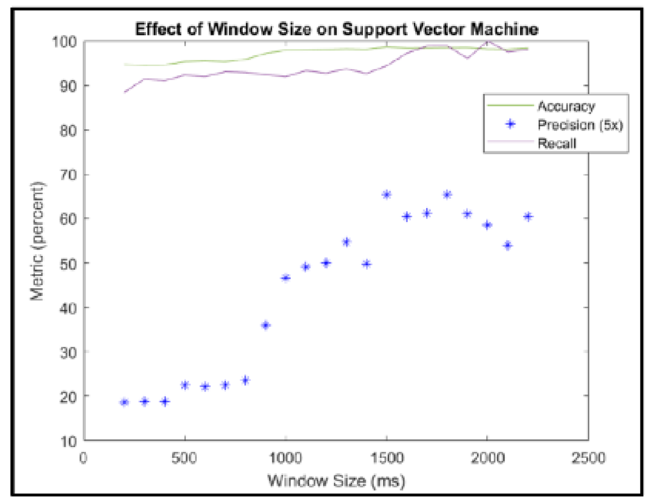


Fig 2. Varying Window Size on SVM

In the literature, there is no consensus on the optimal size of the sliding window. Window sizes were iterated from 200ms to 2200ms for the whole Machine Learning process (Fig 2). 2000ms is optimal.

Feature Extraction

The Weightlessness and Impact features (Figure 1) are chosen for feature extraction to detect a weightlessness as well as an impact phase that might signal a fall. The extracted features are shown in Table 1.

Table 1. Feature Extraction

Feature Groups	Features Extracted
Legacy [1]:	• A_{res} = *resultant acceleration vector*
	• S_{min} = *minimum A_{res}*
	• S_{max} = *maximum A_{res}*
	• dS = *difference between S_{min} and S_{max}*
New Proposed	• Impact = *(max(Ares) > 2g*)*
	• Weightlessness = *(max(Z_Acc*) > 9 && min(Z_Acc*) < 4)*
	• Variance of PSD

*g = 9.81m/s²

**Z_Acc = Z-axis Acceleration*

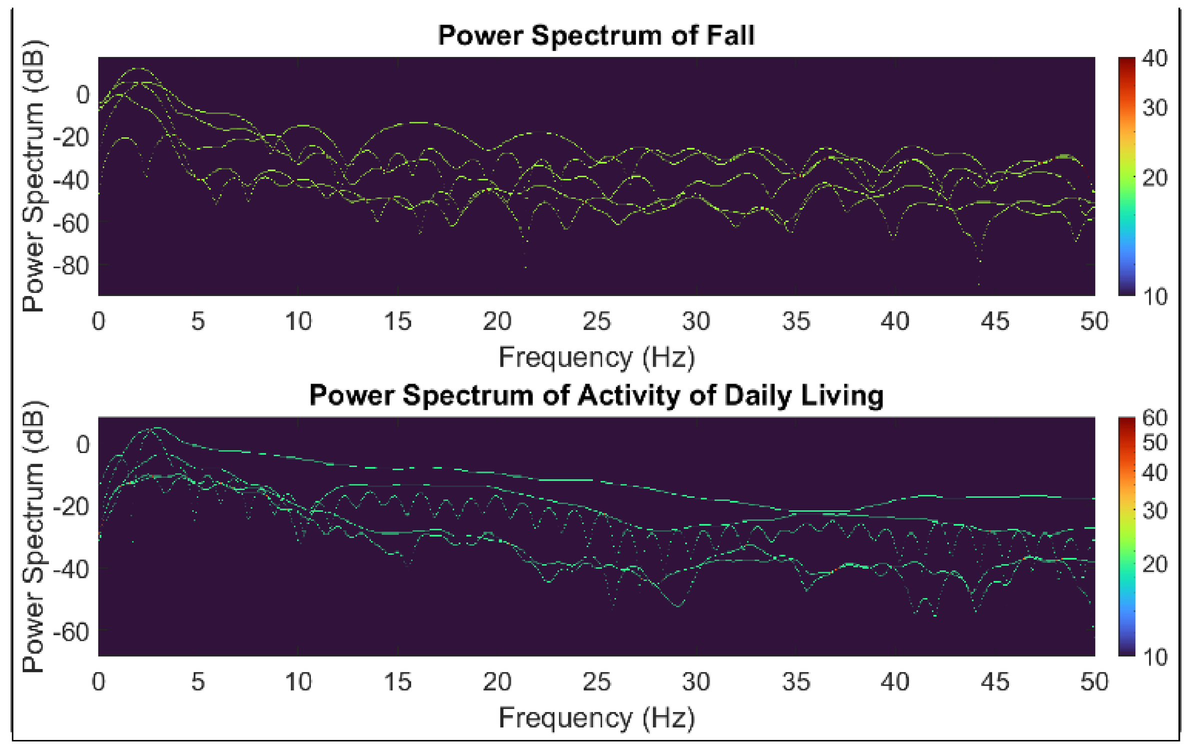


Fig 3. Power Spectrum of Fall and Non-Fall Signals

Variance of Power Spectral Density

- Frequency domain analysis is needed due to overlap between falls and ADLs in time domain.
- Peak power occurs at various frequencies in 23 different falls, but a consistent trend is high variance.

PSD is calculated using Welch's PSD estimate in MATLAB. The variance of PSD is shown in equation 3.

$$\sigma^2 = \frac{\sum_{i=1}^{n}(X_i - \mu)^2}{n} \quad (3)$$

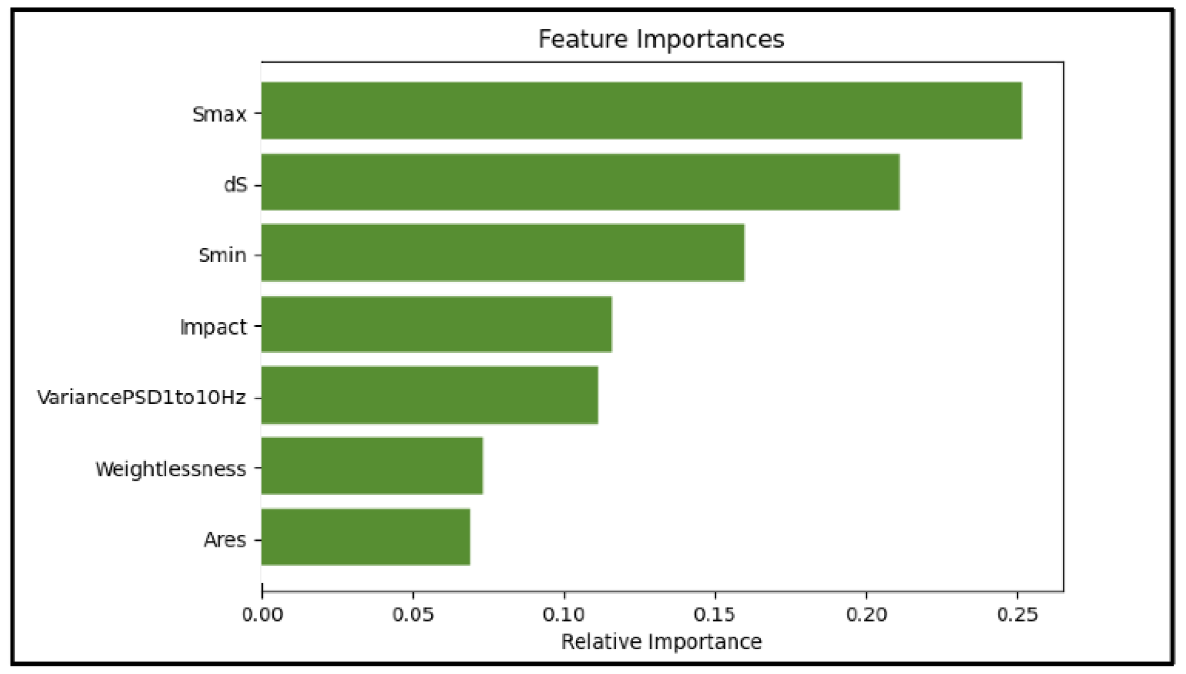
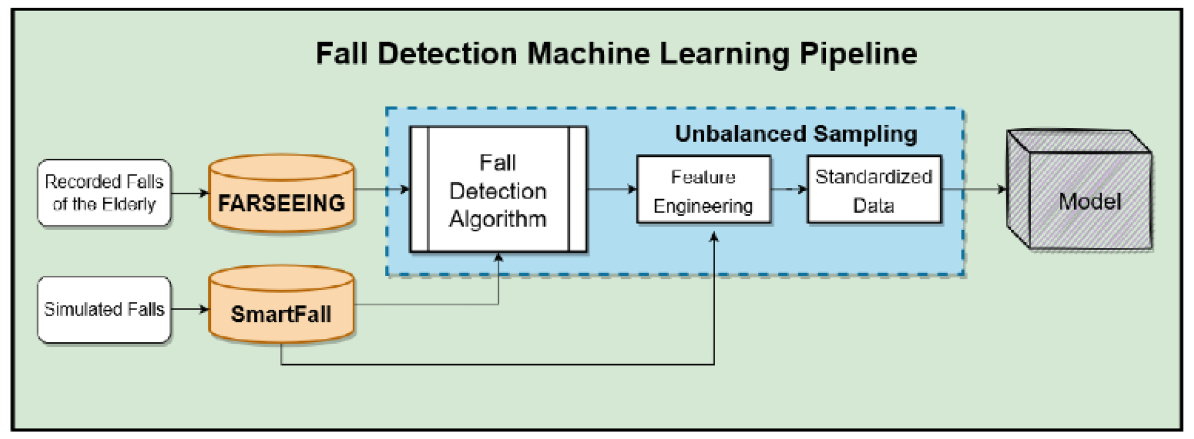


Fig 4. Feature Ranking Results

Machine Learning Pipeline

Data is passed through the following pipeline to assemble labelled datasets and train/test the models.



Methodology

FARSEEING Dataset

Features were calculated in a 2000ms sliding window. Data was split 80/20 training/testing and an imbalance was left to reflect the scarcity of real fall events.

Smartwatch Dataset

Data was already divided into training and testing sets. Because the falls were prelabeled in 750ms windows, features were calculated over a 750ms sliding window.

Preliminary Results

Preliminary results comparison with previous literature can be seen in Table 1.

Table 1: Comparison of Fall Detection Performance on 2 Datasets

		SmartFall Results		Our Results	
		NB	SVM	NB*	SVM*
FARSEEING	*Precision*	0.01	0.44	0.29	0.74
	Recall	0.82	0.55	0.80	0.84
	Accuracy	0.87	0.99	0.99	0.99
Smartwatch	*Precision*	0.60	0.68	0.71	0.75
	Recall	0.92	0.86	0.61	0.44
	Accuracy	0.65	0.73	0.92	0.91

*Our Results, NB=Naive Bayes, SVM=Support Vector Machine

This work demonstrates the potential benefit of adding spectral and temporal features in automatic fall detection. Performance improved with Naïve Bayes and Support Vector Machine classifiers when applied to either database.

Further improvements may be possible with analysis on a larger dataset. In the future, we would like to implement a neural network with this feature set.

References

[1] Mauldin, R. et al. SmartFall: A Smartwatch-Based Fall Detection System Using Deep Learning. Sensors 2018, 18, 3363.

[2] Shin, I., Son, J., Ahn, S., Ryu, J., Park, S., Kim, J., Cha, B., Choi, E., & Kim, Y. (2015). A Novel Short-Time Fourier Transform-Based Fall Detection Algorithm Using 3-Axis Accelerations. Mathematical Problems in Engineering, 2015, [394340]. https://doi.org/10.1155/2015/394340

[3] Klenk, J., Schwickert, L., Palmerini, L. et al. The FARSEEING real-world fall repository: a large-scale collaborative database to collect and share sensor signals from real-world falls. Eur Rev Aging Phys Act 13, 8 (2016). https://doi.org/10.1186/s11556-016-0168-9

Acknowledgements

This work was supported by the National Science Foundation REU Award #1757395.