Online Fall Prediction with TinyML

Alireza Rahimi[1,2], Daiyan Khan[2], Varun Sundaram[2], Matthew Laskey[2], Hung Cao[1,2]

[1]Analytics Everywhere Lab, University of New Brunswick
[2]Faculty of Computer Science, University of New Brunswick, NB, CA

UNB Analytics Everywhere Lab
EST. 1785 UNIVERSITY OF NEW BRUNSWICK

Abstract

The rising prevalence of fall-related injuries in the elderly demands an urgent need for rapid and accurate fall detection mechanisms. This research aims to offer a robust and power-efficient solution for fall detection implementation. We propose a novel online fall prediction approach leveraging Advanced Edge Computing, IoT, and TinyML. Specifically, our method relies on local computation on micro-controllers. Therefore, it eliminates data transfer steps. As a result, it enhances response time and can conserve power on wearable devices. Preliminary results indicate promising improvements in accuracy (by 5%) and latency compared to existing methods. The proposed system demonstrates a total delay of 4ms for fall detection, outperforming previous approaches in accuracy.

Motivation

- Falls cause a major health burden for older adults, with 30–40% of those over 70 experiencing falls annually, especially in long-term care settings, accounting for over half of their injuries [2]. These alarming statistics highlight the critical importance of timely interventions to prevent injuries and potentially save lives. Detecting a fall and deploying a safety mechanism must happen within a few milliseconds to prevent injuries.
- Traditional fall detection methods [1, 3, 4] often face challenges related to response time, accuracy, and power consumption, especially when relying on external devices or cloud-based processing.
- Recent emerging advancements in Edge Computing, IoT, and wearable technology offer a compelling opportunity to develop innovative solutions that address these limitations.

Proposed Method

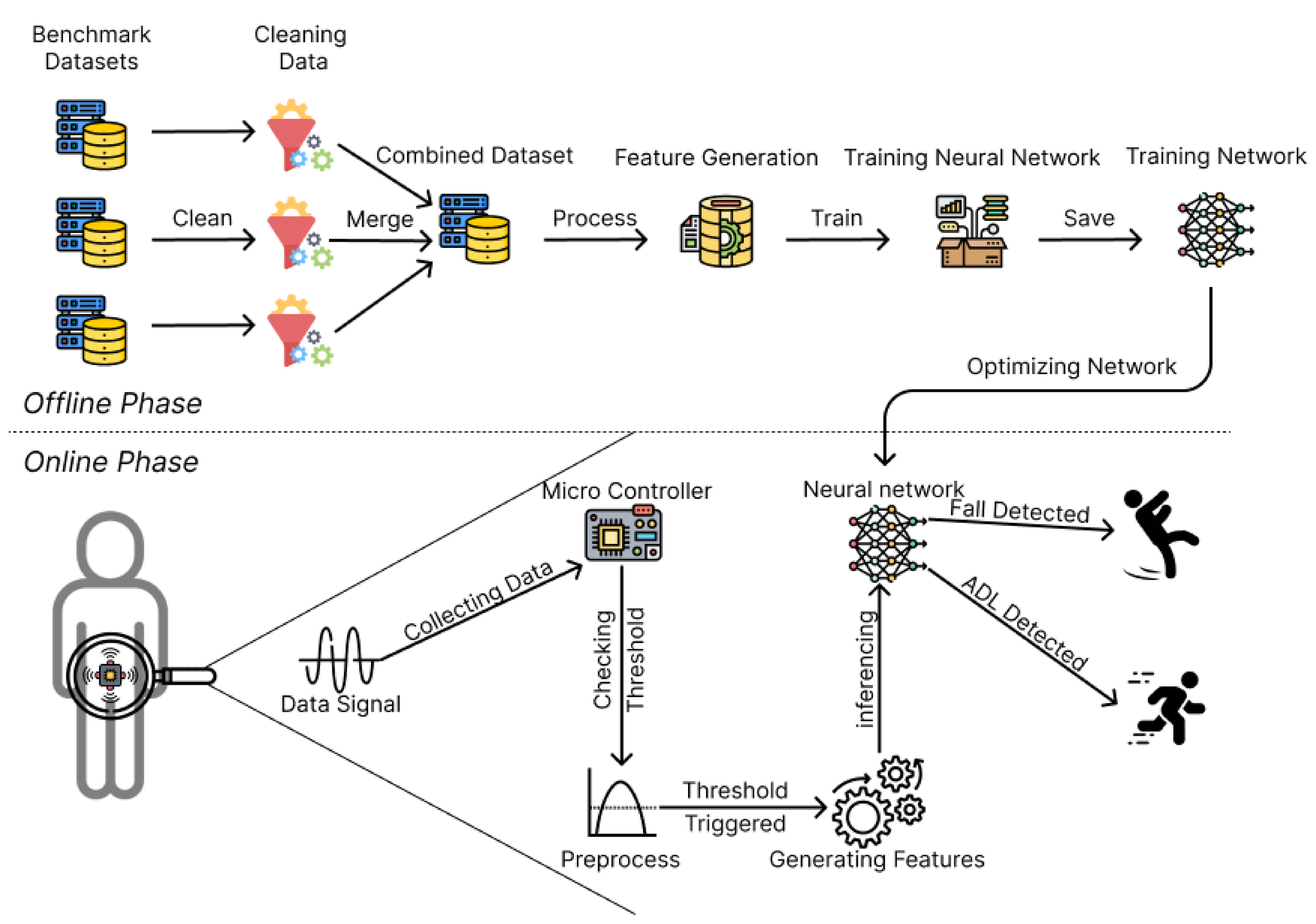

Figure 1. Overview proposal for online fall prediction with TinyML

Figure 1 illustrates the overview of our proposed method which comprises two distinct phases: offline and online, aimed at robust fall detection utilizing accelerometer data.

- **Offline phase:** Multiple benchmark datasets have been cleaned and merged to create a comprehensive training dataset. Next, important features would be extracted and used for training a neural network model. Then, the model would be optimized for deployment on a micro-controller of the user's wearable devices.
- **Online phase:** Accelerometer data streams is continuously collected and monitored on the user's wearable devices. A threshold check operation would be leveraged to detect potential fall events. Upon surpassing the threshold, relevant features are extracted and used for inference on the optimized pre-trained neural network model, enabling real-time prediction of fall occurrences.

Implementation

Below are the critical steps and techniques employed to optimize fall detection performance for deployment on micro-controllers:

1. Data Processing: Figure 2 described data processing steps. First, we applied a moving Gaussian average to suppress noise in raw data while retaining signal characteristics to enhance fall detection accuracy. Next, the calculation of moving standard deviation identified segments indicative of falls, isolating significant fluctuations for further analysis and training of neural network models. This technique extracts meaningful sections, enhancing the efficiency of fall detection systems.

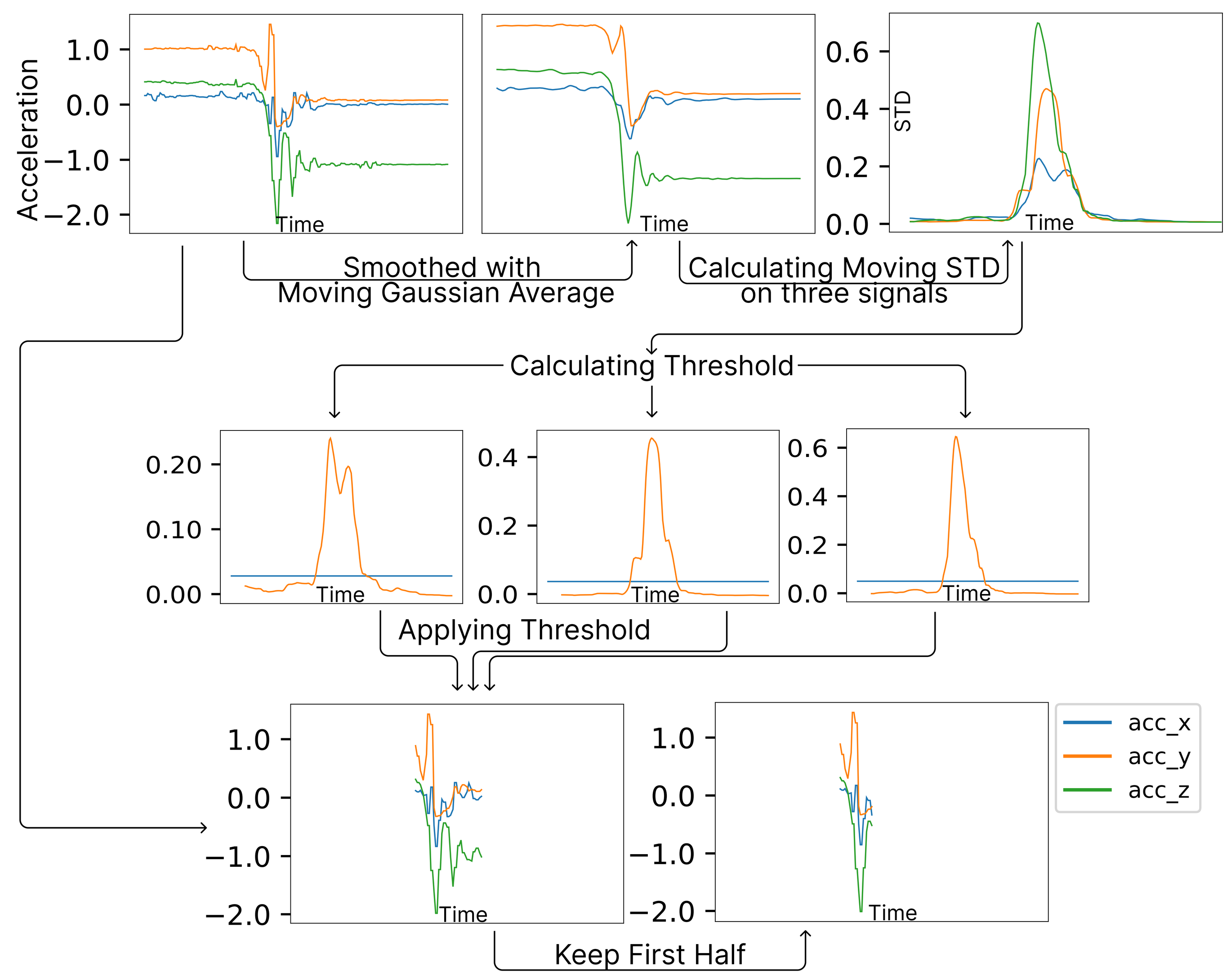

Figure 2. Data Processing Overview

2. Feature Generation: Efficient representation of accelerometer data is achieved through statistical and spectral feature generation. By computing RMS, skewness, kurtosis, and FFT, we extracted 33 features from 1-second windows, condensing input size while preserving essential characteristics.
3. Neural Network: We trained a network with 32-bit float weights, then quantized to 8-bit integers for efficient deployment without sacrificing performance. Also, we used an optimized inference engine on micro-controller to reduce the memory usage.
4. Threshold: Integrating a threshold mechanism alongside neural network usage conserves energy while maintaining effective fall detection. By activating the network only upon surpassing predefined thresholds, false positives are mitigated, optimizing energy efficiency without compromising reliability.

Preliminary Results and Discussion

Our model has shown promising improvements over the previous ones regarding accuracy and latency.

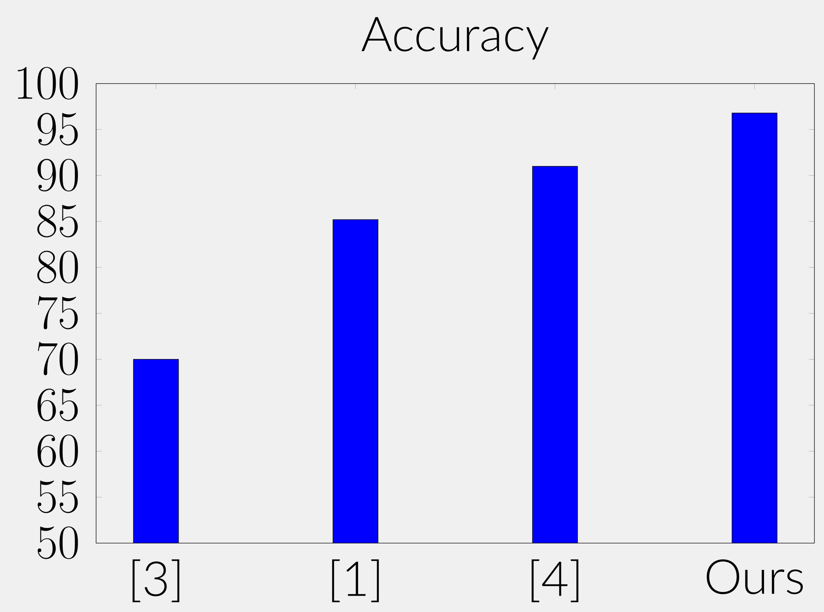

Figure 3. Comparison of different models

- **Accuracy:** Our approach achieved an accuracy of 96.8% which is more than a 5% improvement over the previous state-of-the-art model [4]. This reduces the error to 3.2%, thereby optimizing the performance and enhancing the deployment of a safety mechanism.
- **Delay:** The delay in our proposed optimized model is 1 ms for the neural network to inference on the micro-controller, 2 ms for feature extraction, and about 1 ms more for threshold, which adds up to a total delay of 4 ms.

Overall, our model has outperformed previous models and gives scope for deploying enhanced safety mechanisms with fall predictions.

Conclusion

We have proposed a novel online approach for fall detection that outperformed state-of-the-art methods in case of accuracy. We leverage TinyML methods to train and optimize a neural network that can be deployed on a micro-controller with a limited amount of memory and CPU. Our approach does not rely on any auxiliary device to transfer data or offload a computational task. By this means, we reduce the response time to create a promising opportunity for future safety mechanics which can be enabled upon fall detection to prevent injuries in elderly people.

References

[1] Dharmitha Ajerla, Sazia Mahfuz, and Farhana Zulkernine.
A real-time patient monitoring framework for fall detection.

[2] Efraim Jaul and Jeremy Barron.
Age-related diseases and clinical and public health implications for the 85 years old and over population.

[3] Taylor R. Mauldin, Marc E. Canby, Vangelis Metsis, Anne H. H. Ngu, and Coralys Cubero Rivera.
SmartFall: A smartwatch-based fall detection system using deep learning.

[4] Farah Othmen, Mouna Baklouti, André Eugenio Lazzaretti, and Monia Hamdi.
Energy-aware IoT-based method for a hybrid on-wrist fall detection system using a supervised dictionary learning technique.

Acknowledgement: The devices used in the experiments were supported by CFI Project Number 39473 - Smart Campus Integration and Testing (SCIT Lab)

alireza.rahimi@unb.ca