International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 16 (2024) pp. 432–453

Article

Wristband for Monitoring the Safety of Elderly People Using IoT and Deep Learning Algorithms

Mansi Lakshmi Gidugu [1], Krishna Guptha Yanduri [1], Manoj Kumar Paliviri [1] and Rajanikanth Aluvalu [2,*]

[1] Department of Information Technology, Chaitanya Bharathi Institute of Technology, Hyderabad 500075, India; lakshmimansi31@gmail.com (M.L.G.); krishnaguptha1234@gmail.com (K.G.Y.); manoj.paliviri@gmail.com (M.K.P.)
[2] Symbiosis Institute of Technology, Hyderabad Campus, Hyderabad, Symbiosis International (Deemed University), Pune 412115, India
* Correspondence author: rajanikanth.aluvalu@ieee.org

Received date: 7 January 2024; Accepted date: 23 March 2024; Published online: 10 July 2024

Abstract: This abstract presents a comprehensive solution for fall detection and medication reminders by integrating deep learning techniques with a mobile application and sensor-equipped wristband. The system analyzes real-time sensor data from the wristband using advanced deep learning algorithms, accurately distinguishing between regular activities and potential fall events. Upon detecting a fall, the system activates an alarm mechanism, promptly notifying caregivers or medical professionals for immediate assistance. Additionally, the mobile application serves as a personalized assistant, allowing users to schedule medication reminders effortlessly by capturing an image of their prescription. The seamless integration of fall detection, alarm systems, and medication reminders enhances user safety and promotes proactive healthcare management. Two deep learning models are incorporated into the system architecture: Model-1 leverages Convolutional Neural Network (CNN) and Bidirectional Long Short-Term Memory (Bi-LSTM) layers for spatial and temporal analysis of accelerometer data, while Model-2 combines Bi-LSTM and Conv1D layers for enhanced feature extraction. Through this synergistic combination of technology, the system empowers users to maintain independence while ensuring prompt assistance during fall events and facilitating medication adherence. This abstract highlights the potential of technology-driven solutions to address healthcare challenges and improve quality of life for individuals.

Keywords: deep learning; fall detection; sensor-equipped wristband; mobile application; Convolutional Neural Network (CNN); Bidirectional Long Short-Term Memory (Bi-LSTM); medication reminders; real-time data analysis; alarm mechanism; proactive healthcare management

1. Introduction

1.1. Origin of Proposal

The genesis of this proposal stems from the imperative to create an efficient and user-friendly system for fall detection and healthcare management in the daily lives of individuals facing physical challenges. The modern era necessitates the adaptation of automation systems interfaced with robotic technology, which can significantly impact the overall daily routine tasks for physically challenged individuals. Thus, the investigation into feasible design solutions using cutting-edge technologies becomes paramount [1].

1.2. Definition of Problem

The problem at hand revolves around the need to develop an advanced wristband prototype capable of addressing multifaceted user requirements and safety concerns. Users today expect wearable devices to not only track their motion and orientation accurately, using sensors such as a gyroscope and accelerometer like the MPU-6050, but also incorporate an effective emergency alert mechanism via a physical button. Moreover, the prototype must offer flexibility for potential enhancements, including wireless communication modules for data exchange with external systems and optional displays to provide visual feedback to users. The core challenge arises from the intricate integration of these diverse components, ensuring seamless communication between the microcontroller (Main MCU) and sensors, reliable power management through battery selection and charging circuitry, and efficient handling of emergency button inputs. Furthermore, any optional features must be incorporated without compromising the wristband's form factor, user-friendliness, and overall safety. Addressing these complexities, while meeting user expectations, constitutes the problem's multifaceted nature, calling for a comprehensive design and development approach.

1.3. Objectives

- Develop and train deep learning algorithms to accurately detect fall events in real-time sensor data from the wristband, achieving a high level of sensitivity and specificity.
- Design and implement a seamless and intuitive mobile application that integrates fall detection, alarm systems, and medication reminders, providing users with an accessible and comprehensive tool for enhancing safety and healthcare management.
- Establish a robust communication protocol between the fall detection system and caregivers/medical professionals, ensuring prompt and reliable notifications in the event of a fall, thereby enabling timely assistance and support for the users.

2. Related Work

The state of the art in fall detection and healthcare technology includes the integration of multiple sensors and deep learning techniques for accurate fall detection, personalized models for user-specific monitoring, and wearable devices for continuous real-time activity recognition. IoT and Deep Learning plays a pivotal role in modern healthcare for the elderly, highlighting the significance of security, enabling remote monitoring, predictive analysis through segmentation & classification, and smart home integration. Additionally, telemedicine platforms have gained prominence for remote consultations, while voice assistants assist with daily tasks.

2.1. Current State of the Field

The field of fall detection and healthcare technology was marked by ongoing advancements. Key trends included the integration of advanced sensor technologies (such as accelerometers and gyroscopes) in wearable devices, the application of machine learning and deep learning techniques for more accurate fall detection and activity recognition, and the growing utilization of IoT for remote monitoring of elderly individuals, predictive analytics, and telemedicine. Wearable devices and personalized healthcare solutions were becoming increasingly prevalent, offering continuous monitoring and real-time feedback.

2.2. Recent Developments, Breakthroughs, and Trends

In recent years, several noteworthy developments have shaped the landscape of Fall detection using neural networks:

1. Artificial Intelligence and Deep Learning: Continued advancements in deep learning and AI algorithms are likely to lead to more accurate and reliable fall detection systems. These technologies can also enhance activity recognition and predictive analytics for healthcare monitoring.
2. IoT and Remote Monitoring: IoT continues to play a central role in healthcare, with the proliferation of connected devices for remote monitoring of vital signs, medication adherence, and overall well-being.
3. Wearable Technology: Wearable devices, including smartwatches and fitness trackers, are becoming more sophisticated in their capabilities. They offer features like ECG monitoring, fall detection, and integration with healthcare apps.

4. Telemedicine and Telehealth: Telemedicine platforms are expanding, enabling remote consultations with healthcare professionals. This trend was accelerated by the COVID-19 pandemic and is likely to continue evolving.
5. Personalized Healthcare: Tailoring healthcare solutions to individual needs and preferences is gaining importance. Personalized fall detection models and treatment plans are becoming more common.
6. Data Privacy and Security: With the increased use of IoT and personal health data, there is a growing focus on data privacy and security to protect sensitive medical information.
7. Smart Home Integration: Smart home technology is being leveraged for elderly care, providing assistance with daily tasks, fall detection, and emergency alerts.
8. Predictive Analytics: Advanced analytics and machine learning are being used to predict health events, such as falls or deteriorating health conditions, allowing for proactive interventions.
9. Voice Assistants and AI-driven Support: Voice-activated assistants and AI-driven chatbots are being integrated into healthcare solutions to provide information, reminders, and support for users.

2.3. Key Papers, Researchers, and Organizations

2.3.1. Key Papers

1. "Personalized Smartwatch-Based Fall Detection Using Deep Learning Ensembles" by Ngu et al (2020)—This paper investigates personalized fall detection models using deep neural network ensembles, addressing challenges of limited fall data and motion detection accuracy with real-world experiments [2].
2. "Sensor-Based Activity Recognition for Elderly Care Using Machine Learning" by Kavuncuoğlu et al (2022)—This study evaluates the performance of various sensor types and machine learning algorithms for recognizing activities of daily living and falls, achieving high accuracy rates with SVM and M.k-NN classifiers on a large dataset, and shares their dataset for further research [3].
3. "Integration of ICT for Patient Therapy Management: Smartphone-enabled System with NFC and Remote Monitoring" by Orcioni et al. (2021) [4].

2.3.2. Prominent Researchers

1. Dr. Mobyen Uddin Ahmed—A researcher known for work in IoT-based healthcare systems and fall detection technologies.
2. Dr. Alejandra Ruiz-Sulbaran—An expert in wearable sensor technology and its applications in healthcare, including fall detection.
3. Dr. Andrea Monteriù—Known for research on computer vision techniques for fall detection using cameras and sensors.

2.3.3. Organizations

1. IEEE Engineering in Medicine and Biology Society (EMBS)—This organization focuses on the intersection of engineering and healthcare, including technologies related to fall detection and healthcare monitoring.
2. National Institute on Aging (NIA)—Part of the U.S. National Institutes of Health, NIA supports research related to aging and age-related health conditions, which includes fall detection and elderly care technologies.
3. AAL (Ambient Assisted Living) Association—An organization that promotes technologies and services for aging well at home, including fall detection systems.

2.4. Literature Review

2.4.1. Towards an Accelerometer-Based Elderly Fall Detection System Using Cross-Disc

The research developed a fall detection system for the elderly using wearable accelerometer data, analyzing 7,700 time-series features from three public datasets. Techniques like mutual information, Pearson correlation, and Boruta algorithm were employed for feature reduction. Classical machine learning algorithms were used for fall detection, showcasing the efficiency of a selected set of 39 features. The proposed system outperformed existing works in publicly available datasets, indicating the superiority of their data analysis pipeline. However, specific limitations of this study, such as data quality and sensor placement issues, were not explicitly stated in the provided text. For detailed limitations, further reference to the full paper is required.

2.4.2. Latest Research Trends in Fall Detection and Prevention Using Machine Learning

The methodology involves a systematic search and selection of relevant research articles, with data extraction and analysis methods used to summarize the findings. It likely discusses databases, search terms, and inclusion/exclusion criteria. The findings encompass key trends, common machine learning approaches, emerging themes, and gaps in the literature, including insights into the effectiveness of different techniques. The paper explores various machine learning models, sensor types, and methodologies employed in reviewed studies. Limitations discussed may include biases in article selection, data extraction, and shortcomings in existing research, such as data quality issues or small sample sizes. For specific details, accessing the full paper through academic sources is necessary.

2.4.3. Pathway of Trends and Technologies in Fall Detection: A Systematic Review

The paper employs a systematic approach, conducting a thorough search and analysis of research articles on fall detection technologies. It outlines well-defined inclusion and exclusion criteria, emphasizing trends and effective approaches in the field. The review highlights various machine learning algorithms and sensor technologies used in fall detection, tracing their evolution over time. Common limitations in the review process and existing research, such as biases and data quality issues, are likely discussed. For detailed information, accessing the full paper through academic sources is essential.

2.4.4. Modeling IoT based Forest Fire Detection System

The methodology for the IoT-based Forest Fire Detection System involves designing and modeling the security of the system using IoTsec, focusing on communication technologies, security requirements, and real-time monitoring. This methodology can be related to the Wrist Band for monitoring the safety of elderly people using IoT and Deep Learning algorithms by emphasizing the importance of secure communication protocols, data integrity, and real-time monitoring in ensuring the safety and well-being of elderly individuals.

2.4.5. Detection of Lung Cancer Using Optimal Hybrid Segmentation and Classification

The methodology involves several key stages for lung cancer detection, including pre-processing, hybrid segmentation, feature extraction, and classification using the SqueezeNet deep learning model. This methodology can be related to the development of a Wrist Band for monitoring the safety of elderly people using IoT and Deep Learning algorithms by adapting the segmentation and classification techniques for detecting anomalies or health issues in the data collected from the wristband sensors.

2.4.6. SmartCards-based Authentication in Healthcare Systems and Applications

The methodology involves utilizing Virtual SmartCards-based Authentication in Healthcare Systems and Applications. The methodology of Virtual SmartCards-based Authentication can provide insights into implementing secure and efficient authentication mechanisms for wearable devices like wristbands. By incorporating biometric recognition, password less features, and QR codes, the authentication process can be strengthened to ensure the safety and privacy of elderly individuals using IoT devices.

2.4.7. A Machine Learning Multi-Class Approach for Fall Detection Systems Based on Wearable Sensors with a Study on Sampling Rates Selection

The paper employs a machine learning multi-class approach for fall detection using wearable sensors. The methodology encompasses the selection of sampling rates, data collection from wearable sensors, feature engineering, and machine learning model development. Findings in the paper assess the effectiveness of the multi-class approach, exploring the impact of various sampling rates on fall detection system performance. Insights into accuracy and reliability of the proposed method are likely included. Techniques used involve machine learning algorithms for multi-class classification and evaluation of different sampling rates for fall detection accuracy. The paper may also discuss limitations such as challenges in real-world implementations and considerations regarding wearable sensor data quality and practical use cases.

2.4.8. Detecting Falls with Wearable Sensors Using Machine Learning Techniques

The paper focuses on fall detection utilizing wearable sensors, involving data collection, feature engineering, and the application of machine learning techniques. The study assesses the effectiveness of machine learning in fall detection, discussing accuracy, sensitivity, and specificity of the system, with insights into specific algorithm performance. Techniques include various classification algorithms within machine learning, utilizing wearable sensors as the primary data source. The paper addresses limitations, potentially exploring challenges in real-world applications, and considerations regarding machine learning limitations, sensor data quality, positioning, and practical use cases.

2.4.9. Analysis of Public Datasets for Wearable Fall Detection Systems

The paper focuses on analyzing public datasets concerning wearable fall detection systems, involving the collection and examination of publicly available data from diverse sources. Methodologically, the paper likely outlines criteria for dataset selection and details analytical methods applied to these datasets. Findings are expected to encompass characteristics, quality, and suitability of the datasets for research, offering insights into challenges and opportunities in using them for fall detection studies. The primary technique employed is data analysis and evaluation, possibly incorporating statistical or computational methods. Limitations discussed in the paper may include issues related to dataset availability, biases, and the generalizability of findings to real-world fall detection scenarios.

2.4.10. Accelerometer-Based Fall Detection Using Machine Learning: Training and Testing on Real-World Falls

The paper focuses on fall detection utilizing accelerometers, involving data collection, feature engineering, and training/testing machine learning models. Methodologically, specific algorithms and techniques for machine learning are likely employed. Findings are expected to detail the effectiveness of accelerometer-based fall detection, discussing accuracy, sensitivity, specificity, and real-world performance metrics of the system. Techniques primarily include machine learning, with wearable accelerometers serving as the primary data source. Limitations may relate to accelerometer quality, placement, challenges in real-world scenarios, and generalizability to diverse populations and settings.

2.4.11. A Comparison of Accuracy of Fall Detection Algorithms (Threshold-Based vs. Machine Learning) Using Waist-Mounted Tri-Axial Accelerometer Signals from a Comprehensive Set of Falls and Non-Fall Trials

In this paper, the authors conducted a comparative study on the accuracy of fall detection algorithms, specifically evaluating the performance of threshold-based methods and machine learning techniques. They collected data from waist-mounted tri-axial accelerometers, which served as the foundation for their analysis. The findings of this study shed light on which approach, threshold-based or machine learning, demonstrated superior accuracy in fall detection and under what circumstances.

2.4.12. Impact of Sampling Rate on Wearable-Based Fall Detection Systems Based on Machine Learning Models

This paper investigates the impact of sampling rate on wearable-based fall detection systems, employing machine learning models. It examines how varying sampling rates affect the performance of these systems in terms of accuracy, sensitivity, specificity, and other metrics. The study aims to determine the optimal sampling rate for effective fall detection. Nevertheless, limitations may arise from the choice of wearable sensors, data representativeness, and practical considerations regarding the applicability of different sampling rates in real-world scenarios.

2.4.13. Activity-Aware Fall Detection and Recognition Based on Wearable Sensors

This paper investigates activity-aware fall detection and recognition utilizing wearable sensors. The methodology likely includes data collection from these sensors, feature engineering, and the implementation of machine learning models for the task. Findings from this study should shed light on the system's effectiveness in accurately distinguishing falls from other activities, as well as its performance metrics. However, limitations may include challenges related to accuracy and real-world applicability, particularly in distinguishing falls from other activities.

2.4.14. A Smartphone-Based Fall Detection System

This paper presents a smartphone-based fall detection system. The methodology likely involves the development of a smartphone application or algorithm for fall detection, utilizing smartphone sensor data like accelerometers or gyroscopes. Findings from this study should provide insights into the effectiveness of the system, including its accuracy, sensitivity, specificity, and real-time capabilities. However, limitations may encompass issues such as sensor placement, data quality, and practical considerations for real-world use, including battery life and smartphone positioning.

2.4.15. Detecting Falls with Wearable Sensors Using Machine Learning Techniques

This paper investigates the detection of falls using wearable sensors in combination with machine learning techniques. The methodology likely involves data collection from these sensors, feature engineering, and the application of machine learning models. The findings from this study are expected to reveal the effectiveness of machine learning techniques in fall detection, including metrics such as accuracy, sensitivity, specificity, and insights into the performance of specific machine learning algorithms. However, limitations may include considerations related to sensor quality and placement, challenges in real-world fall scenarios, and practical issues like sensor maintenance and user comfort.

2.4.16. Detecting Falls as Novelties in Acceleration Patterns Acquired with Smartphones

This paper delves into the realm of fall detection by utilizing smartphone acceleration patterns. The methodology likely involves data collection from smartphone sensors, feature extraction, and the application of novelty detection techniques. The findings from this study are expected to reveal the effectiveness of detecting falls as novelties in smartphone-acquired acceleration patterns, including metrics such as accuracy, sensitivity, specificity, and insights into the system's capability to distinguish falls from regular activities based on novelty detection. However, limitations may encompass considerations related to the quality and accuracy of smartphone sensor data, challenges in discerning falls as novelties, and practical aspects, including variations in sensor placement and smartphone models.

2.4.17. Novel Hierarchical Fall Detection Algorithm Using a Multiphase Fall Model

This paper introduces a novel hierarchical fall detection algorithm that leverages a multiphase fall model. The methodology likely encompasses the development of this algorithm, based on the phases of the fall model. Findings from this study should reveal the algorithm's effectiveness, including metrics such as accuracy, sensitivity, specificity, and insights into how the multiphase fall model enhances fall detection. However, the paper may discuss limitations associated with real-world performance, challenges in model training, and considerations for adapting the algorithm to diverse populations or settings.

2.4.18. Accelerometer-Based Human Fall Detection Using Convolutional Neural Networks

This paper explores human fall detection by employing accelerometer data and Convolutional Neural Networks (CNNs). The methodology is likely to include data collection, preprocessing, and the description of the CNN model architecture. Findings from this study should shed light on the effectiveness of CNNs for accelerometer-based human fall detection, including metrics such as accuracy, sensitivity, specificity, and insights into the advantages of using CNNs in this context. However, the paper may discuss limitations related to accelerometer quality and placement, challenges in distinguishing falls from other activities, and practical considerations for real-world applications, including power consumption and sensor positioning.

2.4.19. Human Fall Detection on Embedded Platform Using Depth Maps and Wireless Accelerometer

This paper explores human fall detection through the utilization of depth maps and a wireless accelerometer on an embedded platform. The methodology likely involves developing a fall detection algorithm that integrates data from these sources. Findings from this study should provide insights into the effectiveness of the proposed fall detection system, including metrics such as accuracy, sensitivity, specificity, and the advantages of combining depth maps and accelerometer data for improved fall detection. However, the paper may discuss limitations related to the quality and availability of depth maps, challenges in distinguishing falls from other activities, and practical considerations for deploying the fall detection system on embedded platforms.

2.4.20. Human Fall Detection from Acceleration Measurements Using a Recurrent Neural Network

This paper delves into human fall detection through the utilization of acceleration measurements, employing a Recurrent Neural Network (RNN) as part of its methodology. The methodology likely encompasses data collection, preprocessing, and the description of the RNN model's architecture for fall detection. Findings from this study should offer insights into the effectiveness of using an RNN for human fall detection based on acceleration measurements, including performance metrics like accuracy, sensitivity, specificity, and the role of RNNs in this context. However, the paper may discuss limitations related to accelerometer quality and placement, challenges in distinguishing falls from other activities, and practical considerations for real-world applications, including sensor positioning and model training data.

2.4.21. Fall Detection System for the Elderly Based on the Classification of Shimmer Sensor Prototype Data

This paper introduces a fall detection system for the elderly, leveraging data from a Shimmer sensor prototype. The primary focus is on developing a classification algorithm for detecting falls based on sensor data. Findings show the system's effectiveness in detecting falls among elderly users, but potential limitations are discussed, including sensor quality and fall differentiation challenges.

2.4.22. An Event-Triggered Machine Learning Approach for Accelerometer-Based Fall Detection

This paper explores fall detection using accelerometer data with an event-triggered machine learning approach. It aims to reduce false alarms and improve the accuracy of fall detection. Findings should highlight the system's effectiveness and its potential to distinguish falls from other activities. The paper may discuss practical challenges in real-world use.

2.4.23. Evaluation of Accelerometer-Based Fall Detection Algorithms on Real-World Falls

The paper evaluates accelerometer-based fall detection algorithms using real-world fall data. It examines the performance and accuracy of these algorithms when applied in practical scenarios. The findings shed light on how well these algorithms can detect real-life falls, but potential limitations and challenges in generalization are considered.

2.4.24. Developing a Mobile Phone-Based Fall Detection System on Android Platform

This project focuses on creating a fall detection system for mobile phones, particularly on the Android platform. The methodology details the choice of sensors, data collection, and the development of the Android application. Findings discuss the system's effectiveness in detecting falls and practical considerations, such as user acceptance and battery life.

2.4.25. Research of Fall Detection and Fall Prevention Technologies: A Systematic Review

This paper conducts a systematic review of research related to fall detection and prevention technologies. It aims to provide insights into the state of the field, including trends, technologies, and gaps in the literature. The paper discusses challenges and potential future directions, along with practical applications in healthcare and aging populations.

2.4.26. Wearable Fall Detector Using Recurrent Neural Networks

This work involves the development of a wearable fall detection system utilizing recurrent neural networks (RNNs). The methodology outlines the use of sensors, RNNs, and data preprocessing techniques. Findings should cover the system's effectiveness in detecting falls and practical considerations such as user comfort and real-time capabilities.

2.4.27. Development of a Wearable-Sensor-Based Fall Detection System

This project focuses on creating a wearable-sensor-based fall detection system. The methodology describes the design, sensors, and algorithms used for fall detection. Findings should highlight the system's effectiveness, potential limitations, and practical considerations, such as user acceptance and device comfort.

2.4.28. Survey on Fall Detection and Fall Prevention Using Wearable and External Sensors

This paper conducts a survey of research and technologies related to fall detection and prevention using wearable and external sensors. It offers insights into trends, challenges, and practical applications in improving safety and well-being. The paper discusses limitations and potential future research directions in the field.

2.4.29. An Internet of Things-Based Fall Detection System for Patients with Neurological Disorders Using Recurrent Neural Networks

This paper presents an Internet of Things (IoT)-based fall detection system tailored for patients with neurological disorders. The system employs recurrent neural networks (RNNs) to enhance fall detection accuracy by combining data from various sensors, including smartphones/wearables and cameras. The findings demonstrate the system's effectiveness in managing patient safety by detecting falls and anomalies in daily activities. While the abstract doesn't specify limitations, common challenges may involve issues related to false positives or data privacy.

2.4.30. Sensitivity and False Alarm Rate of a Fall Sensor in Long-Term Fall Detection in the Elderly

This study evaluates a fall detection system's sensitivity and false alarm rate in real-life, long-term conditions. Using accelerometry-based sensors, the system was tested on a substantial dataset of older individuals, including both fallers and nonfallers. Findings reveal a good sensitivity rate, with 80% of real-life falls detected, and a low false alarm rate. While not explicitly mentioned in the abstract, limitations could include the need for further validation and considerations regarding system acceptance and accuracy.

Numerous studies have explored fall detection and healthcare technologies, each offering unique insights. A personalized fall detection system focuses on customization for enhanced accuracy and recall [1,2]. Wearable motion sensors and machine learning algorithms play a key role in fall detection [3], with an emphasis on sensor diversity and the potential of methods like SVM and k-NN. Medication adherence is improved through smartphones, NFC, and web technologies [4], despite limitations like sample size and ethical concerns.

The details of datasets used by literature survey have been carefully collected from various publicly accessible sources as shown in the below Table 1, following a thorough examination of existing research. These datasets have been widely utilized in previous studies and publications, indicating their importance in the field of fall detection.

Table 1. Publicly available wearables-based Datasets.

Reference	Dataset	Sensors Used	Type of Data
[5]	Smartwatch	MS Band2	Falls, ADLs
[5]	Notch	MS Band2	7 ADLs, 4 Falls
[6]	Usc-HAD	Single MotionNode, Miniature laptop	12 ADLs
[6]	LDPA	Wearing four tags (left, right ankle, belt, chest)	11 ADLs
[6]	The German Aerospace Center (Dlr)	Inertial sensor	7 ADLs

[6,7]	Farseeing	ActivePAL3, McRobert Dynaport minimode	23 Falls
[8,9]	MobiAct	Smartphone	9 ADLs, 4 Falls
[8,9]	MobiAct RealWorld (HAR)	Smartphone	9 ADLs, 4 Falls
[9]	UMA Fall	4 Bluetooth sensors motes, Smartphone	ADL, 3 Falls
[9]	Shoaib PA, Shoaib SA	-	ADLs
[9]	tFall	Accelerometer	ADLs, 4 Falls
[9]	UCI HAR, UCI HAPT	Smartphone	6 ADLs
[9]	DMPSBFD	Smartphone	ADLs, Falls
[10]	KTH	Static camera	6 ADLs
[9,11]	MobiFall	Smartphone	ADLs and Falls
[11]	SKODA	20 Accelerometer	Gestures
[12,13]	SisFall	2 Accelerometer, Gyroscope	19 ADL, 15 Fall
[14]	SmartFall	Smartwatch	ADLs and Falls
[15]	UR Fall	Microsoft Kinect cameras, Accelerometer	5 ADLs, 4 Falls
[15]	UP Fall	Wearable, Ambient sensors, Vision devices	6 ADLs, 5 Falls
[16]	DaLiAc	4 SHIMMER sensors	3 ADLs
[17]	mHealth	4 Sensors	12 ADLs
[17]	UbiqLog	Smartphone, Smartwatch	-
[17]	CrowdSignals	Smartphone, Smartwatch	8 Activities
[17]	ExtraSensory	Smartphone, Smartwatch	51 Activities
[18]	FSP	5 Smartphones	7 Activities
[18]	RFID	RFID sensor	ADLs
[18]	Smartphone	Smartphone	7 Activities
[19]	SBHAR	Smartphone	6 ADLs
[19]	WISDM V.1.1 and V.2.0	Smartphone	6 ADLs
[20]	UniMiB SHAR	Smartphone	9 ADLs, 8 Falls
[21]	PAMAP2	3 IMUs, Heart rate monitor	18 Activities
[22]	CASAS	-	ADLs
[23]	KARD	Kinect sensor	18 Activities
[24]	CAD-60	MS Kinect sensor	12 Activities

We also collected the details regarding the algorithms used and accuracies achieved according to the literature survey as shown in the below Table 2.

Table 2. Algorithms used and accuracies achieved according to the literature survey.

Citation	Android	IOS	Front Fall	Back Fall	Right Fall	Left Fall	Dataset	Accelerometer	Gyroscope	Algorithm	Accuracy in %
[24]	Y	Y	Y	Y	Y	Y	N/A	Y	N	N/A	80
[25]	Y	N	N	N	N	N	Y	Y	N	TBA	83.3–95.8
[26]	Y	N	Y	Y	Y	Y	Y	Y	N	TBA	0
[27]	Y	N	Y	Y	Y	Y	Y	Y	Y	TBA	86.67
[28]	Y	N	Y	Y	Y	Y	Y	Y	Y	TBA	0
[29]	Y	N	Y	Y	Y	Y	Y	Y	N	TBA	99.38
[30,31]	Y	N	Y	Y	Y	Y	Y	Y	Y	TBA, MKL-SVM, SVM, ANN, K-NN, Naive Bayes	91.7
[32]	Y	N	-	-	-	-	Y	Y	Y	TBA, SVM, DT, RF, KNN, Naive Bayes	78.63–96.65
[33]	Y	N	Y	Y	Y	Y	Y	Y	Y	Naive Bayes, SVM, ANN, LSM	87.5
[34]	Y	N	-	-	-	-	Y	Y	Y	K-NN, LSM, SVM, ANN	98
[35]	Y	N	-	-	-	-	Y	Y	N	TBA, KNN, ANN, SVM, J48	91.83
[36]	Y	N	N	N	N	N	Y	Y	N	TBA, ANN, Fuzzy Logic, AdaBoost	0
[37]	Y	N	Y	Y	Y	Y	Y	Y	Y	TBA, DT, K-NN, Naive Bayes	77.5–93.7

Older individuals benefit from a fall detection system that utilizes wearable accelerometers and classical machine learning algorithms [38].

A systematic review delves into machine learning for fall detection and prevention, addressing research trends and potential biases [39]. Elderly fall detection systems are surveyed, covering various techniques and technologies [40]. Multi-class fall detection explores the impact of sampling rates [41], while contactless fall detection employs time-frequency analysis and Convolutional Neural Networks [42,43]. Fall detection using wearable sensors [44] and machine learning emphasizes precision and specificity [45]. A smartwatch-based system utilizes deep learning techniques [46]. Understanding public datasets for wearable fall detection is emphasized [47], and fall detection algorithms are compared, highlighting accuracy disparities [48]. These studies collectively contribute to our understanding of fall detection and healthcare technology, addressing both challenges and future potential.

Through a detailed analysis, we aim to gain a deeper understanding of the patterns present within these datasets. The datasets referenced in this pilot study have been specifically chosen for their relevance to IoT-based fall detection systems [49–51]. This comprehensive approach ensures that our findings are grounded in a solid foundation of established data, allowing for meaningful insights to be drawn.

3. Methodology

3.1. Collecting Dataset

SisFall Dataset:

The SisFall dataset was sourced from the Signal Processing Laboratory (LaPS) repository at the University of Porto. This dataset comprises accelerometer and gyroscope data collected from wearable sensors worn by individuals during various activities, including falls and activities of daily living (ADLs). The dataset encompasses recordings from 19 participants, providing a diverse range of movement patterns and scenarios.

UMAFall Dataset:

The UMAFall dataset was obtained from the Falls Laboratory repository at the University of Malaga. This dataset contains accelerometer data captured from smartphones carried by subjects during simulated falls and routine activities. It encompasses data from multiple participants across different age demographics and encompasses various types of falls, such as forward, backward, and lateral falls.

The sample image of the accelerometer data from the datasets is shown in Figure 1.

3.2. Preprocessing the Dataset

The accelerometer data, sampled at 20 Hz, is initially represented in nine columns within each data file, with varying row counts corresponding to test durations. The first three columns denote the acceleration readings along the X, Y, and Z axes, respectively, as captured by the ADXL345 sensor for SisFall and MPU-9250 sensor for UMAFall where both had same configurations while collection.To render the accelerometer data interpretable, a conversion from bits to gravitational units (g) is necessary. The sensors, operating with a resolution of 13 bits and a range of ±16g, necessitates the utilization of a conversion equation:

$$Acceleration\ [g] = \left(\frac{2 \times Range}{2^{Resolution}}\right) \times AD$$

where:

- Range = ±16
- Resolution = 13

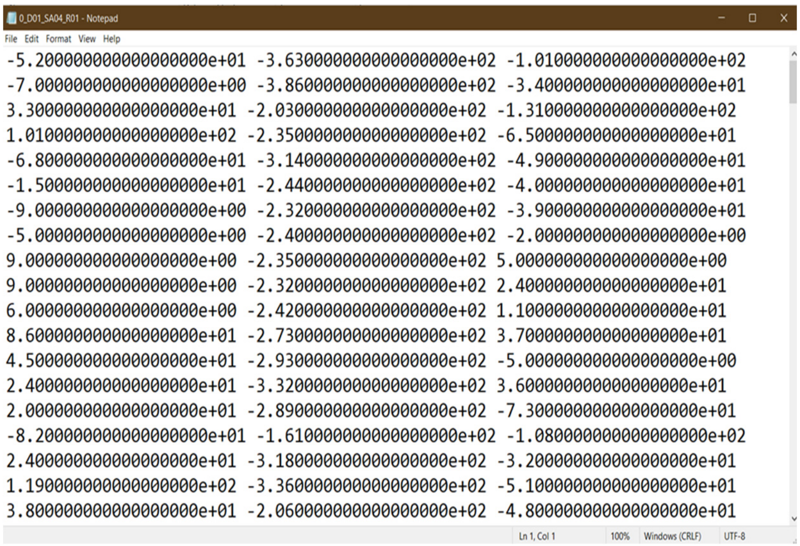
This conversion is implemented within the preprocessing pipeline to ensure the data's consistency and meaningfulness for subsequent analysis.

The preprocessing workflow segments the data into smaller windows, facilitating more granular analysis. Adjustable parameters such as window size and step size offer flexibility in customizing the segmentation process to suit specific experimental needs.

Furthermore, the dataset undergoes label encoding to represent activities of daily living (ADL) and fall events numerically. This encoding ensures compatibility with deep learning algorithms for classification tasks.Finally, the preprocessed dataset is partitioned into training and validation subsets using an 80:20 train-test split ratio. This separation enables the evaluation of model performance on unseen data, contributing to the robustness and generalizability of the analysis.

3.3. Developing Deep Learning Model for Fall Detection

Two distinct models were crafted for this purpose, each uniquely tailored to capture relevant features indicative of fall events. In Model-1, a combination of Convolutional Neural Network (CNN) layers followed by Bidirectional Long Short-Term Memory (Bi-LSTM) layers is employed, enabling the extraction of both spatial and temporal information crucial for accurate detection. Incorporating batch normalization and dropout layers enhances model robustness and mitigates overfitting risks. On the other hand, Model-2 utilizes a blend of Bi-LSTM and Conv1D layers, strategically interspersed with dropout and layer normalization techniques, to effectively capture intricate temporal dependencies inherent in the data. Both models are designed to culminate in dense layers followed by sigmoid activation functions, facilitating binary fall detection. Through rigorous evaluation against benchmark datasets, these models aim to significantly enhance fall detection accuracy and contribute to the advancement of dependable assistive technologies for fall prevention. The architecture of both the models are visually depicted in Figures 2 and 3.



0_D01_SA04_R01 - Notepad
File Edit Format View Help

```
-5.200000000000000000e+01 -3.630000000000000000e+02 -1.010000000000000000e+02
-7.000000000000000000e+00 -3.860000000000000000e+02 -3.400000000000000000e+01
3.300000000000000000e+01 -2.030000000000000000e+02 -1.310000000000000000e+02
1.010000000000000000e+02 -2.350000000000000000e+02 -6.500000000000000000e+01
-6.800000000000000000e+01 -3.140000000000000000e+02 -4.900000000000000000e+01
-1.500000000000000000e+01 -2.440000000000000000e+02 -4.000000000000000000e+01
-9.000000000000000000e+00 -2.320000000000000000e+02 -3.900000000000000000e+01
-5.000000000000000000e+00 -2.400000000000000000e+02 -2.000000000000000000e+00
9.000000000000000000e+00 -2.350000000000000000e+02 5.000000000000000000e+00
9.000000000000000000e+00 -2.320000000000000000e+02 2.400000000000000000e+01
6.000000000000000000e+00 -2.420000000000000000e+02 1.100000000000000000e+01
8.600000000000000000e+01 -2.730000000000000000e+02 3.700000000000000000e+01
4.500000000000000000e+01 -2.930000000000000000e+02 -5.000000000000000000e+00
2.400000000000000000e+01 -3.320000000000000000e+02 3.600000000000000000e+01
2.000000000000000000e+01 -2.890000000000000000e+02 -7.300000000000000000e+01
-8.200000000000000000e+01 -1.610000000000000000e+02 -1.080000000000000000e+02
2.400000000000000000e+01 -3.180000000000000000e+02 -3.200000000000000000e+01
1.190000000000000000e+02 -3.360000000000000000e+02 -5.100000000000000000e+01
3.800000000000000000e+01 -2.060000000000000000e+02 -4.800000000000000000e+01
```

Ln 1, Col 1 100% Windows (CRLF) UTF-8

Figure 1. Sample image of the accelerometer data from the datasets.

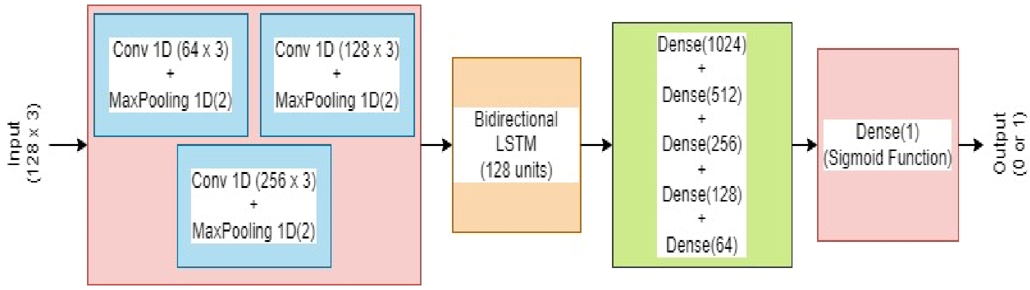


Figure 2. Overview of Model-1 Architecture.

3.3.1. Model-1 Architecture

Model-1 is a deep learning architecture designed for fall detection applications, combining Convolutional Neural Network (CNN) and Bidirectional Long Short-Term Memory (Bi-LSTM) layers to analyze accelerometer data. The architecture begins with a series of Conv1D layers, each followed by MaxPooling1D layers, to extract spatial features from the input accelerometer readings across different axes. These layers employ the rectified linear unit (ReLU) activation function to introduce non-linearity and downsample the feature maps, reducing computational complexity while preserving important features.

Following the convolutional layers, a Bidirectional LSTM (Bi-LSTM) layer is introduced to capture temporal dependencies within the accelerometer data. This layer processes the input sequences in both forward and backward directions, enabling the model to learn from past and future timestamps simultaneously. Batch normalization is applied to stabilize the training process and improve generalization.

After the Bi-LSTM layer, a Flatten layer reshapes the output into a one-dimensional vector, which is then passed through a series of densely connected (Dense) layers. These layers progressively extract high-level features and non-linear transformations from the data. Dropout layers are inserted after each dense layer to prevent overfitting by randomly dropping a fraction of neurons during training.

The final output layer consists of a single neuron with a sigmoid activation function, producing a probability score indicating the likelihood of a fall event. This architecture facilitates binary classification, where a threshold can be applied to determine whether a fall event is detected based on the probability score.

Model-1's architecture enables the comprehensive analysis of accelerometer data for fall detection, leveraging both spatial and temporal information to achieve accurate classification results. Now let's look into Figure 3 for model-2 architecture.

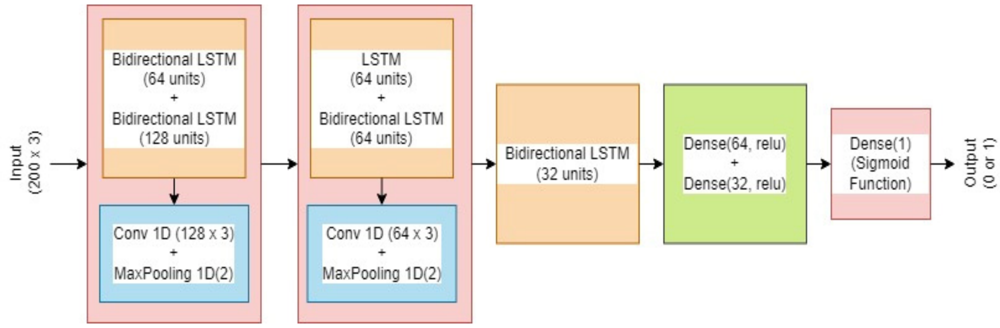


Figure 3. Overview of Model-2 Architecture.

3.3.2. Model-2 Architecture

Model-2 is a deep learning architecture for fall detection tasks, incorporating a combination of Bidirectional Long Short-Term Memory (Bi-LSTM), Conv1D, and densely connected (Dense) layers to analyze accelerometer data. The architecture begins with a Bidirectional LSTM layer with 64 units and return sequences enabled, allowing the model to capture temporal dependencies within the input data effectively. A dropout layer with a dropout rate of 0.2 is applied to mitigate overfitting by randomly deactivating a portion of the neurons during training.

Following the initial Bi-LSTM layer, another Bidirectional LSTM layer with 128 units and return sequences enabled further captures temporal patterns in both forward and backward directions. Another dropout layer with a dropout rate of 0.2 is introduced to enhance model generalization. Subsequently, a Conv1D layer with 128 filters and a kernel size of 3, coupled with the ReLU activation function, extracts spatial features from the input data.

After the convolutional layer, a MaxPooling1D layer with a pool size of 2 is applied to down sample the feature maps, reducing computational complexity while retaining essential features. A subsequent LSTM layer with 64 units and return sequences enabled continues to capture temporal dependencies, followed by a Layer Normalization to stabilize the training process.

A dropout layer with a dropout rate of 0.3 is introduced to prevent overfitting before the model proceeds to another Bidirectional LSTM layer with 64 units and return sequences enabled, capturing additional temporal dependencies. Another dropout layer with a dropout rate of 0.3 is applied for further regularization.

After the bidirectional LSTM layer, a Conv1D layer with 64 filters and a kernel size of 3, followed by a MaxPooling1D layer with a pool size of 2, extracts additional spatial features from the data. Subsequently, a Bidirectional LSTM layer with 32 units further processes the temporal information.

This architecture has densely connected (Dense) layers, including two layers with 64 and 32 units, respectively, and ReLU activation functions. These layers facilitate the extraction of high-level features and non-linear transformations from the data. Finally, the output layer consists of a single neuron with a sigmoid activation function, enabling binary classification based on the probability score generated by the model.

Model-2's architecture is characterized by its sophisticated design, leveraging a combination of LSTM and convolutional layers to effectively analyze accelerometer data for fall detection purposes.

Key Points about both the architectures:

1. Purpose: Both Model-1 and Model-2 are designed for fall detection applications, aiming to analyze accelerometer data to accurately detect fall events.
2. Architectural Components: Both architectures incorporate a combination of Convolutional Neural Network (CNN) and Bidirectional Long Short-Term Memory (Bi-LSTM) layers to extract spatial and temporal features from the accelerometer data.
3. Spatial and Temporal Analysis: Model-1 primarily focuses on spatial analysis by employing Conv1D layers followed by pooling layers, while Model-2 emphasizes temporal analysis with Bidirectional LSTM layers capturing long-term dependencies in the data.
4. Output Layer: Both architectures culminate in a single neuron output layer with a sigmoid activation function, facilitating binary classification based on the probability score indicating the likelihood of a fall event.
5. Model Complexity: Model-2 exhibits a more intricate architecture compared to Model-1, incorporating additional Bidirectional LSTM layers and convolutional layers for comprehensive spatial and temporal analysis.

3.4. Training and Evaluation

3.4.1. Model-1

Initially, model-1 compiles using a binary cross-entropy loss function and the Adam optimizer with an initial learning rate of 0.001. It sets the initial batch size to 64 and runs for 20 epochs.

Subsequently, it defines early stopping and learning rate reduction callbacks to monitor validation accuracy during training. The early stopping callback halts training if validation accuracy does not improve for 5 consecutive epochs, while the learning rate reduction callback decreases the learning rate by a factor of 0.5 if validation accuracy does not improve for 3 consecutive epochs, with a minimum learning rate of 10^{-6}.

The model then enters a loop that iterates five times. Within each iteration, the model trains on the training data with the specified batch size and epochs, using the early stopping and learning rate reduction callbacks. After training, it checks if the validation accuracy decreased compared to the previous iteration. If so, it adjusts the hyperparameters by decreasing the batch size by 20% (with a minimum of 8) and increasing the number of epochs by 5. The model is then recompiled with the updated hyperparameters. This process continues until the validation accuracy stops decreasing.

Finally, the model can be used with the final hyperparameters determined during the iterative training process as per the Figure 4.

```
315/315 [==============================] - 3s 11ms/step - loss: 0.2466 - accuracy: 0.9220 - val_loss: 0.2065 - val_accuracy: 0.9471 - lr: 0.0010
Epoch 8/40
312/315 [============================>.] - ETA: 0s - loss: 0.1784 - accuracy: 0.9458
Epoch 8: ReduceLROnPlateau reducing learning rate to 0.0005000000237487257.
315/315 [==============================] - 4s 11ms/step - loss: 0.1778 - accuracy: 0.9459 - val_loss: 0.1839 - val_accuracy: 0.9593 - lr: 0.0010
Epoch 9/40
315/315 [==============================] - 4s 13ms/step - loss: 0.1293 - accuracy: 0.9644 - val_loss: 0.1932 - val_accuracy: 0.9618 - lr: 5.0000e-04
Epoch 10/40
315/315 [==============================] - 3s 11ms/step - loss: 0.1126 - accuracy: 0.9672 - val_loss: 0.1845 - val_accuracy: 0.9634 - lr: 5.0000e-04
Epoch 11/40
315/315 [==============================] - 3s 11ms/step - loss: 0.1182 - accuracy: 0.9654 - val_loss: 0.1908 - val_accuracy: 0.9644 - lr: 5.0000e-04
Epoch 12/40
315/315 [==============================] - 4s 13ms/step - loss: 0.1119 - accuracy: 0.9674 - val_loss: 0.1808 - val_accuracy: 0.9659 - lr: 5.0000e-04
Epoch 13/40
315/315 [==============================] - 4s 12ms/step - loss: 0.1044 - accuracy: 0.9683 - val_loss: 0.1778 - val_accuracy: 0.9649 - lr: 5.0000e-04
Epoch 14/40
315/315 [==============================] - 3s 11ms/step - loss: 0.1204 - accuracy: 0.9631 - val_loss: 0.2075 - val_accuracy: 0.9603 - lr: 5.0000e-04
Epoch 15/40
313/315 [============================>.] - ETA: 0s - loss: 0.1146 - accuracy: 0.9651
Epoch 15: ReduceLROnPlateau reducing learning rate to 0.0002500000118743628.
315/315 [==============================] - 3s 11ms/step - loss: 0.1148 - accuracy: 0.9649 - val_loss: 0.1795 - val_accuracy: 0.9613 - lr: 5.0000e-04
Epoch 16/40
315/315 [==============================] - 4s 14ms/step - loss: 0.1110 - accuracy: 0.9667 - val_loss: 0.1789 - val_accuracy: 0.9654 - lr: 2.5000e-04
Epoch 17/40
315/315 [==============================] - 3s 11ms/step - loss: 0.1018 - accuracy: 0.9692 - val_loss: 0.1957 - val_accuracy: 0.9639 - lr: 2.5000e-04
```

Figure 4. Overview of Model-1 Training.

The training process involved multiple epochs with a progressive reduction in loss and increase in accuracy, indicating an effective learning process. Initially, the model exhibited a loss of 0.6989 and an accuracy of 0.6292, which gradually improved over subsequent epochs. Notably, by the fifth epoch, the loss decreased to 0.3145, accompanied by an accuracy of 0.8768, highlighting significant progress. This trend continued, with the model achieving higher accuracy and lower loss values with each epoch.

Epochs 6 and 7 marked a notable improvement in both loss reduction and accuracy enhancement. The loss decreased to 0.2557, while the accuracy increased to 0.9139 by the seventh epoch. This signifies that the model was learning complex patterns in the data and making more accurate predictions. The validation loss and accuracy metrics also demonstrated consistency, indicating that the model was generalizing well to unseen data.

Further improvements were observed in epochs 10 and 11, where the model achieved its highest accuracy of 0.9523 while maintaining a low loss value. This suggests that the model was effectively capturing the underlying patterns in the data. Additionally, the validation accuracy closely tracked the training accuracy, indicating minimal overfitting.

The learning rate was dynamically adjusted throughout training, with reductions occurring when performance plateaued. This adaptive learning rate strategy likely contributed to the model's stability and convergence to a good solution by evaluating Figure 5.

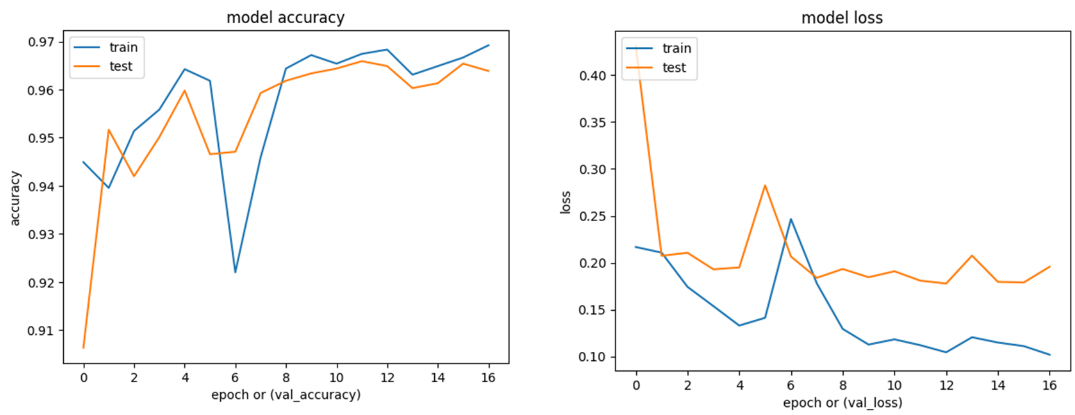


Figure 5. Evaluation of Model-1.

3.4.2. Model-2

Initially, consider the Figure 6, model-2 is compiled with a binary cross-entropy loss function and the Adam optimizer, using an initial learning rate of 0.0001. The model is configured with a batch size of 128 and trained for 10 epochs.

Early stopping is implemented, which monitors validation accuracy and halts training if there is no improvement for 5 consecutive epochs. It also restores the best weights observed during training.

The model is then trained on the training data (X_train and y_train) with the specified batch size and epochs, while monitoring validation performance using the validation data (X_val and y_val). The early stopping callback is applied during training.

After training, the model is recompiled with the same initial learning rate and metrics.

```
Epoch 1/10
67/67 [==============================] - 39s 134ms/step - loss: 0.3826 - accuracy: 0.9013 - val_loss: 0.2175 - val_accuracy: 0.9331
Epoch 2/10
67/67 [==============================] - 6s 83ms/step - loss: 0.1633 - accuracy: 0.9470 - val_loss: 0.1452 - val_accuracy: 0.9402
Epoch 3/10
67/67 [==============================] - 5s 76ms/step - loss: 0.1205 - accuracy: 0.9568 - val_loss: 0.1194 - val_accuracy: 0.9562
Epoch 4/10
67/67 [==============================] - 5s 76ms/step - loss: 0.1028 - accuracy: 0.9653 - val_loss: 0.1143 - val_accuracy: 0.9567
Epoch 5/10
67/67 [==============================] - 6s 85ms/step - loss: 0.0946 - accuracy: 0.9687 - val_loss: 0.1037 - val_accuracy: 0.9656
Epoch 6/10
67/67 [==============================] - 5s 76ms/step - loss: 0.0869 - accuracy: 0.9724 - val_loss: 0.0990 - val_accuracy: 0.9619
Epoch 7/10
67/67 [==============================] - 5s 82ms/step - loss: 0.0784 - accuracy: 0.9769 - val_loss: 0.0858 - val_accuracy: 0.9713
Epoch 8/10
67/67 [==============================] - 5s 78ms/step - loss: 0.0751 - accuracy: 0.9774 - val_loss: 0.0868 - val_accuracy: 0.9718
Epoch 9/10
67/67 [==============================] - 5s 79ms/step - loss: 0.0757 - accuracy: 0.9765 - val_loss: 0.0798 - val_accuracy: 0.9703
Epoch 10/10
67/67 [==============================] - 6s 84ms/step - loss: 0.0722 - accuracy: 0.9765 - val_loss: 0.0826 - val_accuracy: 0.9722
```

Figure 6. Overview of Model-2 Training.

The training process involved ten epochs, each progressively refining the model's performance. In the initial epoch, the model achieved a loss of 0.3826 and an accuracy of 0.9013 on the training data, with validation metrics of 0.2175 for loss and 0.9331 for accuracy. This indicated a strong starting point, with room for improvement.

As training continued, the model consistently improved, with the loss decreasing to 0.0722 and accuracy increasing to 0.9765 by the tenth epoch. This demonstrated the model's ability to learn from the training data and make increasingly accurate predictions. Moreover, the validation metrics showed similar trends, with the loss reaching 0.0826 and accuracy peaking at 0.9722.

Throughout the epochs, the model showcased robust performance, with minimal overfitting and stable convergence. The training and validation metrics closely tracked each other, indicating that the model generalized well to unseen data. Additionally, the training process was computationally efficient, with each epoch completing in a reasonable timeframe, by evaluating the Figure 7.

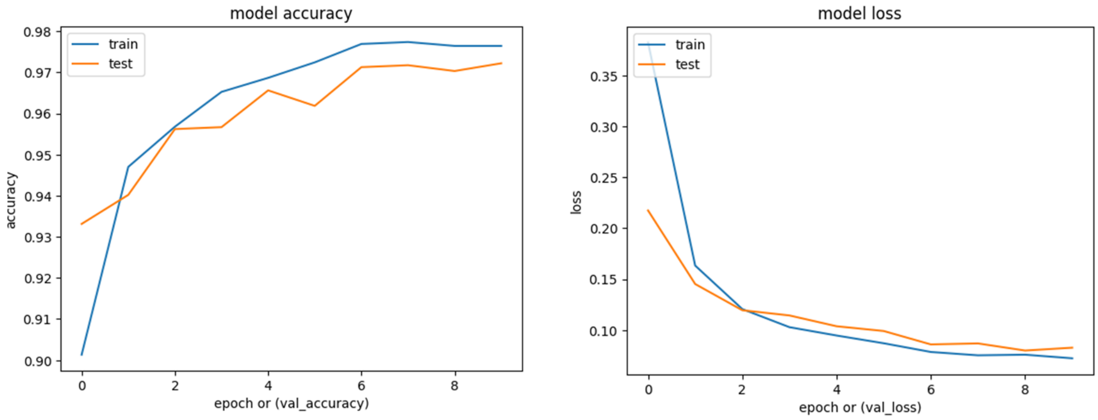


Figure 7. Evaluation of Model-2.

3.5. Integrating Deep Learning Model with Mobile Application and IOT

To enable real-time fall detection and medication reminders, deep learning models are integrated into a mobile application. This integration enables the processing of accelerometer data in real-time, allowing for immediate detection of fall events. Additionally, the application provides medication reminders based on user-defined schedules, enhancing proactive healthcare management. A proposed UI is shown in Figures 8 and 9 as shown below.

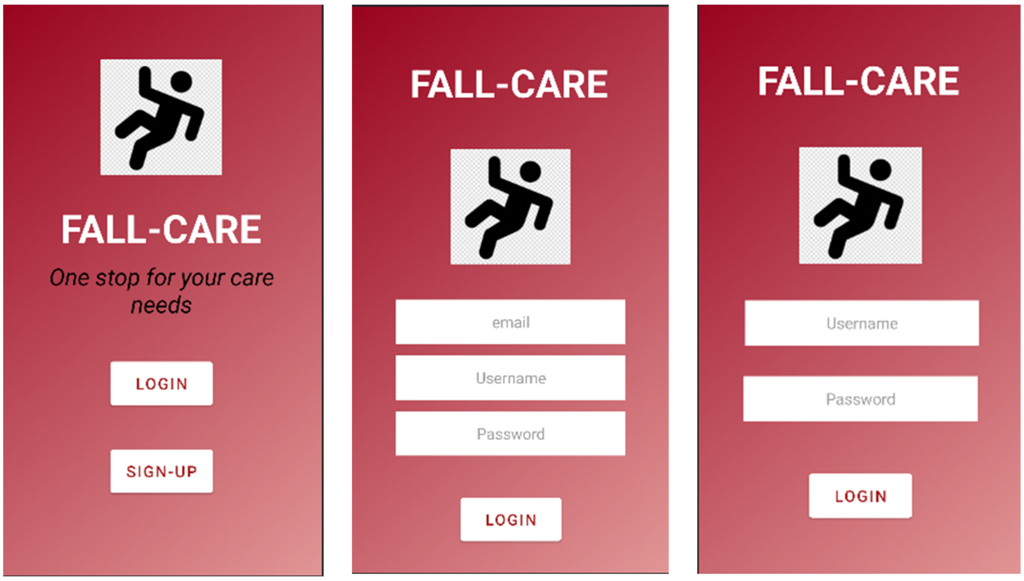


Figure 8. Login & Sign-Up Page of Mobile application.

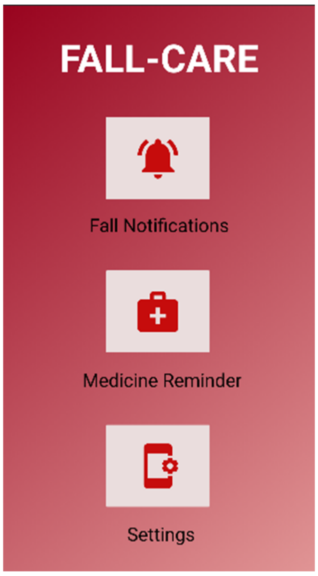


Figure 9. Notification and Reminder Page of Mobile application.

Building upon the conceptual framework a physical prototype is implemented to portray the fall detection system using the specified components. At the core of the prototype is the NodeMCU ESP8266 microcontroller, which serves as the central processing unit and the communication hub for the system. The prototype integrates the MPU6050 sensor module (as shown in Figure 10), which houses a 3-axis accelerometer and a 3-axis gyroscope. This sensor array is responsible for capturing the user's motion data, including acceleration and angular velocity, in real-time. The sensor module is connected to the NodeMCU (as shown in Figure 11) using the I2C communication protocol, allowing for seamless data transfer between the components.

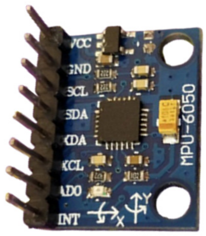


Figure 10. MPU6050 sensor.

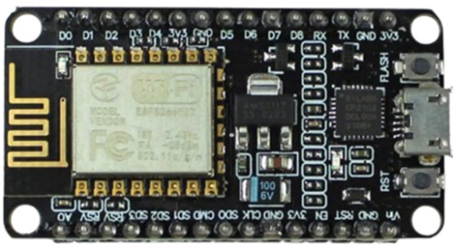


Figure 11. NodeMCU ESP8266 microcontroller.

To facilitate the fall detection algorithm, the custom firmware on the NodeMCU is implemented. This firmware continuously monitors the incoming sensor data, analyzes the acceleration and angular velocity patterns, and applies the deep learning models described in the paper to identify potential fall events. The deep learning models, which were trained on labeled datasets of falls and normal activities, enable the prototype to distinguish between these scenarios with a high degree of accuracy.

Upon the detection of a fall, the prototype triggers an alarm mechanism, through leveraging the Wi-Fi connectivity of the NodeMCU to interface with a mobile application and cloud-based services, such as IFTTT, to promptly notify designated caregivers or emergency contacts about the fall event.

The prototype's design below in the Figure 12 also incorporates a user-friendly wristband form factor.

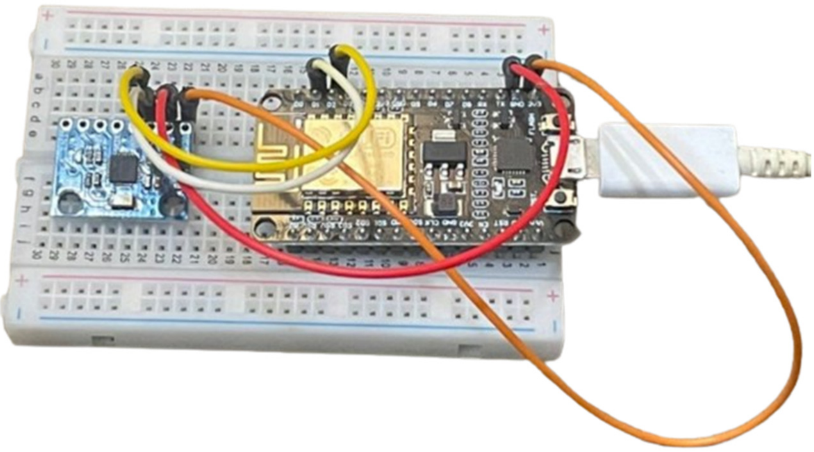


Figure 12. Prototype.

3.6. Monitoring with the integration of IoT, Deep Learning, and Mobile Applications

Combining IoT, deep learning, and mobile applications allows for comprehensive monitoring solutions that leverage real-time data processing and intelligent analysis. By integrating IoT devices such as sensors or cameras with deep learning algorithms, the system can capture and analyze data from the physical environment. This integration enables the detection of complex patterns or anomalies that may indicate potential issues or events of interest.

Mobile applications serve as a user-friendly interface for accessing and interacting with the monitoring system. Users can receive notifications, view real-time data, and control IoT devices remotely through the application. Deep learning algorithms deployed on the mobile device or cloud server analyze the data collected from IoT devices, providing insights and actionable information to users.

For example, in the context of fall detection, IoT sensors worn by individuals can continuously monitor movement patterns. Data from these sensors are transmitted to a mobile application, where deep learning algorithms analyze the data in real-time to detect falls. If a fall is detected, the application can immediately notify caregivers or emergency services, enabling timely assistance.

The integration of IoT, deep learning, and mobile applications offers powerful monitoring solutions that enhance situational awareness, enable proactive decision-making, and improve overall safety and efficiency. Hence, the design flow of the application is shown in Figure 13 as shown below.

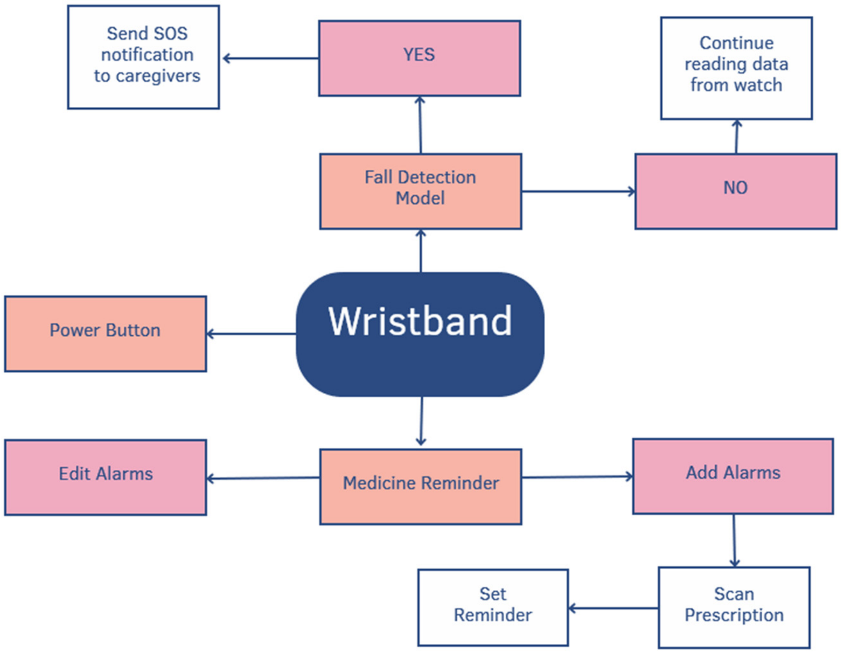


Figure 13. Overview of the integration of IOT, Deep Learning and Mobile Application.

4. Results and Discussion

Two models were developed and evaluated for a classification task. The first model from Figure 14 achieved a validation accuracy of 96.59%, with a loss of 0.1808, while the second model from the Figure 15 outperformed the first, achieving a validation accuracy of 97.22% and a loss of 0.0826. These results demonstrate clear improvements in performance between the two models as shown in the above Figures 11 and 12.

Analyzing the results, it’s evident that both models performed well in terms of accuracy. However, the second model exhibited superior performance with higher accuracy and lower loss compared to the first model. This indicates that the second model was better able to generalize to unseen data and make more accurate predictions.

The improvement observed in the second model can be attributed to several factors. Firstly, the architecture of the second model is more sophisticated, allowing it to capture more complex patterns in the data. Additionally, the second model has been trained for a longer duration or with more extensive data augmentation techniques, leading to better generalization.

```
62/62 [==============================] - 2s 4ms/step - loss: 0.1808 - accuracy: 0.9659
Validation Accuracy: 96.59%
```

Figure 14. Model-1 Validation Accuracy.

```
67/67 [==============================] - 7s 23ms/step - loss: 0.0826 - accuracy: 0.9722
Validation Accuracy: 97.22%
```

Figure 15. Model-2 Validation Accuracy.

5. Conclusions

In conclusion, the proposed fall detection system, driven by deep learning, offers a robust and user-friendly approach to addressing fall-related concerns. Through the synergistic combination of a sensor-equipped wristband and a feature-rich mobile application, the system empowers users to maintain an independent lifestyle while ensuring prompt assistance during fall events. Furthermore, the medication reminder functionality adds an extra layer of support, aiding users in managing their healthcare regimen. This highlights a holistic solution that underscores the potential of technology to safeguard well-being and promote healthy living in everyday context. In addition, it involved the development and evaluation of two models for a classification task. While both models exhibited strong performance in terms of accuracy, the second model surpassed the first with higher accuracy and lower loss. This improvement can be attributed to factors such as model architecture, training duration, and data augmentation techniques. However, it's essential to consider computational efficiency and scalability when selecting the final model for deployment. Overall, the results underscore the importance of iterative model development and evaluation in achieving optimal performance for real-world applications.

Author Contributions

M.L.G., K.G.Y., M.K.P. and R.A. conceived of the presented idea. M.L.G. and K.G.Y. developed the theory and performed the computations. M.L.G. and R.A. verified the analytical methods. K.G.Y. and M.L.G. encouraged M.L.G. to investigate [a specific aspect] and supervised the findings of this work. All authors discussed the results and contributed to the final manuscript. All authors have read and agreed to the published version of the manuscript.

Funding

No funding is received for this Article.

Conflict of Interest Statement

The authors have NO Conflict of Interest.

Data Availability Statement

In this study, data supporting the reported results were obtained from publicly archived datasets. Specifically, the SisFall dataset was utilized, which can be accessed at https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5298771/, and the UMAFall dataset available at https://figshare.com/articles/dataset/UMA_ADL_FALL_Dataset_zip/4214283. For transparency and reproducibility, both datasets are openly accessible and were essential for conducting the experiments reported in this study.

References

1. Mansi Lakshmi Gidugu , Krishna Guptha Yanduri , Manoj Kumar Paliviri , Rajanikanth Aluvalu. Wristband for monitoring the safety of elderly people using IoT and Deep Learning algorithms: A Review. *15th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2023).*
2. Ngu, A. H., Metsis, V., Coyne, S., Chung, B., Pai, R., & Chang, J. (2020, March). Personalized fall detection system. In *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)* (pp. 1-7). IEEE.
3. Kavuncuoğlu, E., Uzunhisarcıklı, E., Barshan, B., & Özdemir, A. T. (2022). Investigating the Performance of Wearable Motion Sensors on recognizing falls and daily activities via machine learning. *Digital Signal Processing*, *126*, 103365
4. Orcioni, S., Pellegrini, R., Seepold, R., Gaiduk, M., Madrid, N. M., & Conti, M. (2021). Medication adherence supported by mHealth and NFC. *Informatics in Medicine Unlocked*, 23, 100552.
5. R. Mauldin, Taylor, Marc E. Canby, Vangelis Metsis, Anne H. H. Ngu and Coralys Cubero Rivera. (2018) "Smartfall: A smartwatch-based fall detection system using deep learning." *Sensors* vol 18, 1–19.
6. Liu, Zhi, Yankun Cao, Lizhen Cui, Jiahua Song, and Guangzhe Zhao. (2018) "A Benchmark Database and Baseline Evaluation for Fall Detection Based on Wearable Sensors for the Internet of Medical Things Platform." *IEEE Access* vol 6, 51286–51296.
7. Hind, M., Noura, O., & Abraham, A. (2023). Modeling IoT based Forest Fire Detection System with IoTsec. *Int. J. Comput. Inf. Syst. Ind. Manag. Appl.,* 15, 201-213.
8. Ajerla, Dharmitha, Sazia Mahfuz and Farhana Zulkernine. (2019) "A real-time patient monitoring framework for fall detection." *Wireless Communications and Mobile Computing* vol 2019, Article ID 9507938.
9. Micucci, Daniela, Marco Mobilio and Paolo Napoletano. (2017) "UniMiB SHAR: A dataset for human activity recognition using acceleration data from smartphones." *Applied Science* 7, 1101.
10. Hsieh, Yi-Zeng and Yu-Lin Jeng. (2017) "Development of Home Intelligent Fall Detection IoT System Based on Feedback Optical Flow Convolutional Neural Network." *IEEE Access* 6, 6048–6057.
11. He, Jian, Zihao Zhang, Xiaoyi Wang, and Shengqi Yang. (2019) "A low power fall sensing technology based on fd-cnn." *IEEE Sensors Journal* Vol. 19, NO. 13, 5110–5118.
12. Yacchirema, Diana, Jara Suarez de Puga, Carlos Palau and Manuel Esteve. (2019) "Fall detection system for elderly people using IoT and ensemble machine learning algorithm." Personal and Ubiquitous Computing.
13. Jin Park, Yu, Seol Young Jung, Tae Yong Son and Soon Ju Kang. (2021) "Self-Organizing IoT Device-Based Smart Diagnosing Assistance System for Activities of Daily Living." *Sensors*, vol 21, 785.
14. Farsi, Mohammed. (2021) "Application of ensemble RNN deep neural network to the fall detection through IoT environment." *Alexandria Engineering Journal*, vol 60, 199–211.
15. M. Galvao, Yves, Janderson Ferreira, Vinicius A. Albuquerque, Pablo Barros and Bruno J.T. Fernandes. (2021) "A multimodal approach using deep learning for fall detection." *Expert Systems with Applications,* vol 168,114226.
16. Zdravevski, Eftim, Petre Lameski, Vladimir Trajkovik and Andrea Kulakov. (2017) "Improving Activity Recognition Accuracy in Ambient-Assisted Living Systems by Automated Feature Engineering." *IEEE Access* 5, 5262–5280.
17. Vitabile, Salvatore, Michal Marks and Dragan Stojanovic. (2019) "Medical Data Processing and Analysis for Remote Health and Activities Monitoring." *Medical Data Processing and Analysis* 186–220.
18. Awais, Muhammad, Mohsin Raza, Kamran Ali, Zulfiqar Ali, Muhammad Irfan, Omer Chughtai, Imran Khan, Sunghwan Kim and Masood Ur Rehman. (2019) "An internet of things based bed-egress alerting paradigm using wearable sensors in elderly care environment." *Sensors* 19, 1–17.

19. Zheng, Xiaochen, Meiqing Wang and Joaquín Ordieres-Mere. (2018) "Comparison of data preprocessing approaches for applying deep learning to human activity recognition in the context of industry 4.0." *Sensors* 18, 2146.
20. Al-kababji, Ayman, Abbes Amira, Faycal Bensaali, Abdulah Jarouf, Lisan Shidqi, Hamza Djelouat. (2021) "Biomedical Signal Processing and Control An IoT-based framework for remote fall monitoring." *Biomedical Signal Processing and Control*, vol 67, 102532.
21. Khowaja, Sunder Ali, Aria Ghora Prabono, Feri Setiawan, Bernardo Nugroho Yahya and Seok-Lyong Lee. (2018) "Contextual activity based Healthcare Internet of Things, Services, and People (HIoTSP): An architectural framework for healthcare monitoring using wearable sensors." *ComputerNetworks*, 145,190–206.
22. Al Zamil, Mohammed GH, Majdi Rawashdeh, Samer Samarah, M. Shamim Hossain, Awny Alnusair and SK MD Mizanur Rahman. (2017) "An Annotation Technique for In-Home Smart Monitoring Environments." *IEEE Access*, 6, 1471–1479.
23. Saini, Rajkumar, Pradeep Kumar, Barjinder Kaur, Partha Pratim Roy, Debi Prosad Dogra and K. C. Santosh. (2019) "Kinect sensor-based interaction monitoring system using the BLSTM neural network in healthcare." *International Journal of Machine Learning and Cybernetics*, 10, 2529–2540.
24. Shen, V.R.L., Lai, H.Y., Lai, A.F. (2013). Application of High-Level Fuzzy Petri Nets to fall detection system using smartphone. In *Proceedings of the 2013 International Conference on Machine Learning and Cybernetics*, Tianjin, pp. 1429- 1435.
25. Colón, L.N.V., DeLaHoz, Y., Labrador, M. (2014). Human fall detection with smartphones. In *Proceedings of the 2014 IEEE Latin-America Conference on Communications (LATINCOM)*, Cartagena de Indias, pp. 1-7.
26. Thammasat, E., Chaicharn, J. (2012). A simply fall detection algorithm using accelerometers on a smartphone. In *Proceedings of the 5th 2012 Biomedical Engineering International Conference*, Ubon Ratchathani, pp. 1-4.
27. Rakhman, A.Z., Nugroho, L.E., Widyawan, Kurnianingsih. (2014). Fall detection system using accelerometer and gyroscope based on smartphone. In *Proceedings of the 2014 The 1st International Conference on Information Technology, Computer, and Electrical Engineering*, Semarang, pp. 99-104.
28. Lee, J., Tseng, H. (2019). Development of an enhanced threshold-based fall detection system using smartphones with built-in accelerometers. *IEEE Sensors Journal*, 19(18): 8293-8302.
29. Hsieh, S., Yang, C., Li, H. (2017). Combining wristbandtype devices and smartphones to detect falls. In *Proceedings of the 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Banff, AB, pp. 2373-2377.
30. Forth, K., Aiden, E.L. (2019). Identifying Fall Risk Using Machine Learning Algorithms. U.S. Patent.
31. Shahzad, A., Kim, K. (2019). FallDroid: An automated smart-phone-based fall detection system using multiple kernel learning. *IEEE Transactions on Industrial Informatics*, 15(1): 35-44.
32. Dogan, J.C., Hossain, M.S. (2019). A novel two-step fall detection method using smartphone sensors. In *Proceedings of the 2019 IEEE International Conference on Smart Computing (SMARTCOMP)*, Washington, DC, USA, pp. 434-438.
33. Vallabh, P., Malekian, R., Ye, N., Bogatinoska, D.C. (2016). Fall detection using machine learning algorithms. In *Proceedings of the 2016 24th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, Split, pp. 1-9.
34. Cheffena, M. (2016). Fall detection using smartphone audio features. *IEEE Journal of Biomedical and Health Informatics*, 20(4): 1073-1080.
35. Tsinganos, P., Skodras, A. (2017). A smartphone-based fall detection system for the elderly. In *Proceedings of the 10th International Symposium on Image and Signal Processing and Analysis*, Ljubljana, pp. 53-58.
36. Madansingh, S., Thrasher, T.A., Layne, C.S., Lee, B. (2015). Smartphone based fall detection system. In *Proceedings of the 2015 15th International Conference on Control, Automation and Systems (ICCAS)*, Busan, pp. 370-374..
37. Aguiar, B., Rocha, T., Silva, J., Sousa, I. (2014). Accelerometer-based fall detection for smartphones. In *Proceedings of the 2014 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, Lisboa, pp. 1-6.
38. Al Nahian, M. J., Ghosh, T., Al Banna, M. H., Aseeri, M. A., Uddin, M. N., Ahmed, M. R., e al. (2021). Towards accelerometer-based elderly fall detection system using cross-disciplinary time series features. *IEEE Access*, 9, 39413-39431.
39. Usmani, S., Saboor, A., Haris, M., Khan, M. A., & Park, H. (2021). Latest research trends in fall detection and prevention using machine learning: A systematic review. *Sensors*, 21(15), 5134.
40. Tanwar, R., Nandal, N., Zamani, M., & Manaf, A. A. (2022, January). Pathway of trends and technologies in fall detection: a systematic review. *Healthcare* (Vol. 10, No. 1, p. 172).
41. Thakur, N., & Han, C. Y. (2021). A study of fall detection in assisted living: Identifying and improving the optimal machine learning method. *Journal of Sensor and Actuator Networks*, 10(3).
42. Wang, X., Ellul, J., & Azzopardi, G. (2020). Elderly fall detection systems: A literature survey. Frontiers in *Robotics and AI*, 7, 71.
43. Sadreazami, H., Bolic, M., & Rajan, S. (2021). Contactless fall detection using time-frequency analysis and convolutional neural networks. *IEEE Transactions on Industrial Informatics*, 17(10), 6842-6851.
44. Zurbuchen, N., Wilde, A., & Bruegger, P. (2021). A machine learning multi-class approach for fall detection systems based on wearable sensors with a study on sampling rates selection. *Sensors*, 21(3), 938.
45. Özdemir, A. T., & Barshan, B. (2014). Detecting falls with wearable sensors using machine learning techniques. *Sensors*, 14(6), 10691-10708.
46. Casilari, E., Santoyo-Ramón, J. A., & Cano-García, J. M. (2017). Analysis of public datasets for wearable fall detection systems. *Sensors*, 17(7), 1513.

47. Palmerini, L., Klenk, J., Becker, C., & Chiari, L. (2020). Accelerometer-based fall detection using machine learning: Training and testing on real-world falls. *Sensors*, 20(22), 6479.
48. Aziz, O., Musngi, M., Park, E. J., Mori, G., & Robinovitch, S. N. (2017). A comparison of accuracy of fall detection algorithms (threshold-based vs. machine learning) using waist-mounted tri-axial accelerometer signals from a comprehensive set of falls and non-fall trials. *Medical & Biological Engineering & Computing*, 55, 45-55.
49. Kaur, R., & Sharma, R. (2022). Analyzing Wearables Dataset to Predict ADLs and Falls: A Pilot Study. *arXiv* arXiv:2209.04785.
50. Hind, M., Noura, O., & Abraham, A. (2023). Modeling IoT based Forest Fire Detection System with IoTsec. Int. J. Comput. Inf. Syst. Ind. Manag. Appl, 15, 201-213.
51. Aldarwish, A.J., Patel, K., Yassin, A.A., & Yaseen, A.A. (2023). Virtual SmartCards-based Authentication in Healthcare Systems and Applications. *International Journal of Computer Information Systems & Industrial Management Applications*, 15.